



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

Aplicación de la plataforma virtual Chamilo como un apoyo para la impartición de la materia Programación Orientada a Objetos

C A S O P R Á C T I C O
Que para obtener el Título de:
INGENIERO EN COMPUTACIÓN
P R E S E N T A:
Roberto Téllez Vallejo
Asesora: Dra. Nelly Rigaud Téllez



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS Y DEDICATORIAS.

Agradezco a mis padres por haberme apoyado durante todos mis estudios y después de haber terminado mis estudios, ya que esto llevo a que se terminara este trabajo.

Agradezco a mis profesores de la FES Aragón y a los profesores que conocí en C.U, ya que gracias a los conocimientos brindados y a el esfuerzo hicieron para que yo aprendiera, he logrado muchas de mis metas y sueños.

Agradezco a mis amigos de la Universidad y amigos de toda la vida por haber estado conmigo en momentos difíciles, en momentos de felicidad y haberme brindado su apoyo siempre.

Agradezco a nuestro creador, ya que a pesar de no practicar ninguna religión siempre he pensado que existe algo superior que nos creó de alguna manera y que me permitió existir y vivir la grandiosa vida que he llevado hasta el momento.

También agradezco a mi hermana Cinthia por haber estado conmigo y por siempre haberme apoyado en lo que estaba a su alcance

Agradezco a mi asesora la Dra. Nelly Rigaud por haberme apoyado en la realización de este trabajo así como por su paciencia y tiempo que invirtió en mí.

Además quiero agradecer a las compañeras de pedagogía Sandra y Yahaira por el apoyo que me dieron para en la parte práctica de este trabajo.

Finalmente quiero dedicar este trabajo a mi padre Jose I. Tellez Pinzón, ya que siempre me mostro el camino, siempre me aconsejo y porque siempre vi y sigo viendo en él un ejemplo a seguir.

La verdadera educación consiste en obtener lo mejor de uno mismo. ¿Qué otro libro se puede estudiar mejor que el de la Humanidad?

Mahatma Gandhi (1869-1948) Político y pensador indio.

RESUMEN

El objetivo principal de esta implementación es crear material didáctico de apoyo a profesores y alumnos de la materia Programación Orientada Objetos (POO) de la carrera Ingeniería en Computación, este material consta del temario propuesto de la materia POO, el desarrollo de un capítulo de este temario, además de los ejercicios de este capítulo.

En la implementación este caso práctico se hace uso de las Tecnologías de la Información y Comunicación (TIC)¹. En específico se utiliza una plataforma educativa virtual para almacenar y administrar material didáctico generado en este trabajo. Se decidió ocupar estas tecnologías con la meta poner a disponibilidad los recursos didácticos y para proporcionar a los profesores, herramientas necesarias para mejorar el proceso de enseñanza - aprendizaje en la materia de Programación Orientada a Objetos, además para poder generar sinergias entre los grupos de trabajo en el aula de clase.

El software utilizado en este caso práctico se lista a continuación:

- Apache HTTP Server versión 2.2.20
- PHP versión 5.3.4
- My SQL Server 5.5
- Plataforma virtual educativa Chamilo versión 1.8

En cuestiones de hardware solo se necesita una computadora que hará la función de servidor de datos, esta deberá tener más de 250GB de disco duro y una conexión de banda ancha (en el apéndice se dan más detalles y características sobre el hardware recomendado).

¹Según la *Asociación americana de las tecnologías de la información* (Information Technology Association of America, ITAA): sería «*el estudio, el diseño, el desarrollo, el fomento, el*

ÍNDICE

RESUMEN	1
ÍNDICE.....	2
ÍNDICE DE FIGURAS Y DIAGRAMAS.....	3
INTRODUCCIÓN.....	6
1. PLANTEAMIENTO DEL PROBLEMA	10
1.1 JUSTIFICACIÓN Y DELIMITACIÓN DEL PROBLEMA.....	11
1.1.1 LOS PROBLEMAS DE LOS PROCESOS DE ENSEÑANZA Y APRENDIZAJE EN LA UNIVERSIDAD.....	12
2. OBJETIVOS.....	19
2.1. OBJETIVO PRINCIPAL.....	20
2.2. SUB OBJETIVOS.....	20
2.3. ESTRATEGÍA PARA ALCANZAR LOS OBJETIVOS.....	20
3. MARCO TEÓRICO DE REFERENCIA.....	24
3.1. PARADIGMAS Y LENGUAJES DE PROGRAMACIÓN.....	25
3.1.1. LOS LENGUAJES DE PROGRAMACIÓN	27
3.1.2. PROGRAMACIÓN ESTRUCTURADA.....	31
3.1.3. PROGRAMACIÓN ORIENTADA A COMPONENTES	32
3.1.4. PROGRAMACIÓN ORIENTADA A ASPECTOS	33
3.2. LA METODOLOGÍA DE PROGRAMACIÓN ORIENTADA A OBJETOS.....	34
3.3. CONCEPTOS BÁSICOS SOBRE SISTEMAS DE GESTIÓN DE APRENDIZAJE.....	41
4. ANÁLISIS, DISEÑO E IMPLEMENTACIÓN.....	46
4.1. ANÁLISIS DEL OBJETIVO PRINCIPAL DEL USO DE CHAMILO.....	47
4.2. DISEÑO DE LOS MÓDULOS PARA LA IMPLEMENTACIÓN DEL MATERIAL DIDÁCTICO DE POO.....	48
4.3. DESCRIPCIÓN DE LOS MÓDULOS QUE CONFORMAN LA PLATAFORMA EDUCATIVA CHAMILO.....	49
4.4. TEMARIO PROPUESTO PARA LA MATERIA PROGRAMACIÓN ORIENTADA A OBJETOS.....	53
4.5. PROCESO DE IMPLEMENTACIÓN DEL CAPÍTULO 1 DE LA MATERIA POO.....	54
4.6 PRÁCTICAS OCUPANDO CHAMILO.....	65
5. CONCLUSIONES Y RECOMENDACIONES.....	75
APENDICE	79

ÍNDICE DE FIGURAS Y DIAGRAMAS

Figura 1. Gráfica de pastel de los índices de reprobación de la carrera Ing. en Computación, periodo 2011-1. (FES Aragón U. , 2011)	13
Figura 2. Etapas del desarrollo de los contenidos	18
Figura 3. Sub – Objetivo 1	21
Figura 4. Sub - Objetivo 2	22
Figura 5. Sub - Objetivo 3	22
Figura 6. Objetivo Principal	23
Figura 7. Circuito de la comunicación.	25
Figura 8. Fases que componen la estructura de un compilador.	27
Figura 9. Relación entre las instrucciones de alto nivel, ensamblador y máquina.	29
Figura 10. Proceso de conversión entre lenguajes que realiza un ensamblador.	29
Figura 11. Encapsulación de clases reusables en un componente.	33
Figura 12. Pilares de la Programación Orientada a Objetos.....	35
Figura 13. Diagrama UML en el que se ilustra el mecanismo de herencia en la POO.	38
Figura 14 UML de la clase Imagen	39
Figura 15. Diagrama de UML donde se representa la herencia.	40
Figura 16.Esquematzación de bLearning http://www.langrow.com	43
Figura 17. Manejo de contenidos plataforma virtual educativa Chamilo.....	48
Figura 18. Interacción, plataforma virtual educativa Chamilo.	48
Figura 19. BPMN (<i>Business Process Modeling Notation</i>) de los procesos que se seguirán para la implementación del caso practico	2
Figura 20. Administración de usuarios Chamilo.	58
Figura 21. Pantalla Principal Chamilo.....	59
Figura 22. Descripción General.....	60
Figura 23. Interfaz de usuario de Chamilo para las lecciones.	61
Figura 24. Interfaz de Usuario para crear ejercicios.	62
Figura 25. Interfaz de usuario de Chamilo para poder agregar preguntas a los ejercicios.....	62
Figura 26. Chamilo, agregar objeto de aprendizaje.....	63
Figura 27. Chamilo, agregar ejercicios a una lección.	64
Tabla 1. Cuadro comparativo de materias con alto índice de reprobación compartidas por las ingenierías (FES Aragón U. , 2011)	14
Tabla 2. Cuadro comparativo de las materias con alto índice de reprobación de la FES Aragón. (FES Aragón U. , 2011)	15
Tabla 3. Comparativa de Plataformas Educativas.....	44

Ilustración 1. Plataforma Chamilo 1.....	66
Ilustración 2. Plataforma Chamilo 2.....	67
Ilustración 3. Plataforma Chamilo 3.....	67
Ilustración 4. Plataforma Chamilo 4.....	68
Ilustración 5. Plataforma Chamilo 5.....	68
Ilustración 6. Plataforma Chamilo 6.....	69
Ilustración 7. Descripción General.....	70
Ilustración 8. Descripción General 2.....	70
Ilustración 9. Paradigmas de Programación.....	71
Ilustración 10. ¿Qué es un objeto?.....	71
Ilustración 11. Mensajes entre objetos.....	72
Ilustración 12. Preguntas abiertas.....	72
Ilustración 13. Preguntas de opción múltiple.....	73
Ilustración 14. Preguntas varias.....	73
Ilustración 15. Tipos de preguntas.....	74
Ilustración 16.....	81
Ilustración 17.....	82
Ilustración 18.....	82
Ilustración 19.....	83
Ilustración 20.....	83
Ilustración 21.....	84
Ilustración 22.....	84
Ilustración 23.....	85
Ilustración 24.....	85
Ilustración 25.....	86
Ilustración 26.....	86
Ilustración 27.....	87
Ilustración 28.....	87
Ilustración 29.....	88
Ilustración 30.....	88
Ilustración 31.....	89
Ilustración 32.....	90
Ilustración 33.....	90
Ilustración 34.....	91
Ilustración 35.....	91
Ilustración 36.....	92
Ilustración 37.....	92
Ilustración 38.....	93
Ilustración 39.....	93
Ilustración 40.....	94
Ilustración 41.....	94
Ilustración 42.....	95

Ilustración 43	95
Ilustración 44	96
Ilustración 45	96
Ilustración 46	97
Ilustración 47	97
Ilustración 48	98
Ilustración 49	101
Ilustración 50	102
Ilustración 51	102
Ilustración 52	103
Ilustración 53	103
Ilustración 54	104
Ilustración 55	104
Ilustración 56	105
Ilustración 57	105
Ilustración 58	106
Ilustración 59	107
Ilustración 60	107



INTRODUCCIÓN



En esta sección se busca dar una breve presentación de las ventajas que dan las tecnologías aplicadas en la educación. Además se presenta un resumen del caso práctico desarrollado en este trabajo, la estructura de todo el trabajo y una breve descripción de lo que se aborda en cada uno de los capítulos.

El uso de nuevas tecnologías en la educación es y ha sido de vital importancia durante mucho tiempo, ya que de esta manera se ha logrado generar un mayor impacto sobre el nivel de conocimientos que se puede adquirir durante la impartición de un curso. Un ejemplo de este impacto se puede observar en el uso las tecnologías actuales, ya que es como se han logrado romper barreras espacio-temporales. Hoy en día, cualquier persona, en cualquier parte de mundo y a cualquier hora, tiene la posibilidad de aprender o tomar una clase, siempre y cuando cuente con una conexión a internet y un dispositivo que permita visualizar los contenidos educativos.

Las tecnologías aplicadas en la educación han sido utilizadas desde épocas muy remotas en las que se utilizaban dispositivos como los ábacos y calculadoras hasta recientemente con la aparición de los dispositivos móviles y las computadoras. Estos últimos son dispositivos de alta importancia ya que a través de las computadoras se ha podido magnificar el uso y distribución de programas educativos con un enfoque de docencia y la posibilidad de generar contenidos digitales con enfoques culturales. Hoy en día con el uso de smartphones, el iPad, los dispositivos móviles, las computadoras, internet y en general los dispositivos que se comunican a través de redes de información; se ha logrado un mayor impacto sobre la distribución de contenidos educativos, esto debido a que la mayoría de estos dispositivos son portables y es posible descargar o almacenar contenidos educativos a través de internet y visualizar estos en cualquier lugar y hora

Como un apoyo para desarrollar este caso práctico se hace uso de las tecnologías que son aplicadas a la educación, y más específicamente de aquellas que se basan en el uso de internet como un medio para la transmisión de conocimientos y como un medio de retroalimentación entre el alumno y el profesor. Estas tecnologías son conocidas como LSM (por sus siglas en inglés *Learning Management Systems*) y se basan en los conceptos de *bLearning* y *eLearning*². Estos sistemas de aprendizaje virtuales son accesibles a través de internet y son de magnifica utilidad a profesores ya que se les sirve como una herramienta adicional para impartir sus clases.

En este trabajo se realiza la implementación de un caso práctico para almacenar contenidos didácticos utilizando la plataforma educativa virtual Chamilo, este material se creó como un medio de apoyo a las clases de profesores de la materia Programación Orientada a Objetos (POO) que se imparte en carreras como Ingeniería en Computación, Ciencias de la Computación, Informática, etc. El

² Los conceptos de *bLearning* y *eLearning* son diferentes aunque *bLearning* incluye a *eLearning*, estos conceptos son definidos más adelante.

material didáctico consta del desarrollo de los recursos necesarios para un capítulo de la materia de POO, incluyendo ejercicios de ejemplo resueltos y ejercicios que se dejan para resolver al alumno. Con este ejemplo se muestra el procedimiento para crear el material del primer capítulo y sus ejercicios correspondientes, por lo que el desarrollo de los capítulos posteriores y de los ejercicios faltantes, se dejan como tarea al profesor de la asignatura.

Además con este trabajo se pretende que los profesores y alumnos de la materia POO, obtengan un panorama general de lo que es eLearning y bLearning, el impacto que estas tecnologías tienen sobre la educación en el nivel superior y en el caso de profesores sabrán cual es el procedimiento para crear sus herramientas didácticas que le servirán de apoyo a sus clases y conocerán las ventajas que proporciona el uso de las plataformas virtuales educativas en la docencia.

La estructura y el contenido del trabajo es la siguiente:

El nombre del primer capítulo es **“PLANTEAMIENTO DEL PROBLEMA”**. En este capítulo se justifica el desarrollo de este caso práctico y se presentan las consecuencias negativas que se observan hoy en día en el proceso de enseñanza-aprendizaje que se usa en México.

Para el segundo capítulo **“OBJETIVOS”** se presentan los objetivos de la implementación de este caso práctico. En este capítulo se menciona los objetivos principales y secundarios, así como las tareas necesarias para poder cumplir este objetivo principal y la estrategia que se eligió para abordar este problema.

En el capítulo tres **“MARCO TEÓRICO DE REFERENCIA”** se aborda de manera general toda la teoría necesaria para poder cumplir los objetivos planteados en este trabajo, se abordan temas acerca de que son los lenguajes de programación, el paradigma de programación orientada a objetos, para que es utilizado el lenguaje de modelado UML y algunos conceptos básicos y definiciones sobre lo que son las plataformas educativas virtuales, en específico los conceptos se enfocan a la plataforma Chamilo. Con toda la teoría abordada en este capítulo se dan las herramientas de consulta necesarias para que el profesor tenga el panorama general para desarrollar el material didáctico de los temas faltantes de la materia POO.

El cuarto capítulo **“ANÁLISIS, DISEÑO E IMPLEMENTACIÓN”** es la parte principal de este trabajo, se presenta la propuesta del temario de la materia de POO, además se especifica cual fue el proceso de análisis, diseño e implementación que se siguió para obtener el material didáctico para el primer capítulo del temario. En este capítulo se colocan diagramas de cómo está estructurada la información, la descripción de cada una de las secciones que conforman la plataforma virtual

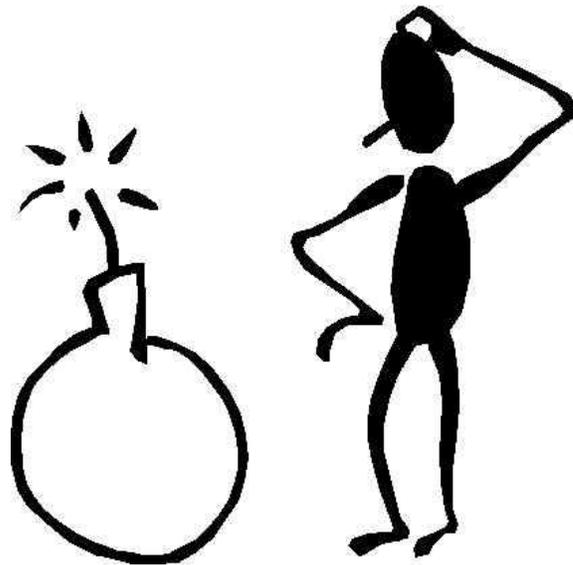
educativa, las relaciones que existen entre estos módulos y por qué se decidió tener estas secciones dentro de la plataforma educativa virtual Chamilo.

Finalmente en el último capítulo “**CONCLUSIONES Y RECOMENTACIONES**” se presentan los resultados y conclusiones que se obtuvieron al realizar la implementación de este caso práctico. El objetivo principal de este capítulo es vaciar la experiencia que se tuvo al realizar la implementación de todo el proceso, además de colocar las fallas y observaciones que se obtuvieron durante este, se podría decir que en este capítulo se coloca todo el aprendizaje obtenido durante todo el desarrollo.

Adicionalmente al trabajo realizado de esta implementación se proporcionan los manuales de Administrador, Profesor y Alumno de Chamilo, para que de esta manera se tengan fuentes de información para los diferentes usuarios de la plataforma educativa.



CAPITULO 1



1. PLANTEAMIENTO DEL PROBLEMA

En este capítulo se justifica el desarrollo de este caso práctico y se presentan las consecuencias negativas que se observan hoy en día en el proceso de enseñanza-aprendizaje que se usa en México, por ello se comienza con la estructuración de la problemática que se tiene en el ámbito educativo en México en el contexto universitario.

1.1 JUSTIFICACIÓN Y DELIMITACIÓN DEL PROBLEMA.

Un tema crítico en México es el de la calidad de educación, ya que actualmente se tiene un déficit educativo considerable en la población. Un punto importante a tocar es si se lograra combatir este problema, ya que permitiría un mejoramiento del desarrollo y calidad de vida, facilitando la convivencia entre estas.

Ha habido muchos esfuerzos para buscar remediar el problema educativo pero no han sido suficientes. Se han implementado técnicas educativas novedosas y se ha hecho uso de la tecnología pero no lo suficiente para tener una mejora importante en la educación.

En este contexto, es necesario analizar las deficiencias que se tienen en la educación superior sobre todo en las áreas de las ingenierías, ya que esto ha llevado a que los profesionales egresados de estas áreas no alcancen un nivel competitivo en el área laboral y que por lo tanto les cueste más trabajo encontrar empleo³.

Si se toma como referencia la problemática educativa de la Facultad de Estudios Superiores Aragón en las áreas de las ingenierías, se percibe que existen problemas tanto en la forma en que los profesores imparten los cursos y también en los alumnos que muchas veces no poseen los conocimientos necesarios para cierta materia.

Es notable los diversos estilos para la impartición de clases, debido a que en muchas de las ocasiones en que se imparte un curso, el profesor no le da un enfoque práctico parcial, cuya consecuencia lleva a que el alumno no comprenda completamente muchos de los conceptos que se presentan en la materia, en este punto es necesario mencionar que a través de la práctica es más fácil recordar un tema.

Otro de los problemas que surgen en el ámbito docente en la FES Aragón, se debe a las dificultades de intercambio de información y en la retroalimentación que hay entre el profesor y el alumno, ya que en ocasiones el alumno se queda con dudas acerca de algún tema en particular y por alguna razón muy particular el alumno no le pregunta al profesor, lo que conlleva a que el alumno se quede con la duda y no comprenda adecuadamente el tema.

³ La situación es más crítica para las personas con estudios de educación media superior y superior, para quienes la desocupación avanzó al pasar de 33% en el cuarto trimestre de 2008, o alrededor de 634 mil personas, a 35% de los desempleados en el cuarto trimestre de 2010, de acuerdo con información del Instituto Nacional de Estadística y Geografía (INEGI) (El Universal: Desempleo impactó más a profesionistas,).

La poca disponibilidad de tiempo para cumplir completamente los planes de estudio de la FES Aragón es otro problema, ya que en algunas de las veces los profesores no alcanzan a cumplir el plan de estudio propuesto debido a la poca duración de los periodos de clases y a la gran cantidad de temas que se tienen que ver. Esto llega a afectar en las materias subsecuentes en la carrera, ya que muchos de los temas que no se ven, son necesarios para entender conceptos que se toman en otras materias e incluso llega a pasar que en el temario no existen temas que son muy necesarios en la actualidad.

En el uso de recursos tecnológicos para las clases también se pueden hacer observaciones, ya que en algunas clases de Ingeniería en Computación se emplean parcialmente herramientas tales como el uso del chat para comunicarse con el profesor, uso de correo electrónico, uso de foros de discusión, uso de documentos online, uso de videos online para complementar temas, compartir artículos a través de repositorios, uso de software para conexión remota, uso de plataformas virtuales educativas, etc.

Por todas estas observaciones se nota que adquiere relevancia el uso de nuevas herramientas de enseñanza-aprendizaje, esto debido a los impactos negativos que se están teniendo en la educación superior de la actualidad. Estas herramientas tecnológicas tienen que apoyar a profesores en la impartición de clases, permitiendo una mayor interacción entre el alumno y el profesor, y fomentando la comprensión de los temas que se ven en clase; todo esto tomando en cuenta la integración de muchos de los recursos tecnológicos y los dispositivos móviles que son utilizados en la actualidad.

1.1.1 LOS PROBLEMAS DE LOS PROCESOS DE ENSEÑANZA Y APRENDIZAJE EN LA UNIVERSIDAD.

"Nada cambiará en educación, ni siquiera con tecnología, si previamente no se modifican los procedimientos pedagógicos" (BELTRAN LLERA, 2003)

Como ya se mencionaba en el apartado anterior, en ocasiones las técnicas de enseñanza y aprendizaje que son utilizadas en muchas de las universidades actuales son fragmentadas, esto es se utilizan técnicas inadecuadas y los procesos de enseñanza-aprendizaje que se ocupan no logran su objetivo. En estos procesos de enseñanza-aprendizaje naturalmente intervienen el profesor, los alumnos y las técnicas de enseñanza utilizadas.

Es importante considerar todos los problemas educativos que existen en nuestro entorno, un ejemplo de ello es la alta demanda de alumnos que desean ingresar a

la UNAM, esta gran cantidad de alumnos ha ido creciendo con el tiempo⁴ y por consecuencia se han ido acrecentando resultados negativos en el sector educativo de México, ya que se ha visto que cada vez se obtienen más alumnos reprobados independientemente de la gran cantidad de alumnos que sean aceptados. Lo anterior se puede ver en la FES Aragón, esto se refleja en la gráfica de pastel (FES Aragón U. , 2011) de la Figura 1, donde se indica el número de alumnos registrados y el número de alumnos aprobados y reprobados en la carrera de Ing. en Computación.

EXAMENES ORDINARIOS					
REGISTRADOS		APROBADOS		REPROBADOS	
TOTAL	%	TOTAL	%	TOTAL	%
3,825	100%	2,723	71%	1,102	29%

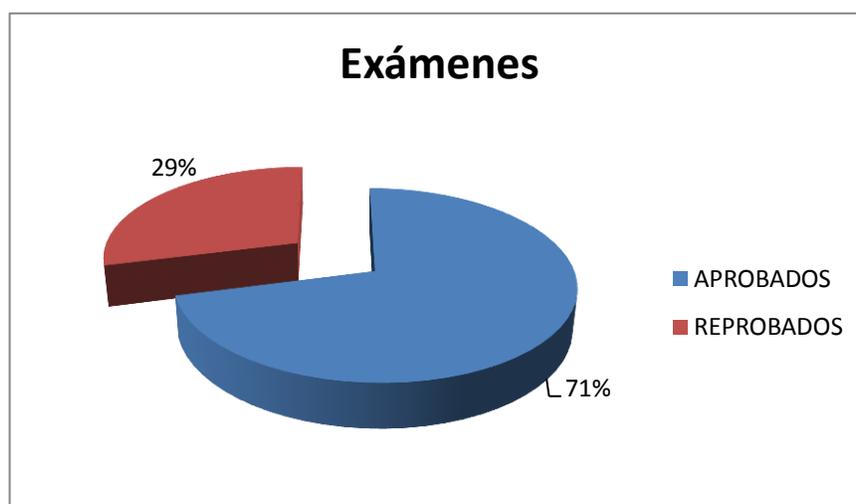


Figura 1. Gráfica de pastel de los índices de reprobación de la carrera Ing. en Computación, periodo 2011-1. (FES Aragón U. , 2011)

El uso inadecuado de estas técnicas de enseñanza-aprendizaje ha tenido repercusiones negativas en el sector educativo, principalmente en las universidades. En este caso en particular la investigación se va a centrar en las técnicas de enseñanza y en los recursos tecnológicos que se utilizan para impartir clases en la FES Aragón, donde también se han observado en diversos estudios, resultados negativos en los porcentajes de aprobación.

⁴ Para el periodo 2007-2008 hubo 161 mil 457 solicitudes y sólo se seleccionó a 15 mil 470 jóvenes. Para los ciclos 2008-2009, 2009-2010 y 2010-2011, el número de aspirantes fue de 174 mil 151, 175 mil 350 y 170 mil 558, respectivamente (Emir Olivares Alonso <http://www.jornada.unam.mx/2011/02/26/sociedad/039n1soc> consultada enero 2011)

Es necesario mencionar que en el área de las ingenierías y en las ciencias exactas es donde se presentan mayores índices de reprobación, y principalmente en las materias relacionadas con matemáticas y programación, esto es posible comprobarlo en la siguiente tabla que representa los porcentajes de las materias de altos índices de reprobación en la Facultad de Estudios Superiores Aragón, UNAM. (FES Aragón U. , 2011).

En la Tabla 1 que se muestra abajo (FES Aragón U. , 2011) se observan cuáles de las materias que son compartidas por las ingenierías y presentan estos altos índices de reprobación.

Se pueden apreciar que algunas de las materias que presentan altos índices de reprobación son las que pertenecen al área de las matemáticas, entre estas materias se encuentran algunas como: cálculo vectorial, probabilidad y estadística, geometría analítica.

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO								
FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN								

CUADRO COMPARATIVO DE MATERIAS CON ALTOS ÍNDICES DE REPROBACIÓN

EXAMEN ORDINARIO

MATERIAS COMPARTIDAS POR LAS INGENIERÍAS PLAN 1992

<i>MATERIAS CON ALTOS ÍNDICES DE REPROBACIÓN</i>						<i>% DE REPROBACIÓN</i>		
<i>CLAVE</i>	<i>NOMBRE DE LA ASIGNATURA</i>	<i>2008-1</i>	<i>2008-2</i>	<i>2009-1</i>	<i>2009-2</i>	<i>2010-1</i>	<i>2010-2</i>	<i>2011-1</i>
0063	CÁLCULO VECTORIAL	"-"	"-"	"-"	"-"	"-"	53%	56%
0071	ELECTRICIDAD Y MAGNETISMO	"-"	"-"	51%	"-"	51%	"-"	"-"
0642	PLANEACIÓN	"-"	"-"	"-"	"-"	"-"	64%	"-"
0712	PROBABILIDAD Y ESTADÍSTICA	"-"	"-"	"-"	"-"	56%	"-"	63%
0994	OPTATIVA DE HUMANIDADES	"-"	"-"	51%	54%	"-"	"-"	"-"
1108	GEOMETRÍA ANALÍTICA	"-"	"-"	"-"	"-"	52%	"-"	"-"
1115	COMPUTADORAS Y PROGRAMACIÓN	52%	"-"	"-"	"-"	"-"	"-"	82%
1301	DINÁMICA	"-"	"-"	"-"	"-"	"-"	"-"	67%
1201	CINEMÁTICA	"-"	"-"	"-"	"-"	"-"	"-"	"-"
1204	ALGEBRA LINEAL	"-"	53%	"-"	"-"	"-"	"-"	"-"
1206	ESTÁTICA	"-"	53%	"-"	"-"	"-"	"-"	"-"
1302	ECUACIONES DIFERENCIALES	57%	"-"	"-"	60%	"-"	"-"	"-"
1400	ANÁLISIS DE SISTEMAS Y SEÑALES	"-"	51%	"-"	"-"	"-"	"-"	"-"
2135	ROBÓTICA	"-"	"-"	"-"	"-"	100%	"-"	"-"

Tabla 1. Cuadro comparativo de materias con alto índice de reprobación compartidas por las ingenierías (FES Aragón U. , 2011)

En esta Tabla 1 se observan las materias compartidas por las ingenierías y que tienen altos índices de reprobación, cabe resaltar la materia de **computadoras y programación**, que es una de las materias que se imparten en los primeros

semestres de la carrera de INGENIERÍA EN COMPUTACIÓN de la FES Aragón del plan 2008 y es una de las materias que se consideran más importantes, ya que los conocimientos adquiridos en esta materia se utilizan en los semestres posteriores para la desarrollo de programas de computadora. Además esta materia es de gran impacto en los egresados de la carrera de ingeniería en computación, ya que muchos de los empleos les solicitan habilidades en programación.

En la siguiente Tabla 2 se observan cada uno de los periodos que comprenden desde el año del 2008 y en cada uno de estos periodos se observan los porcentajes de reprobación de cada materia.

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN, FES Aragón

CUADRO COMPARATIVO DE MATERIAS CON ALTOS ÍNDICES DE REPROBACIÓN

EXAMEN ORDINARIO

INGENIERÍA EN COMPUTACIÓN

MATERIAS CON ALTOS ÍNDICES DE REPROBACIÓN						% DE REPROBACIÓN		
CLAVE	NOMBRE DE LA ASIGNATURA	2008-1	2008-2	2009-1	2009-2	2010-1	2010-2	2011-1
0043	BIOINGENIERÍA OPTATIVA	62%	"_"	"_"	"_"	"_"	"_"	"_"
0075	PROGRAMACIÓN ESTRUCTURADA Y CARACTERÍSTICAS DEL LENGUAJE	"_"	"_"	"_"	57%	67%	"_"	"_"
0401	INGENIERÍA DE SISTEMAS	"_"	"_"	"_"		90%	"_"	55%
0442	LENGUAJES FORMALES AUTÓMATAS	"_"		"_"	57%	"_"	"_"	"_"
0559	MEMORIAS Y PERIFERICOS	"_"	57%	"_"		"_"	56%	"_"
1203	PROGRAMACIÓN ORIENTADA A OBJETOS	"_"	54%	"_"	52%	"_"		"_"
1303	ECUACIONES DIFERENCIALES	"_"		"_"	"_"	"_"	55%	"_"
1401	ESTRUCTURAS DISCRETAS	"_"		"_"	"_"	63%	"_"	"_"
2138	SISTEMAS EXPERTOS	"_"	73%	"_"	"_"	60%	"_"	"_"

Tabla 2. Cuadro comparativo de las materias con alto índice de reprobación de la FES Aragón. (FES Aragón U. , 2011)

En la Tabla 2 se visualizan los porcentajes de reprobación cuando se presentan los exámenes ordinarios. Se revela que la materia de **programación orientada a objetos** presenta índices de reprobación superiores al 50%. Esta es una justificación para atender la problemática referida obligado poner énfasis en esta materia ya que es sobre la que hace referencia en este trabajo.

Cabe mencionar que los altos índices de reprobación son visibles particularmente en la materia de POO.

Por las razones anteriores se puede deducir que se necesitan medios y recursos que faciliten la integración de nuevas tecnologías en las aulas de clase (Serrano, 2009), esto porque la tecnología seguirá avanzando y continuará proporcionando elementos para mejorar el cómo se adquieren conocimientos en las aulas ya sea que la persona asista físicamente o de manera on-line. Además es necesario mencionar que se necesitan los procesos y documentación necesaria para poder implantarlas, ya que muchas veces la falta de esta documentación es objeto de que no se obtengan los resultados deseados cuando se adopta una de estas tecnologías.

Para poder atacar estos problemas y poder disminuir los índices de reprobación y mejorar los indicadores académicos en el egreso de alumnos, es necesario tener herramientas y procesos que permitan al profesor (Aviles, 2006) impartir su clase de una manera más sencilla, de manera controlada esto usando herramientas que hacen uso de los conceptos de eLearning y bLearning. Estas herramientas deben permitir tener una mayor interacción y control sobre los alumnos, deben ser muy intuitivas en su interfaz y utilizar tecnologías muy usadas por los jóvenes actualmente tales como redes sociales y el internet.

Es aquí donde las plataformas educativas virtuales apoyan para atacar estos problemas, ya que estas dan muchas facilidades para poder tomar una clase en cualquier hora y cualquier lugar, llevar un control sobre los alumnos, proporcionar a los estudiantes temas que no se verán en las clases presenciales, además de fomentar la colaboración entre el profesor y el alumno, entre otros beneficios.

En este trabajo se considera que la creación de los recursos didácticos que apoyan al proceso de enseñanza – aprendizaje en la carrera de Ing. en Computación que se imparte en la FES Aragón, y consta de la creación de recursos didácticos y la documentación de los procesos para implementar estos recursos.

Con este trabajo se benefician tanto los profesores, los alumnos como la institución:

- a) Al alumno, porque le proporciona documentos adicionales para complementar los temas vistos en clase, se le dan herramientas que le facilitan la comunicación con el profesor además de que puede dar continuidad en el aprendizaje de ciertos temas vistos en la materia y porque desarrolla habilidades en TIC con el uso de estas herramientas on-line.

- b) Al profesor, porque le facilita la enseñanza de la materia de POO en las carreras relacionadas con el desarrollo de sistemas, se le facilita la interacción con el alumno a través de herramientas como el chat y los foros, puede contar con información importante como cuantas horas paso el alumno en el sitio virtual del curso realizando ejercicios, cual ha sido el grado de avance sobre el curso, que nivel de conocimientos sobre la materia ha adquirido, que conceptos le cuestan más trabajo entender; además le es posible proporcionar al alumno documentos y artículos adicionales de temas que no se ven en clase.
- c) A la institución, porque sigue las indicaciones para los organismos acreditadores⁵ que establecen no usar no recursos tradicionales y porque cumple con uno de los objetivos de estos organismos (CONAIC⁶), que es promover y contribuir al mejoramiento de la calidad en la formación de los profesionales de la informática y la computación⁷.

El material didáctico creado se almacena en los repositorios de la plataforma educativa virtual Chamilo instalada en el equipo de cómputo que funge como servidor de ingeniería de la FES Aragón, de esta manera los alumnos y profesores podrán acceder a estos recursos educativos desde cualquier sitio y a cualquier hora y realizar los ejercicios propuestos.

Durante la realización de este trabajo se siguió un procedimiento de 4 etapas para poder obtener el material didáctico educativo, este material se almaceno en un repositorio virtual la plataforma educativa virtual Chamilo, las 4 etapas que se siguieron son las mostradas en la Figura 2:

⁵ Acreditación de la carrera Ingeniería en Computación, Facultad de Estudios Superiores (FES) Aragón, UNAM <http://noticias.universia.net.mx/ciencia-nn-tt/noticia/2008/08/28/20081/certificadas-doce-licenciaturas-imparten-fes-aragon-unam.html>, consultada Junio 2011

⁶ CONSEJO NACIONAL DE ACREDITACIÓN EN INFORMATICA Y COMPUTACIÓN A.C <http://www.conaic.net/>.

⁷ Certificados y Acreditaciones, FES Aragón, <http://ingenieria.aragon.unam.mx/nosotros/certificados-y-acreditaciones.php>, consultada Junio 2011

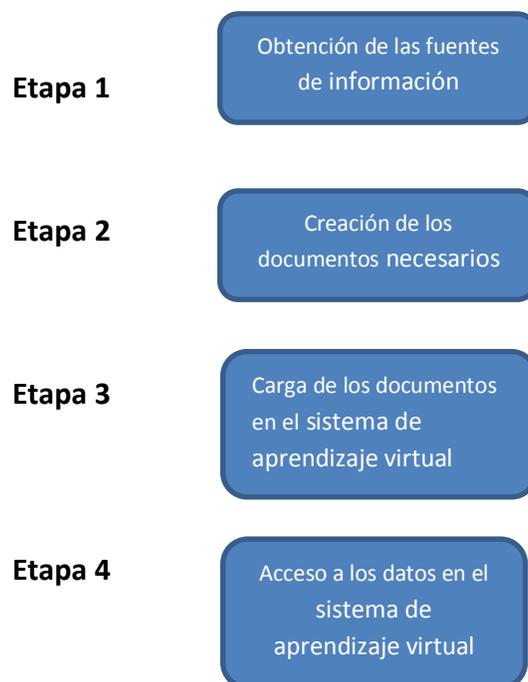


Figura 2. Etapas del desarrollo de los contenidos

Los detalles acerca de cómo se realizaron cada una de estas etapas anteriores es abordada en la sección de análisis, diseño e implementación. Cada una de las etapas anteriores lleva un largo proceso que inicia desde que obtienen las fuentes de información de acuerdo al material que se va a crear, hasta la etapa final en la que se tiene la plataforma educativa ya con el material incluido y lista para que sea usada por los alumnos.



CAPITULO 2



2. OBJETIVOS

En este capítulo se presentan los objetivos de este caso práctico, los sub objetivos que se tienen que ir realizando para lograr el objetivo principal y el proceso que se siguió para llegar al objetivo principal.

2.1. OBJETIVO PRINCIPAL.

Implementar material didáctico para la materia de Programación Orientada a Objetos, con el propósito de apoyar el proceso de enseñanza – aprendizaje tanto a profesores como alumnos de ingeniería en computación, mediante la aplicación de las Tecnologías de Información y Comunicación (TIC).

Para lograr el arriba mencionado objetivo es necesario contar con los cursos de acción específicos, llamados subjetivos que permiten lograr el objetivo principal.

En este caso para poder lograr el objetivo principal se tienen que realizar varias etapas de trabajo, ya que para implementar este material se tiene que realizar investigación sobre el tema que es la POO, analizar los procesos de enseñanza – aprendizaje que se utilizan, diseñar y analizar las tecnologías que se van a ocupar para poner los recursos didácticos a disposición.

2.2. SUB OBJETIVOS.

Analizar el papel de las TIC y en específico de las plataformas virtuales educativas para entender la funcionalidad y utilidad que tienen hoy en día en el ámbito educativo.

Determinar la estructura conceptual de la materia Programación Orientada a Objetos y orden de los temas que se van a abordar en esta.

Diseñar los módulos necesarios en la plataforma educativa Chamilo para implementar el material didáctico de la clase POO.

2.3. ESTRATEGÍA PARA ALCANZAR LOS OBJETIVOS

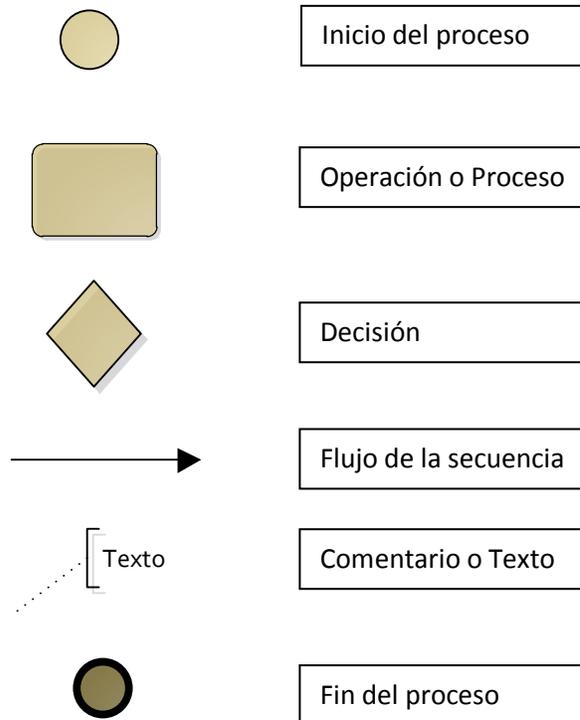
El proceso para alcanzar los objetivos planteados se lleva en varios pasos ya que primero se tienen que tomar en cuenta cumplir los sub objetivos para finalmente alcanzar el objetivo principal de este trabajo. Para esto se decidió incluir una sección en la que se muestra la estrategia que se siguió para alcanzar el objetivo principal.

A continuación se presentan la representación gráfica o algoritmos del proceso que permite alcanzar los sub-objetivos y finalmente el del objetivo principal.

Para realizar el esquema ha sido necesario considerar la planificación previa, con respecto a la definición del objetivo y el establecimiento del resultado de dicho proceso.

Así mismo, se ha buscado expresar claramente el proceso real de la presente investigación.

Simbología Empleada:



Sub - Objetivo 1: *Analizar el papel de las TIC y en específico de las plataformas virtuales educativas para entender la funcionalidad y utilidad que tienen hoy en día en el ámbito educativo*

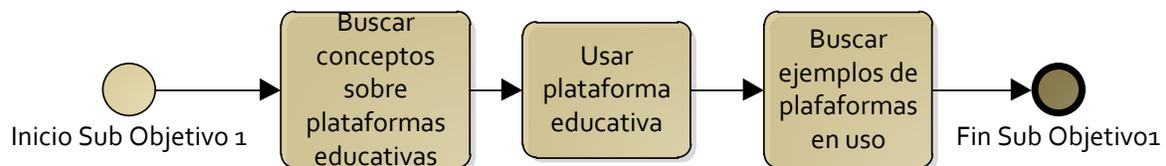


Figura 3. Sub – Objetivo 1

Sub - Objetivo 2: *Determinar la estructura conceptual de la materia Programación Orientada a Objetos y orden de los temas que se van a abordar en esta.*

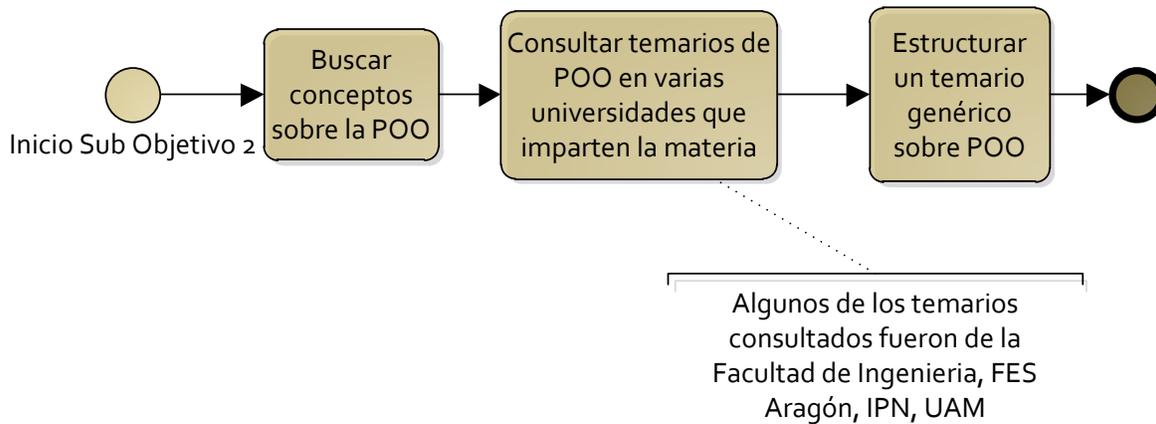


Figura 4. Sub - Objetivo 2

Sub - Objetivo 3: *Diseñar los módulos necesarios en la plataforma educativa Chamilo para implementar el material didáctico de la clase POO.*

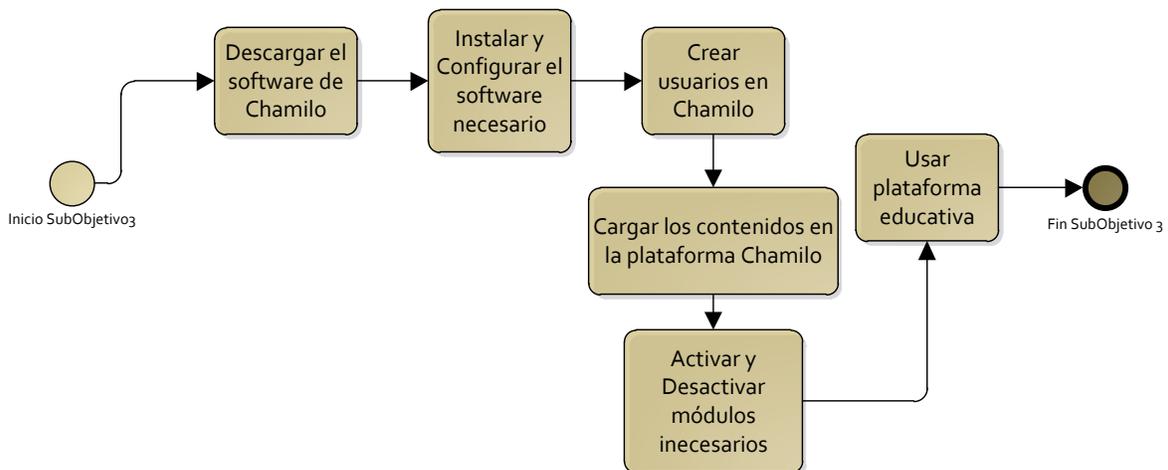


Figura 5. Sub - Objetivo 3

Objetivo Principal: *Implementar material didáctico para la materia de Programación Orientada a Objetos, con el propósito de apoyar el proceso de enseñanza – aprendizaje tanto a profesores como alumnos de ingeniería en computación, mediante la aplicación de las Tecnologías de Información y Comunicación (TIC).*

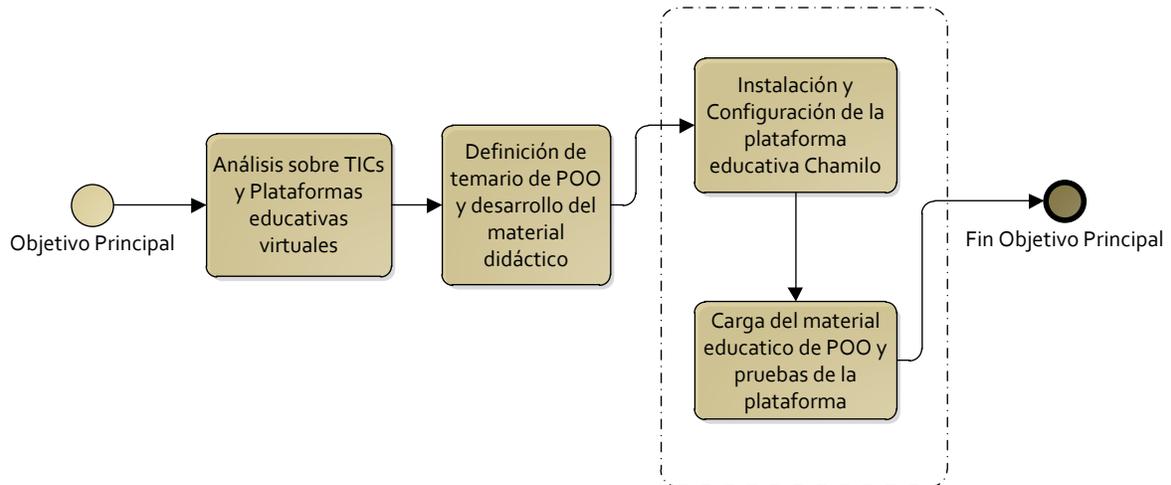
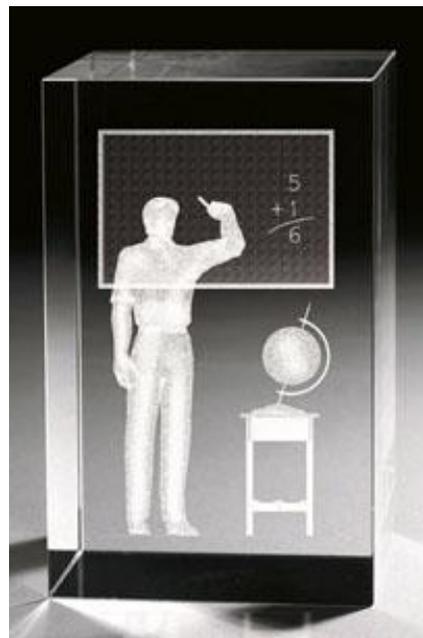


Figura 6. Objetivo Principal

Los diagramas que se mostrados en la Figura 3, Figura 4, Figura 5 y Figura 6; ilustran los pasos que se tuvieron que seguir para cumplir primero los objetivos particulares (sub – objetivos) y al final cumplir el objetivo principal. Al cumplir este objetivo se obtiene como resultado el material educativo de POO cargado en la plataforma educativa virtual Chamilo.



CAPITULO 3



3. MARCO TEÓRICO DE REFERENCIA

En este capítulo se aborda de manera exploratoria un resumen de la teoría necesaria para poder desarrollar el material didáctico que se propone en este trabajo, así como los conceptos que son las plataformas educativas virtuales y las definiciones de lo que es e-Learning y b-Learning.

3.1. PARADIGMAS Y LENGUAJES DE PROGRAMACIÓN.

Desde los principios de la humanidad el hombre ha tenido la necesidad de comunicarse con otros hombres u animales, esta necesidad lleva como objetivo transmitir información hacia otra persona, esta pueden ser sentimientos, emociones, ideas, etc. Para poder lograr esta transmisión de información el hombre invento su propio medio de comunicación, el lenguaje, que se puede definir como la facultad del hombre para comunicarse con otro hombre. Esta idea de cómo se concibe el lenguaje es un debate filosófico y de investigación en la actualidad, es por eso que se debe considerar un lenguaje como un medio de comunicación para facilitar la comprensión del tema.

A continuación se muestra la Figura 7 donde se observan los elementos que forman parte de la comunicación, entre una de ellas se encuentra el lenguaje.



Figura 7. Circuito de la comunicación.

Del concepto de lenguaje y de su uso para la comunicación entre seres vivos, nace la idea de darle instrucciones a una máquina a través de un lenguaje, esto no quiere decir que se esté comunicando con la máquina sino que se usó el concepto de lenguaje para representar tareas que se van a dar a una máquina. Algunas de las características de un lenguaje para computadoras son las siguientes: que el lenguaje sea suficientemente robusto y flexible, que permita indicar con claridad instrucciones, que permita construir instrucciones más complejas a partir de instrucciones más simples. Esto ha permitido que a través de un lenguaje de programación un programador de computadoras pueda ordenar a la máquina realizar una o varias tareas a través de programas.

En la historia de los lenguajes, es necesario mencionar que de los primeros lenguajes que usaban las computadoras fue el lenguaje máquina,⁸ y una de las primeras formas para dar instrucciones a una computadora era a través de interruptores que se podían encender o apagar, representando con esto una instrucción en la que podían utilizarse sólo 2 dígitos. Después de este lenguaje nace el lenguaje ensamblador, un lenguaje complicado pero que facilitaba la programación y finalmente vienen los lenguajes de alto nivel y lenguajes visuales que facilitan aún más la programación de computadoras.

Hay que tener en cuenta que antes se tenían lenguajes muy complicados que trabajaban a nivel de microprocesador como el lenguaje máquina y ensamblador y a partir de estos se generaron los conceptos y la teoría necesaria para hoy en día tener lenguajes muy sofisticados y fáciles de aprender, estos lenguajes tienen la característica de que permiten la programación de aplicaciones muy complejas en poco tiempo y facilitan la reusabilidad.

Para que la evolución de los lenguajes se llevara a cabo tuvo que existir un elemento que permitió la evolución del lenguaje máquina al ensamblador, del ensamblador al lenguaje de alto nivel y de este último a los lenguajes visuales que existen hoy en día. Este elemento clave se le denomina Compilador y su función es traducir de un lenguaje a otro, en la literatura se toman como referencia los lenguajes de programación de alto nivel y ensamblador por cuestiones didácticas, pero la función de un compilador no sólo es traducir de un lenguaje de alto nivel a uno de más bajo nivel, sino que a veces surge la necesidad de traducir de un lenguaje a otro, sin importar el nivel en el que se encuentran.

En la Figura 8 se muestran las fases de las que consta un compilador, en la parte superior se observa el código fuente en un lenguaje como la entrada para iniciar este proceso. Como se menciona anteriormente en el proceso de compilación no tiene que ser necesariamente un lenguaje de alto nivel el que inicia el proceso, si no que puede ser otro lenguaje de cualquier nivel. Durante este proceso de compilación existen otras 3 etapas que son el análisis léxico, análisis semántico y la generación de un código intermedio, al final de todas las etapas se genera el código máquina mejor conocido como ejecutable, que corresponde a las instrucciones necesarias para que el procesador realice la tarea deseada.

⁸ El lenguaje máquina es aquel que solo toma como símbolos el 1 y 0 en todo su alfabeto, es decir, un alfabeto de solo dos símbolos.

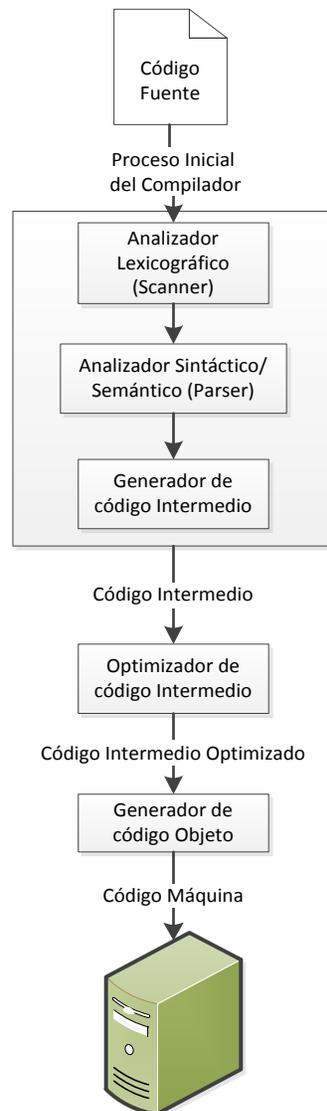


Figura 8. Fases que componen la estructura de un compilador.

3.1.1. LOS LENGUAJES DE PROGRAMACIÓN

Los lenguajes de programación tienen características complejas y existen una gran variedad, que van desde lenguajes visuales que permiten crear aplicaciones muy rápidamente, hasta los más complejos en los que se necesita tener experiencia programando y una habilidad para la abstracción de conceptos complejos, también cada lenguaje proporciona ciertas facilidades para resolver problemas, estas son conocidas como paradigmas de programación.

Anteriormente ya se mencionaba que el más viejo ancestro de los lenguajes de programación de computadoras es el lenguaje de circuitos y switches, el cual

consistía en estar conectando y desconectando cables, activando y desactivando interruptores, esto para lograr hacer que la computadora realizara alguna tarea. Esta manera de programar era demasiado compleja para las personas que operaban computadoras en ese entonces, es difícil imaginar que se llevara horas y un gran esfuerzo de los operadores realizar una suma de números, es por eso que surge la necesidad de tener un lenguaje que facilite esto en cuestiones de tiempo y esfuerzo la programación.

Posteriormente a este lenguaje tan complejo surge el lenguaje máquina, este lenguaje que se caracterizaba por utilizar el sistema de numeración binaria (1's y 0's). Con el uso de este lenguaje surgieron instrucciones que permitían sumar, restar. A pesar de tener instrucciones para hacer operaciones matemáticas simples, las instrucciones para hacer estas operaciones seguían siendo demasiado complejas, por ejemplo, una operación como la suma podría estar representada por la siguiente sucesión de dígitos '10001011', lo anterior era muy difícil para ser recordado por los programadores de esos entonces y era demasiado complicado depurar un programa, por lo que vuelve a surgir la necesidad de un lenguaje más fácil de recordar, entender, depurar y que permitiera realizar los programas más rápido .

Ante la necesidad de un lenguaje que facilite la programación en comparación con el lenguaje máquina nace el lenguaje ensamblador, un lenguaje que permite el uso de nemotécnicos para poder comprender y recordar más fácil las instrucciones. Las instrucciones principales eran para restar, alterar banderas, copiar y extraer datos de registros del procesador. Una de las grandes ventajas de utilizar el lenguaje ensamblador era que permitía hacer programas para computadora fáciles de entender y las instrucciones eran fáciles de recordar, además de que los programas ocupaban poco espacio en memoria.

En la siguiente Figura 9 se ilustra las relaciones entre los lenguajes máquina, ensamblador y de alto nivel; cada circulo significa una agrupación, por ejemplo en la figura se puede observar que una instrucción en ensamblador puede agrupar varias instrucciones binarias.

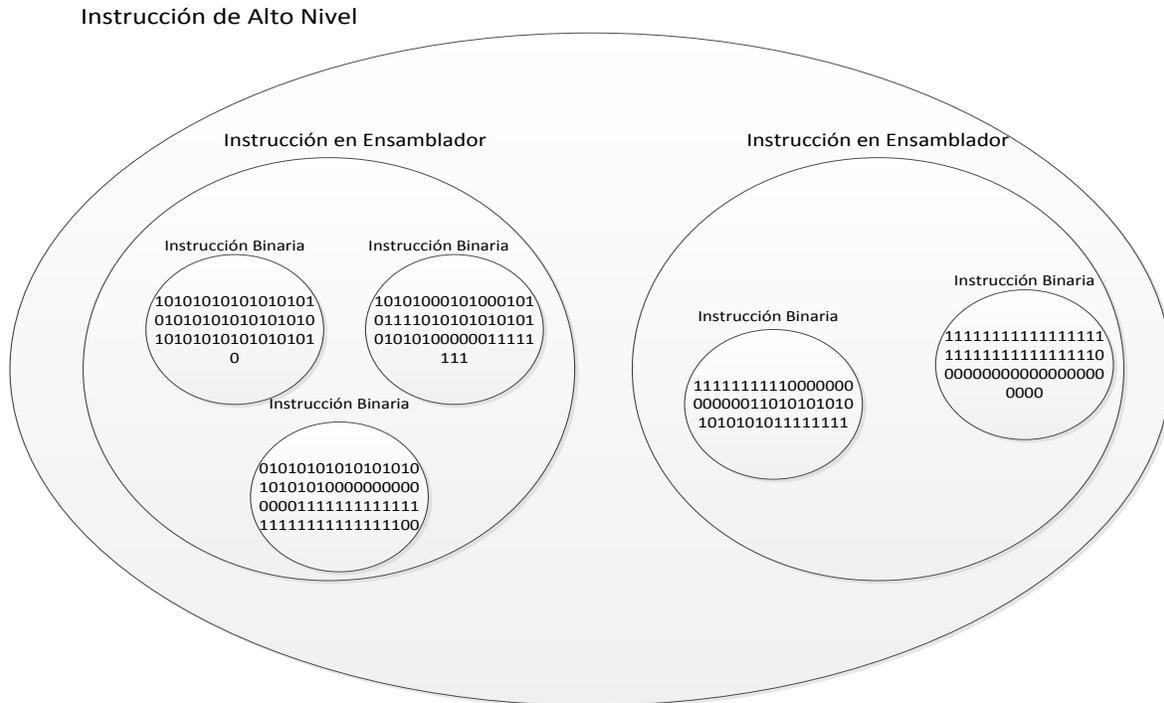


Figura 9. Relación entre las instrucciones de alto nivel, ensamblador y máquina.⁹

Es necesario mencionar que para lograr interpretar las instrucciones en lenguaje ensamblador se necesitaba un programa llamado ensamblador que toma el código en lenguaje ensamblador y genera código en lenguaje máquina específico para nuestro procesador¹⁰.

En la siguiente Figura 10 se ilustra la función de un ensamblador, toma el código en lenguaje ensamblador (lado izquierdo) y genera el lenguaje correspondiente al procesador específico de la máquina (lado derecho).



Figura 10. Proceso de conversión entre lenguajes que realiza un ensamblador.

⁹ http://www.carlospes.com/minidiccionario/lenguaje_de_alto_nivel.php, consultado en Diciembre 2011

¹⁰ Ver figura 4 y figura 6.

Otra característica del lenguaje ensamblador es particularidad a cada máquina, es decir, que dependía del procesador que lo iba a ejecutar, esto debido a que a la hora de programar era necesario saber cuántos y cuáles registros tenía ese procesador, se tenía que conocer las banderas, que arquitectura era RISC o CISC, etc., por lo tanto el programador tenía un conocimiento basto sobre la máquina en la que se iba a programar, esto ocasionaba que el código que generaba el programador en ensamblador no fuera portable a otros equipos y exigía que estos fueran expertos en la arquitectura del procesador del equipo.

Es importante notar el gran salto que se da del lenguaje máquina al lenguaje ensamblador, ya que da una gran facilidad para programar y permite que los programas que antes ocupaban miles de líneas de código se reduzcan considerablemente gracias a los nemotécnicos ocupados por el ensamblador, además las instrucciones eran más familiares para el programador en lenguaje ensamblador ya que eran abreviaciones similares al idioma inglés. Al igual que los lenguajes antecesores al ensamblador este tenía muchos problemas debidos a que si los programas eran muy grandes y realizaban varias tareas complejas la programación en ensamblador se volvía una tortura, además de que el programador debería conocer el procesador de la máquina a detalle. Para resolver estos problemas surgen los lenguajes de alto nivel¹¹.

Un lenguaje de alto nivel permite tener una abstracción mayor del problema que se plantea, además de que permite generar código portable, es decir, el código generado a través de un lenguaje de alto nivel puede ser utilizado en otras computadoras, independientemente de sus características físicas (tales como el procesador y la memoria), esta y otras ventajas dieron preferencia al uso de un lenguaje de alto nivel sobre un lenguaje ensamblador.

Los lenguajes de programación de alto y medio nivel vinieron a resolver muchos problemas que se tenían en el sector de la programación. Fue en el año de 1960 cuando el lenguaje Fortran surge y de este año en adelante surgen más lenguajes de programación como C, C++, Java, entre otros. Las principales características que proporcionan estos lenguajes de programación son que permiten hacer programas más complejos, que facilitan la depuración de programas, son más fáciles de aprender. Actualmente estos en combinación con los lenguajes visuales son los más utilizados por los programadores.

¹¹Al hablar del nivel de un lenguaje es referirse al grado que se encuentra del lenguaje humano, es decir, entre más alto nivel tenga más cerca estará del lenguaje de los humanos

Finalmente llegaron los lenguajes visuales que permiten a través del mouse y con la tecnología *drag and drop*¹², crear aplicaciones en cuestiones de minutos. En este nuevo tipo de lenguajes de programación se incorporan los *designers* que son aplicaciones para realizar prototipos de programas y la principal característica es que permiten arrastrar botones y otros *widgets*. Los lenguajes visuales vinieron a agilizar la programación de aplicaciones y permitieron que gente con pocos conocimientos en programación lograran hacer programas muy complejos en poco tiempo.

Ya se dio un panorama general sobre los lenguajes de programación, desde lenguajes tan complejos como el lenguaje máquina hasta lenguajes con los que podemos crear una GUI (siglas de Graphical User Interface) en cuestión de minutos. Pero es necesario no olvidarnos que la principal función de un lenguaje de programación de cualquier nivel es resolver un problema.

Para poder resolver un problema a través de un lenguaje de programación es necesario conocer las capacidades de este lenguaje, es decir, antes de escoger el lenguaje con el cual vamos a afrontar nuestro problema es necesario saber el paradigma del lenguaje se ocupara, ya que de acuerdo al paradigma proporcionará las facilidades para resolver ciertos problemas. Existen varios paradigmas de programación:

- Programación estructurada o declarativa
- Programación orientada a objetos
- Programación orientada a componentes
- Programación orientada a aspectos

Además de las mencionadas anteriormente existen otros paradigmas, explicare brevemente las 3 de los ya mencionados, el otro paradigma faltante (Programación Orientada a Objetos) tiene una sección dedicada en este trabajo, es por eso que no abundare demasiado en este tema.

3.1.2. PROGRAMACIÓN ESTRUCTURADA

La programación estructurada representada principalmente por lenguajes como C y BASIC¹³ es el tipo de programación que se empezó a usar cuando surgieron los lenguajes de alto nivel, en este tipo de programación encontramos que la unidad principal es la función, la cual está compuesta por un tipo de retorno, nombre de función, y los parámetros que recibe:

¹² La tecnología *drag and drop* (arrastrar y soltar), permite que a través de mouse se puedan arrastrar objetos como botones, imágenes, iconos, etc.; y soltarlos en una zona comúnmente conocida como *canvas*, que es donde se está armando la interfaz de usuario.

¹³ **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode

Tipo_retorno nombreFunc(parametro1, parametro2, ...) -> int suma(a,b)

La programación estructurada también tiene la característica que se pueden usar estructuras de control¹⁴ como: *if, else*; estructuras de repetición como: *for(; ;), while(cond), do while(cond)* o la famosa instrucción *goto*, esta última fue uno de los grandes problemas en la programación no estructurada, ya que esta instrucción creaba código espagueti, es decir, que existía código en el cual se podía saltar de cualquier punto del programa a otro.

3.1.3. PROGRAMACIÓN ORIENTADA A COMPONENTES

Dentro de los paradigmas de programación surgió la programación orientada a componentes, en este paradigma de programación el punto principal es el componente, estos componentes pueden ser tan complejos y realizar tareas muy complejas. Esto permite un mayor reuso de software y flexibilidad a la hora de estar programando, ya que si se necesitara cierta funcionalidad y es necesario tener un componente que realiza esta funcionalidad podemos utilizarlo y cambiarlo con un componente similar siempre y cuando se tenga la misma interfaz de comunicación. Un componente se define como:

Un componente de software es una pieza de código auto contenido, también tiene la capacidad de auto mostrarse, tiene una funcionalidad bien definida y se puede ensamblar con otros componentes a través de su interfaz (Andy Ju An Wang, 2005).

La programación orientada a componentes (o COP por sus siglas en inglés) sería equivalente al concepto de CBSE (Component Based Software Engineering), sólo que esta última es más general e incluye a la COP.

Dentro del código podemos incluir a un componente de 3 maneras:

Caja Blanca. Podríamos incluir al componente directamente en nuestro código pegando su código, y utilizarlo en el nuestro incluyendo el código de este, pero este enfoque tiene la desventaja de que se tiene que recompilar el componente junto con nuestro código, además de que se tiene acceso a la implementación del componente y esto viola los principios de encapsulamiento.

Caja Negra. En este enfoque no se incluye la implementación del componente, sólo se muestra la interfaz de este a todos los clientes (en C/C++ esta interfaz normalmente se encuentra definida en ficheros .h a través de prototipos de funciones), normalmente la implementación se encuentra en librerías dinámicas,

¹⁴ Los lenguajes de programación orientados a objetos también usan estas estructuras de control, la mayoría tomaron estas ideas de lenguajes como Fortran y C.

estas varían de acuerdo al S.O., dll's para Windows o librerías .so para Linux y .dylib para Mac. También se puede encontrar la implementación en librerías estáticas que se incluyen cuando se enlaza el programa estas librerías tienen la extensión .lib para Windows y .a para Linux y Mac, para estas últimas es necesario mencionar que se incluyen en el código en tiempo de enlazado a diferencia de las dinámicas que se cargan en tiempo de ejecución.

Caja Gris. Es una mezcla de los componentes caja negra y blanca, ya que no se oculta totalmente la implementación pero tampoco se muestra completamente.

La Figura 11 se ilustra una propiedad fundamental en los componentes, ya que se observa que el componente A encapsula las clases A y B y de esta manera los clientes A y B pueden acceder a estas a través de la interfaz que proporciona A, esto no se logra en la POO, al contrario de esto, si los clientes quisieran lograr esto tendrían que duplicar código.

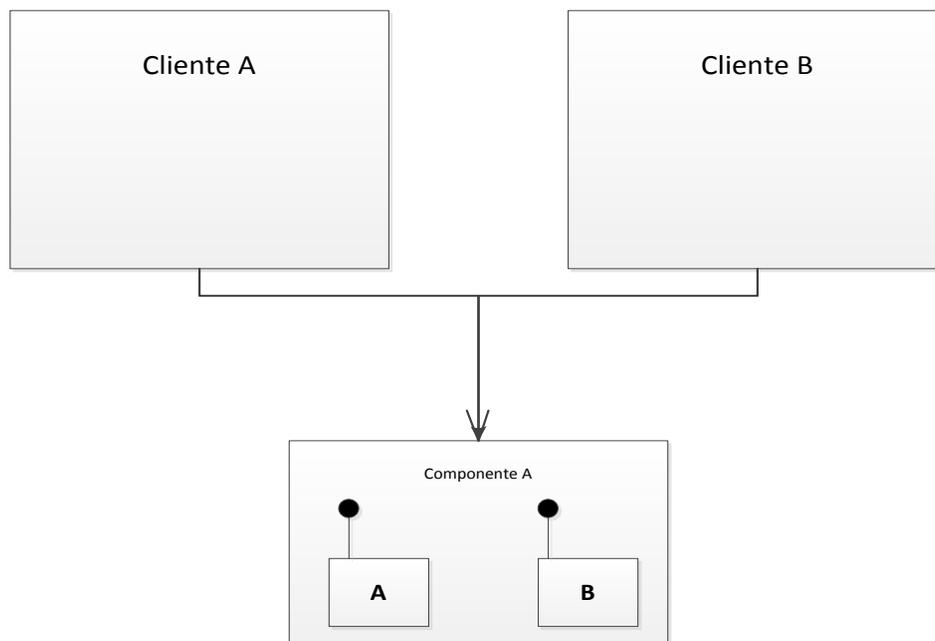


Figura 11. Encapsulación de clases reusables en un componente.¹⁵

3.1.4. PROGRAMACIÓN ORIENTADA A ASPECTOS

Actualmente la programación orientada a aspectos se encuentra en una fase de desarrollo y de pruebas. Podemos observar claramente que la mayoría de software que se desarrolla para la industria es utilizando el paradigma de Programación Orientada a Objetos.

¹⁵ <http://www.c-sharpcorner.com/UploadFile/poolamu/BeginnersGuideWCF04152009031247AM/BeginnersGuideWCF.aspx>, consultada en abril de 2011

Con el uso de la Programación Orientada a Aspectos (AOP por sus siglas en inglés) se pueden solucionar problemas que se tienen cuando se usa la programación orientada a objetos, uno de los problemas que se tienen con la POO es cuando se tiene que agregar funcionalidad a una clase. La funcionalidad que se quiere agregar a una clase puede ser en cuestiones de sincronización o de seguridad para esto se tienen que modificar todas las clases. En este punto es donde la AOP juega un papel importante ya que permite agregar aspectos a una clase, estos aspectos son implementados en otra clase y evita el hecho de tener que modificar la clase.

Los lenguajes orientados a aspectos definen una nueva unidad de programación de software para encapsular las funcionalidades que cruzan todo el código, es decir, que antes que nada se tiene que identificar todos los aspectos del código para su posterior implementación

Para entender bien lo que es un aspecto es necesario conocer su definición. Un aspecto se define según Gregor Kiczales como:

“Un aspecto es una unidad modular que se dispersa por la estructura de otras unidades funcionales. Los aspectos existen tanto en la etapa de diseño como en la de implementación. Un aspecto de diseño es una unidad modular del diseño que se entremezcla en la estructura de otras partes del diseño. Un aspecto de programa o de código es una unidad modular del programa que aparece en otras unidades modulares del programa”.

3.2. LA METODOLOGÍA DE PROGRAMACIÓN ORIENTADA A OBJETOS.

La programación orientada a objetos (POO), surge como una necesidad para poder crear software reutilizable y flexible, en la POO existen cuatro mecanismos básicos que son clave en las capacidades y características que proporciona este paradigma, estos son:

- Abstracción
- Herencia
- Encapsulamiento
- Polimorfismo

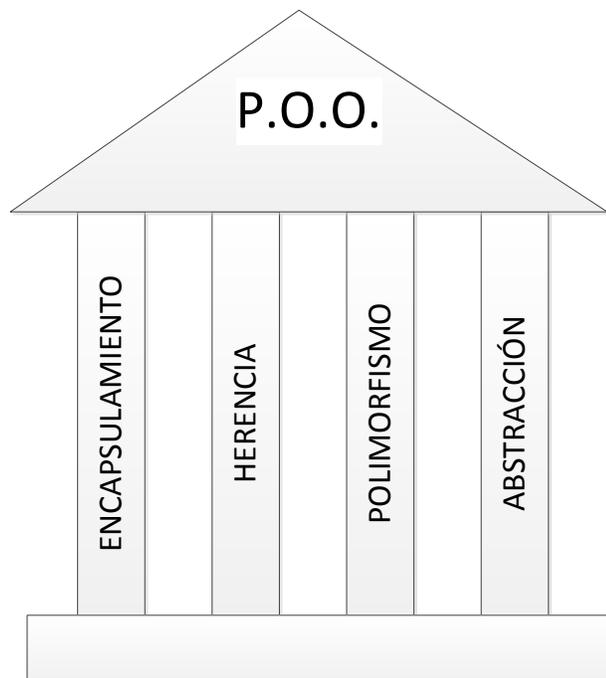


Figura 12. Pilares de la Programación Orientada a Objetos.

En la POO, el contenedor o el molde de los objetos son las clases, en estas se especifican los métodos y las variables miembro. A partir de una clase podemos crear objetos y dentro de estos objetos podemos tener como datos miembro u otros objetos mucho más complejos. En la POO también surgieron otros mecanismos como herencia, polimorfismo, sobre carga de operadores; estos dieron facilidades para crear grandes sistemas dando flexibilidad, reutilización y facilidad a la hora de hacer cambios en la arquitectura del sistema¹⁶.

Las principales ventajas que dio la POO en comparación con la programación estructurada es que la POO facilita la reutilización de código y proporciona una gran flexibilidad a la hora de realizar cambios en la programación. La reutilización se logra cuando escribimos código o bibliotecas que pueden ser utilizadas para crear programas muy complejos, estas bibliotecas poseen un diseño y una arquitectura que permite su reutilización, por ejemplo, actualmente existen bibliotecas gráficas que permiten utilizar sus clases para crear objetos, esto sin la necesidad de conocer cómo se implementa la funcionalidad del objeto, ya que sólo basta con conocer la interfaz de comunicación para poder crear objetos de este tipo.

Como ya se mencionaba anteriormente, el concepto de clase y objeto son los puntos clave para entender lo que es la POO. Para entender el concepto de clase

¹⁶ Véase la figura 8

vamos a pensar que se tiene un carro, este carro tiene características como que es de cierto color, que tiene una marca, puede ser de 4 u 8 cilindros, etc; además contiene elementos como un volante, tiene ruedas, tiene faros, tiene radio, tiene asientos, etc; y también realiza acciones como prender motor, avanzar, retroceder, frenar, acelerar, etc. Todo esto puede ser representado en una clase, es decir, la clase puede contener todas las características, elementos y funciones que componen al carro o sólo puede contener algunas de estas, esto depende de la funcionalidad que se le quiere dar a la clase.

Una clase se podría ver como el molde para crear objetos, es decir, si quisiéramos crear un objeto de tipo carro tendríamos que crear una instancia de la clase carro, a esta instancia se le puede asignar un nombre y utilizarse dentro del código para hacer operaciones con este objeto, así podríamos tener instancias de objetos de tipo carro, tipo avión, tipo tanque y poder crear un videojuego de guerra con todos estos objetos.

En el lenguaje orientado a objetos C++ una clase se representa con la palabra reservada **class**, esto se ilustra en el siguiente fragmento de código:

```
class Carro
{
public:
    void prender( );
    void apagar( );
    void avanzar(36nti Metros);
    void retroceder(36nti Metros);
private:
    string color;
    string marca;
    string modelo;
};
```

Si quisiéramos crear una instancia de esta clase de carro, en lenguaje C++ tendríamos que agregar la siguiente línea dentro de nuestra función *main*.

```
Carro micarro1;
```

Ya se había mencionado que dentro de la programación orientada a objetos existen conceptos más complejos como herencia, polimorfismo o sobrecarga de operadores, estos conceptos son indispensables para comprender lo que es la POO.

La **abstracción** se refiere principalmente a identificar las características principales de un ente para obtener su representación, un ejemplo lo podemos ver en el anterior ejemplo del carro, en este identificamos sus principales características como el modelo, el color, la marca e identificamos sus principales

funcionalidades. Existen otras características que no son tan importantes para la representación de un carro, características como la marca del radio, la marca de las llantas, el color de los asientos, etc. Esto es lo que permite la abstracción obtener las características principales de elemento para poderlo identificar por lo que es¹⁷.

La **encapsulación** se refiere a la idea de tener restricciones o no al acceder a los datos de un objeto, estas restricciones se definen en una clase. Para lograr la encapsulación en un lenguaje orientado a objetos se disponen de palabras reservadas que deniegan el acceso o no a ciertos datos. En el lenguaje C++ estas palabras reservadas son **public**, **private** y **protected**. Estas palabras reservadas permiten que los clientes de la clase accedan o no a los datos, normalmente el acceso a los datos se hace a través de métodos que también pueden tener estos indicadores de acceso; con esto se logra una mayor confiabilidad en los datos a través de la encapsulación.

Para entender el mecanismo de la **herencia** es necesario hacer un símil con la vida real. En la vida real cuando nace una persona tiene algunas características como el color de los ojos, pelo y piel, estas pueden ser heredadas tanto del padre como de la madre a esto se le denomina herencia genética. En la POO se puede usar un mecanismo similar donde se puede heredar de una clase, se puede heredar de varias maneras e incluso se pueden sobrescribir los métodos de la clase padre siempre y cuando el tipo de herencia que se usó lo permita, a través de este mecanismo se heredan las propiedades y métodos de la clase padre, estos pueden ser utilizados dentro de la clase derivada (hija) para implementar cierta funcionalidad.

En la siguiente Figura 13 se muestra un diagrama de UML (*Unified Modeling Language*) que ilustra el concepto de herencia. En este se observa que las clases Ganso, Pato y Gallina; son clases hijas de la clase Ave y de esta manera estas clases adquieren sus propiedades.

¹⁷ La abstracción es un proceso mental que se aplica al seleccionar algunas de las características y propiedades de un conjunto de cosas del mundo real, excluyendo otras no pertinentes (<http://www.alegsa.com.ar/Dic/abstraccion.php>, consultada en Febrero 2011)

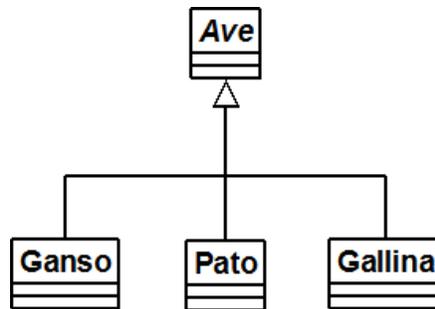


Figura 13. Diagrama UML en el que se ilustra el mecanismo de herencia en la POO.

El mecanismo de **polimorfismo** es uno de los mecanismos más difícil de entender ya que antes es necesario comprender la herencia. El polimorfismo se logra a través de la herencia ya que a través de la jerarquía clase base y clase derivada es posible hacer referencia a una clase derivada siempre y cuando tengamos un apuntador a la clase base. Por ejemplo, si creamos un apuntador a la clase Ave (véase la figura 10), entonces podríamos hacer que apuntara tanto a la clase Ganso, o a la clase Pato, o a la clase Gallina, esto debido a que tanto la clase Ganso, Pato y Gallina son derivadas de la clase Ave. Con este mecanismo de polimorfismo permite que una clase Base (clase Ave) se transforme en otra clase siempre y cuando esa clase sea una clase derivada, para que de esta manera se pueda cambiar de manera dinámica el objeto al que hace referencia un apuntador.

3.2.1. EL LENGUAJE DE MODELADO UML Y LA POO.

Para facilitar la comprensión de un sistema existen diferentes herramientas de modelado, una de las importantes y que es más usada en la actualidad en la ingeniería en software es UML. UML, que significa por sus siglas en inglés Unified Modeling Language, es un lenguaje que permite modelar sistemas de diferentes tamaños¹⁸, en este lenguaje podemos visualizar la funcionalidad de un sistema fácilmente ya que proporciona 13 diagramas que permiten obtener diferentes vistas del sistema de diferentes perspectivas, los 13 diagramas que componen la especificación 2.0 son los siguientes:

1. Diagrama de clases.
2. Diagrama de objetos.
3. Diagrama de componentes.
4. Diagrama de estructura compuesta.
5. Diagrama de casos de uso.
6. Diagrama de secuencia.

¹⁸ El tamaño de un sistema se refiere a la cantidad de código fuente que posee, así como sus componentes.

7. Diagrama de comunicación.
8. Diagrama de estados.
9. Diagrama de actividades.
10. Diagrama de despliegue.
11. Diagrama de paquetes.
12. Diagrama de tiempos.
13. Diagrama global de interacciones.

Cada uno de estos diagramas ofrece una diferente vista para cada uno de las personas que intervienen en el proceso de desarrollo de software, estos van desde los clientes que solicitan ciertos requerimientos para su aplicación hasta los desarrolladores y arquitectos del sistema que se encargan de programar y de planificar cuáles serán los módulos de que compondrá el sistema.

En UML encontramos 4 tipos de elementos estos son: estructurales, de comportamiento, de agrupación y de anotación. En los elementos estructurales se encuentran las clases y las interfaces, en los de comportamiento vamos a encontrar a los mensajes, los estados y las acciones, en los de agrupación se encuentran los paquetes, en los elementos de anotación se encuentran las notas.

En el lenguaje de modelado UML cada concepto de la POO tiene su propia representación, así una clase está definida por el símbolo de la Figura 14.

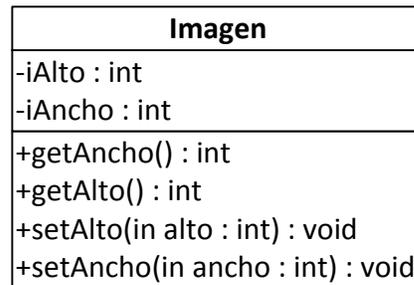


Figura 14 UML de la clase Imagen

La clase anterior ilustra la abstracción de una imagen, en este caso se especifica que tiene ancho, alto y tiene métodos de acceso a estos datos.

Como esta clase se pueden tener cientos y a través de UML se puede definir su estructura, como se conectan, la funcionalidad que tienen. Es así como UML sirvió como apoyo a la POO para que diera un gran paso en los años 90.

También se puede representar el concepto de herencia que ya se explicó anteriormente, y el diagrama para representar la herencia es el mostrado en la Figura 15:

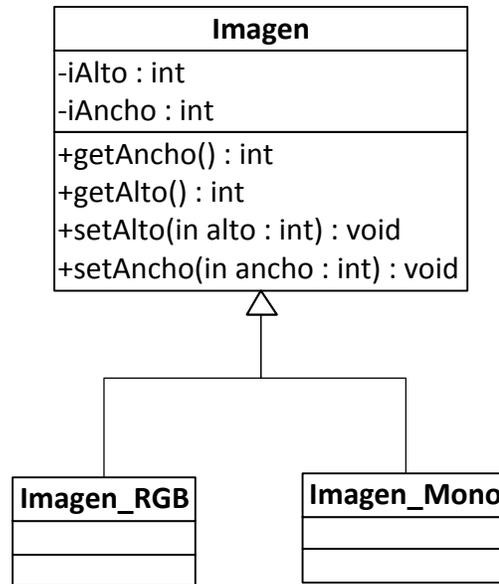


Figura 15. Diagrama de UML donde se representa la herencia.

Se podría decir que UML es como los planos de una casa, ya que una casa se puede concebir a través de varios planos. En esta analogía un sistema también se puede entender mucho más fácil con los diferentes diagramas que componen a este. Así si necesitáramos ver cómo funciona, o con que otros objetos se comunican un objeto podemos consultar varios diagramas entre ellos el diagrama de objetos.

Para más información sobre el lenguaje UML y los diagramas refiérase al libro (Grady Booch, 2006) que se muestra en la bibliografía.

3.3. CONCEPTOS BÁSICOS SOBRE SISTEMAS DE GESTIÓN DE APRENDIZAJE

3.3.1. LAS PLATAFORMAS EDUCATIVAS VIRTUALES

Hoy en día el uso de la tecnología es imprescindible para la educación y existen muchos casos donde se ha usado la tecnología para incrementar el nivel aprendizaje en el estudiante. Un caso particular son las herramientas educativas virtuales ya que han tenido un alto impacto en el proceso de enseñanza aprendizaje. Como lo dice su propio nombre estas plataformas educativas son usadas ampliamente en los centros donde se imparten clases y son herramientas que se usan de manera virtual, es decir, que el alumno no tiene que estar físicamente en el salón para tomar la clase sino que puede estar remotamente, ya sea desde su casa o desde un lugar muy lejano. Poder lograr que un alumno tome una clase desde cualquier lugar y a cualquier hora ha sido gracias al internet, ya que es a través de este medio como se logra que dos personas ubicadas en diferentes lugares lejanos, se puedan comunicar sin ningún problema.

Existen varias definiciones aceptadas sobre lo que es una plataforma virtual educativa, incluso han existido confusiones sobre las siglas, ya que en inglés existen las siglas LMS (Learning Management Systems), VLE (Virtual Learning Environment) y MLE (Managed Learning Environment), se refieren casi a los mismos conceptos, aquí se escogió una definición muy genérica sobre los que son las LMS.

Becta, en el Reino Unido, ha acuñado el término *plataforma educativa* para cubrir MLE y VLE según lo utilizado en el sector educativo.

“El término plataforma educativa describe una amplia gama de sistemas ICT (del inglés Information and Communication Technology, o TIC) usados para el entrenamiento y aprendizaje. A través de una plataforma educativa, hardware, software y los servicios de soporte que confluyen para permitir maneras de trabajo más eficaces, dentro y fuera del aula. En el corazón de cualquier plataforma educativa está el concepto de un espacio en línea de aprendizaje personalizado para el estudiante. Este espacio debe ofrecer a profesores y estudiantes acceso a los recursos almacenados de trabajo, recursos educativos virtuales, a la comunicación y a la colaboración entre estudiantes, y a la facilidad de acceso al progreso del estudiante.” - DfES Making IT Personal leaflet, Marzo 2006¹⁹.

Las plataformas educativas virtuales como se comentaba antes han sido muy útiles para la educación y con el uso del internet es como se han popularizado.

¹⁹ http://es.wikipedia.org/wiki/Ambiente_Educativo_Virtual, consultada en Noviembre 2011

Pero estas plataformas están englobadas o pertenecen a otro concepto más general que es el *e-Learning*, este concepto es importante ya que se refiere al hecho de usar medios electrónicos como la computadora para poder transmitir conocimiento, es en este concepto donde surge la raíz de las plataformas educativas ya que establece las ideas del uso del internet para impartir conocimiento, además dentro de esta misma área es donde surge otro concepto indispensable, que es el de *b-Learning* que se podría decir que deriva de lo que es el *eLearning* y es un híbrido entre la educación a distancia y la educación online.

Al igual que en las plataformas educativas virtuales, los especialistas en estas áreas han dado diferentes definiciones de lo que es el *e-Learning*, pero todas estas definiciones se refieren a las mismas ideas y tienen como medio principal el uso de la tecnología como medio de transmisión de conocimientos. Dos de las definiciones de *e-Learning* elegidas en este trabajo son las siguientes:

1. Capacitación no presencial que, a través de plataformas tecnológicas, posibilita y flexibiliza el acceso y el tiempo en el proceso de enseñanza-aprendizaje, adecuándolos a las habilidades, necesidades y disponibilidades de cada discente²⁰, además de garantizar ambientes de aprendizaje colaborativos mediante el uso de herramientas de comunicación síncrona y asíncrona, potenciando en suma el proceso de gestión basado en competencias.²¹
2. El e-learning consiste en la educación y capacitación a través de Internet. Este tipo de enseñanza online permite la interacción del usuario con el material mediante la utilización de diversas herramientas informáticas.²²

Además de concepto de *e-Learning* surge el concepto de *b-Learning*, que se podría entender con un híbrido entre la educación presencial y la educación a distancia es decir eLearning, la manera en cómo se complementan estas 2 formas de transmitir conocimientos genera un mayor impacto sobre los conocimientos que adquiere el alumno. Si bien uno de los puntos por los que ha sido muy criticado el e-Learning es que evita la interacción persona-persona entre el profesor y el alumno y con el uso del concepto de b-Learning se rompe este

²⁰ adj. y com. [Persona] que cursa estudios y recibe enseñanzas:

²¹ Francisco José García Peñalvo, usal.es, consultado el 28 de noviembre de 2010

²² <http://www.e-abclearning.com/definicion-e-learning>, Consultada en noviembre 2011

punto en contra ²³, ya que a través de este el profesor interactúa con el alumno de manera presencial y además el alumno puede tomar clases y resolver ejercicios de manera virtual. Es menester mencionar que las clases tanto usando el modelo presencial como e-Learning tienen que estar bien estructuradas pedagógicamente ya que debido a que se complementan sería un desperdicio de tiempo que lo que se vio en la clase de un salón, lo repita el alumno a través del curso online.

Actualmente el uso del b-Learning y e-Learning se ha incrementado ya que muchas de las escuelas se han dado a la tarea de invertir en estos medios para transmitir conocimientos y tener un mayor número de alumnos que se inscriban a sus cursos.

El diagrama de la Figura 16 es perteneciente a una escuela de inglés donde se imparten las clases utilizando este concepto de b-Learning. En el diagrama se ilustra el tiempo aproximado que hay que dedicar a la educación presencial y a la educación a distancia.

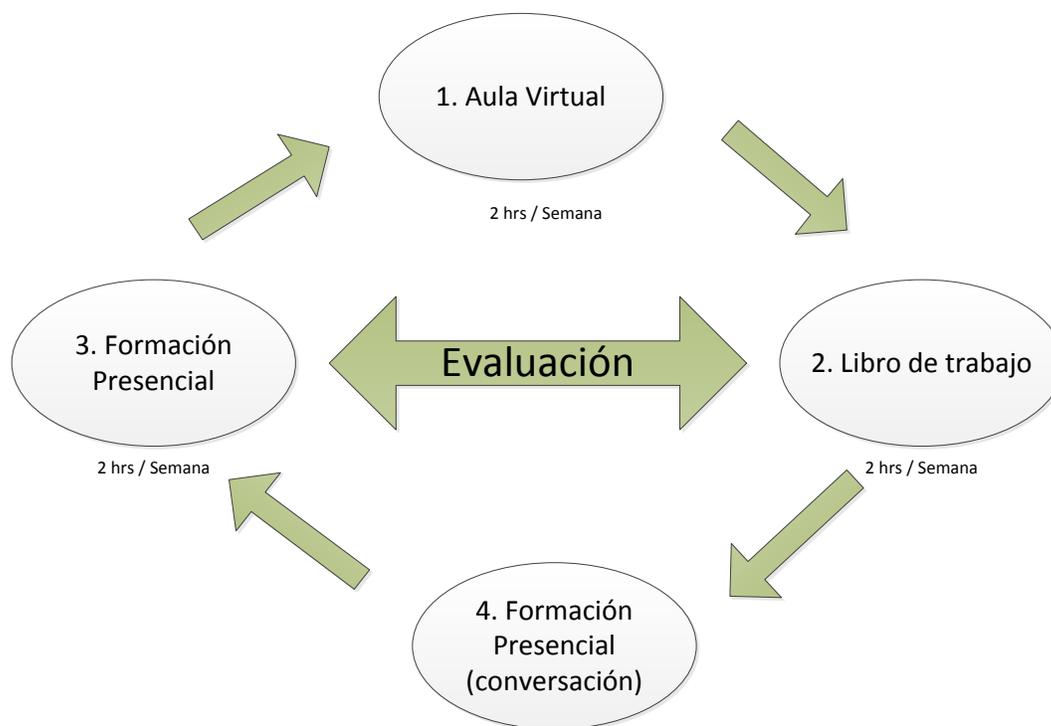


Figura 16. Esquematización de bLearning <http://www.langrow.com>

²³ La eficacia ha sido probada por la dirección general de educación educativa que ha puesto al servicio de la comunidad universitaria diversas aplicaciones de lecciones interactivas <http://dgee.unam.mx>

3.3.2. COMPARATIVA DE LAS PLATAFORMAS EDUCATIVAS VIRTUALES MÁS USADAS.

Hoy en día existe una gran cantidad de plataformas educativas virtuales que son ocupadas para impartir cursos en todos los niveles de la educación, esto debido a que cada vez más personas tienen acceso a computadoras con acceso a internet, además de que los servicios de internet se abaratan.

Para poder decidir el uso de que plataforma educativa vamos a utilizar en nuestro curso o en este caso en el material didáctico que vamos a implementar es necesario conocer un las plataformas que se usan mas así como las ventajas y desventajas que estas proporcionan, por ejemplo, si disponemos de recursos económicos para poder contratar un servicio de alojamiento remoto y para poder comprar una licencia de software, podemos optar por una plataforma de paga; en cambio si no contáramos con estos recursos existen muchas plataformas educativas que son libres.

En la Tabla 3 (ver al final del trabajo mesografía) se presentan algunas de las plataformas educativas más utilizadas evaluadas según ciertos criterios, esto para apoyar la decisión de que plataforma vamos a ocupar.

Plataforma	OpenSource	Interfaz de usuario amigable	Fácil instalación	Licencia	Seguridad	Documentación
Chamilo	Si	Alta	Si	GNU/GPLv3	Alta	Poca
Moodle	Si	Media	Si	GNU/GPLv3	Media	Bastante
Dokeos	Si (pero cuenta con versión de paga)	Alta	No	GNU/GPL	Media	Media
Claroline	Si	Alta	Si	GNU/GPL	Baja	Bastante
eCollege	No	Media	No	CC-BY-SA.	Alta	Media
WebCT	No	Media	Si	WebCT Standard Edition	Alta	Media

Tabla 3. Comparativa de Plataformas Educativas

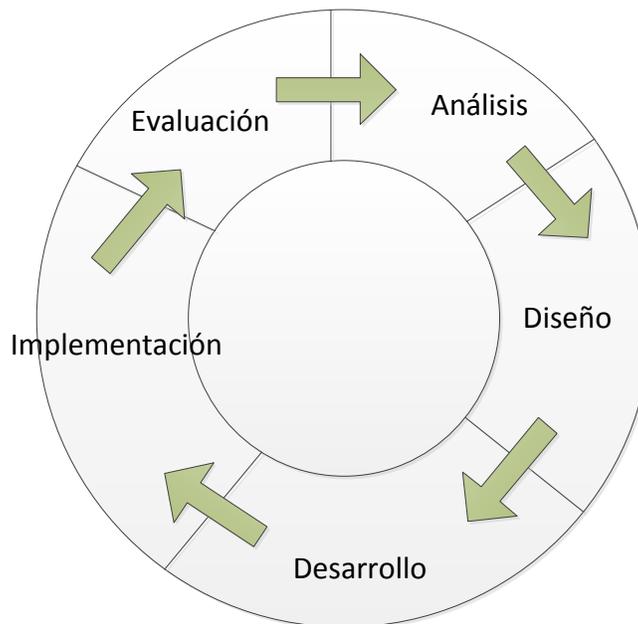
En la actualidad existen una gran variedad de plataformas educativas virtuales entre ellas encontramos Moodle, eCollege, WebCT, Doqueos, Claroline y Chamilo; esta última la que se escogió para este caso práctico. Estas plataformas virtuales educativas dan facilidades tanto para profesores en la transmisión de conocimientos como para alumnos en el aprendizaje de estos. Estas LMS²⁴ proporcionan características que van desde poder crear lecciones con texto, video y audio, hasta poder realizar evaluaciones a los alumnos de manera virtual, además de poder obtener datos como cuando tiempo paso el alumno en la plataforma virtual y que días accedió, etc.

Se decidió ocupar la plataforma educativa virtual Chamilo para el material didáctico de la materia de POO que se propone en este trabajo, debido a que Chamilo ofrece grandes ventajas entre las cuales se encuentran que es un software OpenSource y Freeware, es decir de código abierto y gratis ya que cuenta con la licencia GNU/GPL, además es un software multiplataforma ya que está disponible para varios sistemas operativos y es un sistema relativamente nuevo (surgió en enero del 2010).

²⁴ Por las siglas en ingles Learning Management Systems.



CAPITULO 4



4. ANÁLISIS, DISEÑO E IMPLEMENTACIÓN

En este capítulo se presenta la propuesta del temario de la materia de POO, además se muestra cual fue el proceso de análisis, diseño e implementación de los recursos didácticos en la plataforma educativa virtual Chamilo. Así mismo la metodología que se siguió para obtener el material didáctico para el primer capítulo del temario.

4.1. ANÁLISIS DEL OBJETIVO PRINCIPAL DEL USO DE CHAMILO.

Como ya se mencionó anteriormente el uso de Chamilo proporciona varias ventajas con respecto a otras plataformas educativas virtuales, para analizar porqué se decidió del uso de Chamilo y cuáles son sus ventajas principales es necesario primero definirlo y presentar sus características.

Chamilo es un sistema web que organiza procesos de enseñanza y aprendizaje a través de contenidos instruccionales e interacciones colaborativas.

Se puede decir que Chamilo es:

- Contenedor de cursos.
- Espacio de comunicación.
- Repositorio de documentos.
- Herramienta de estudio independiente/grupos

Chamilo proporciona diferentes características para profesores y alumnos algunas de estas son las siguientes:²⁵:

- Un sistema repositorio de contenidos con manejo de versión y funcionalidad de colaboración.
- Manejo de usuarios y grupo de usuarios.
- Integraciones con algunos de los más usados sistemas-sign-on.
- Manejo avanzado de privilegios.
- Frameworks para permitir el seguimiento de usuarios y reportes en varios formatos.
- Una manera fácil y flexible para crear enlaces entre aplicaciones individuales.
- Amplia variedad de librerías generales para manejo de capas, formularios, traducciones, tablas, importaciones, exportaciones, calendarios, correo y mucho más.

²⁵ <http://www.chamilo.org/es/noticias/liberacion-chamilo-20-estable>, consultado en Diciembre 2011

4.2. DISEÑO DE LOS MÓDULOS PARA LA IMPLEMENTACIÓN DEL MATERIAL DIDÁCTICO DE POO.

Chamilo proporciona una interfaz de usuario muy amigable, a través de esta permite decidir que módulos se mostraran en la plataforma virtual y cuáles no, estos módulos dependen del enfoque que se le quiera dar a la plataforma educativa. En este caso práctico no se pretende que sea un curso de POO, sino más bien es la implementación de material de apoyo para esta materia, debido a esto algunas funcionalidades como Evaluaciones y Asistencia fueron deshabilitadas. En la Figura 17 se presentan los módulos que fueron habilitados en la categoría de manejo de contenidos.



Figura 17. Manejo de contenidos plataforma virtual educativa Chamilo.

La sección de interacción es indispensable ya que a través de esta es como el alumno logra comunicarse con el profesor. En esta existen varios módulos de utilidad como los foros, chat y compartir documentos, además de una agenda, notas personales y encuestas. Aquí fueron deshabilitados algunos módulos como tareas, grupos, wiki y usuarios. En la Figura 18 se presentan los contenidos que se habilitaron para la categoría de interacción con el usuario.

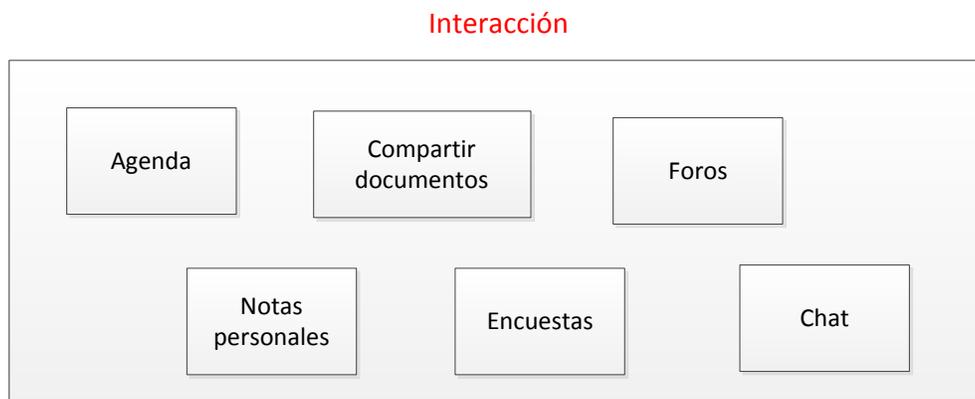


Figura 18. Interacción, plataforma virtual educativa Chamilo.

4.3. DESCRIPCIÓN DE LOS MÓDULOS QUE CONFORMAN LA PLATAFORMA EDUCATIVA CHAMILO.

Como se mencionaba anteriormente Chamilo proporciona varias características, algunas de estas pueden ser habilitadas o deshabilitadas a través de la interfaz de usuario, los módulos que se decidió activar fueron mostrados en la sección anterior estos fueron divididos en módulos de manejo de contenidos y módulos de interacción, a continuación se describen en qué consisten cada uno de estos módulos que fueron activados en Chamilo para el material didáctico de POO que se creó.

a) Módulos de manejo de contenidos

Descripción
del curso

En este módulo se presentan los objetivos del material didáctico que se presentan en la plataforma virtual educativa Chamilo. También se muestra el temario propuesto del curso base de POO que se seguirá para la impartición de la materia. Finalmente también se presenta la metodología y la bibliografía que se sugiere a el alumno de la materia POO, la metodología incluye como está estructurado todo el material y las características que este proporciona.

Documentos

En esta sección se coloca todo el material didáctico adicional proporcionado por el profesor, entre los documentos pueden ser tutoriales de programación, artículos creados por el profesor o por investigadores reconocidos, libros y otros documentos útiles para la materia. Además en esta sección se presentan todos los temas que forman parte de las lecciones, estos textos pueden ser descargados por el alumno, ya sea todos los documentos o cierto documento en específico.

Enlaces

En la sección de enlaces se proporcionan ligas a páginas importantes para la materia de POO, el principal objetivo de esto es que el alumno tenga otras fuentes de información además de la información que le proporciona el profesor. En los enlaces por ejemplo se podrían colocar ligas a definiciones sobre POO o a conceptos definidos por especialistas en la materia, lo anterior de acuerdo al tema tratado.

Anuncios

En esta sección el profesor podrá dar a conocer a los alumnos diferentes tipos de anuncios, entre algunos se encuentran las calificaciones de los alumnos, la fecha en que serán los exámenes, fechas en que no habrá clases, la fecha en que se acaban los cursos, entre otros anuncios. Esta sección es muy útil para mantener comunicación asíncrona del profesor al alumno ya que incluso alumnos que no hayan podido asistir a clase se podrán enterar de datos importantes en la sección de anuncios.

Glosario

El glosario está pensado para colocar las palabras más usadas en el curso de POO, algunas veces el alumno tiene la necesidad de consultar ciertos conceptos que le resultan difíciles de entender o palabras que se encuentran en una lección pero que nunca había escuchado. Es en este momento donde el glosario es de gran ayuda para el alumno.

Lecciones

La sección de lecciones es una de las más importantes del material didáctico que se implementó en la plataforma educativa Chamilo, en esta sección se desarrolla el temario propuesto de la materia de POO. Las lecciones llevan una secuencia de presentación de acuerdo al temario, las lecciones pueden estar compuestas de elementos multimedia, tales como texto, imágenes, video y sonidos, además se pueden complementar con otros documentos, ejercicios y enlaces web. Con el uso

de estos elementos se provoca un mayor impacto en el aprendizaje sobre el alumno que toma la clase.

Ejercicios

Como ya se comentaba en la sección de lecciones, estas pueden ir acompañadas de ejercicios aunque no necesariamente. Los ejercicios normalmente van enlazados a un tema y estos ejercicios pueden ser de varios tipos, entre algunos se encuentran los de completar preguntas, elegir una opción de varias, o elegir más de una opción de varias, enlazar pregunta o concepto con una respuesta, etc., esto puede variar de acuerdo al tipo de pregunta que se esté realizando, además los ejercicios dan un puntaje por cada pregunta acertada, por lo que al terminar de resolver el conjunto de ejercicios el alumno podrá obtener estadísticas de su puntaje y con esto saber su grado de comprensión del tema.

b) Módulo de interacción.

Compartir Documentos

Como su nombre lo dice este módulo permite compartir documentos entre los alumnos que se registraron al curso de POO, la utilidad de esto se manifiesta cuando el alumno tiene la necesidad de compartir con su compañero una tarea que se dejó en equipo o algún documento que le puede ser útil al compañero para resolver alguna duda.

Encuestas

Las encuestas están pensadas para los profesores o para los supervisores de un profesor, ya que a través de esta sección se puede preguntar a los alumnos que den su opinión sobre el curso presencial o el material proporcionado virtualmente a través de la plataforma educativa, incluso puede ser utilizado por los supervisores de profesores para cuestionar al alumno sobre el desempeño y los

conocimientos del profesor en el salón de clases, estos y otros usos pueden ser dados a las encuestas.

Agenda

En la agenda se establecerán las fechas y horas de entregas de ejercicios así como fechas de exámenes y entrega de programas o proyectos, esta sección es muy necesaria para el estudiante, ya que cada vez que ingrese con su cuenta a la plataforma Chamilo podrá entrar a la sección agenda y revisar tareas marcadas y fechas importantes.

Foros

En esta sección de foros en la plataforma Chamilo, se proporciona una herramienta de comunicación asíncrona tanto para el profesor como para los alumnos. Frecuentemente en esta sección se presentan dudas que tienen los alumnos ya sea sobre algún concepto que es difícil de entender o tal vez algún problema o error que le envía el compilador al alumno cuando está haciendo un programa, estas dudas pueden ser aclaradas por el profesor o por otros alumnos que pasaron por la misma situación y ya saben cómo resolver el problema.

Chat

El chat ha sido una herramienta para socializar e intercambiar ideas entre las personas, desafortunadamente ha sido más utilizado para fines de entretenimiento, pero en el caso de Chamilo ha sido implementado con el fin de que los alumnos puedan conversar y aclarar dudas con otros de sus compañeros o con el profesor, esto es una herramienta complementaria a los foros, ya que algunas veces las preguntas en los foros tardan en ser resueltas y en cambio en el chat es una interacción directa (comunicación síncrona) y en el momento.

Notas
personales

La herramienta de notas personales es un plus que da soporte a la plataforma educativa Chamilo, ya que permiten a los alumnos agregar notas que sólo pueden ser revisadas en su sesión, estas notas pueden ser algún dato necesario que surgió cuando tomaban la lección o tal vez alguna duda que tengan es ese momento pero que quieren recordar más tarde.

4.4. TEMARIO PROPUESTO PARA LA MATERIA PROGRAMACIÓN ORIENTADA A OBJETOS.

Para la creación del material didáctico de la materia POO es necesario contar con un temario sobre esta materia, en el cual se vayan presentando cada uno de los conceptos de la POO de una manera ordenada para posteriormente construir conceptos más complejos. Para esto se realizó una propuesta de temarios para la materia de POO, es necesario mencionar que la mayoría de temarios en las universidades tienen muchas similitudes y tal es el caso de la materia POO en la cual generalmente se abordan temas acerca de lo que es un objeto, una clase, lo que es la abstracción, el polimorfismo, herencia, etc., es por esto que se decidió obtener de por una generalización los temas que se ven en varias universidades.

Para esto se realizó una propuesta de temarios a través de consultar especialistas en el tema y alumnos, esto para obtener un temario más genérico y enfocado a la carrera y a las necesidades que hay en el campo de trabajo.

El temario propuesto consta de 5 secciones y está basado en los temarios de varias universidades que actualmente imparten la materia de POO, entre ellas la FES Aragón, Facultad de Ingeniería y el IPN.

INTRODUCCIÓN

SECCIÓN 1. LA PROGRAMACIÓN ORIENTADA A OBJETOS Y OTROS PARADIGMAS

- *El paradigma de la POO*
- *El concepto de objeto*
- *Mensajes entre objetos*
- *Lenguajes orientados a objetos*
- *Otros paradigmas de programación*
- *Ejercicios de autoevaluación*

SECCIÓN 2. BREVE DESCRIPCIÓN DE LOS CONCEPTOS BÁSICOS DE POO

- *Objetos*
- *Clases*
- *Mecanismos de la POO*
- *Ejercicios de autoevaluación.*

SECCIÓN 3. EL LENGUAJE DE PROGRAMACIÓN C++ Y LA POO

- *C++ como lenguaje orientado a objetos*
- *Clases*
- *Constructores y Destructores*
- *Herencia*
- *Polimorfismo*
- *Sobrecarga de operadores*
- *Ejercicios de autoevaluación*
- *Caso práctico: implementación de un editor de texto en Qt*

SECCIÓN 4. CONCEPTOS AVANZADOS DE POO

- *Relaciones*
- *Excepciones*
- *Interfaces y clases abstractas.*
- *Herencia múltiple*
- *Plantillas*
- *Iteradores*
- *Ejercicios de autoevaluación*

SECCIÓN 5. EL LENGUAJE DE MODELADO UML

- *El lenguaje de modelado UML y la arquitectura de sistemas*
- *Fases del ciclo de desarrollo de software.*
- *Diagramas de estructura.*
- *Diagramas de comportamiento.*
- *Diagramas de interacción.*
- *Caso práctico: documentación en UML de la arquitectura del editor de texto.*

4.5. PROCESO DE IMPLEMENTACIÓN DEL CAPÍTULO 1 DE LA MATERIA POO.

El diagrama de la Figura 19 ilustra el proceso que se siguió para la implementación del material didáctico, en este se observan cada una de las etapas y los entregables generados.

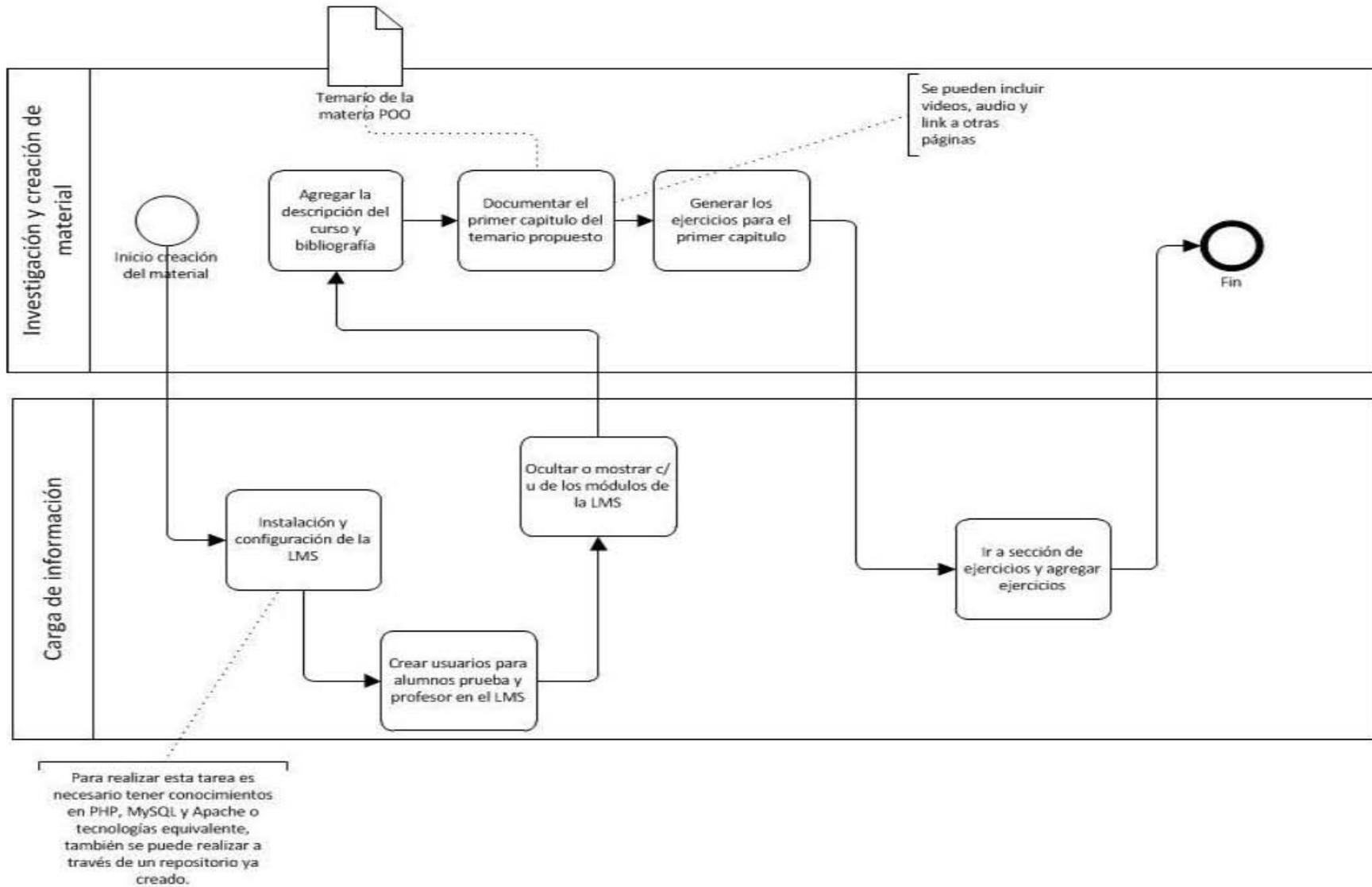


Figura 19. BPMN (*Business Process Modeling Notation*) de los procesos que se seguirán para la implementación del caso práctico

- **Instalación y Configuración de la LMS**

El primer paso es instalar y configurar la plataforma virtual educativa (ver el manual de instalación de Chamilo de manera local en el Apéndice) , en este caso existen 2 maneras de hacerlo, la primera y la más complicada es instalarlo en nuestro propio servidor, para esto se necesitan tener conocimientos en la instalación y configuración de PHP, MySQL y Apache(ver manual de instalación en el Apéndice).

Este procedimiento consiste en tener todos nuestros contenidos en nuestro servidor y poder accederlos desde nuestro equipo (en caso de que los queramos ver sólo en nuestra computadora) o por otro equipo siempre y cuando proporcionemos acceso a nuestro servidor. La principal ventaja es que nosotros podemos administrarlo y no se tiene que depender de otra empresa u otro servicio que proporcione un repositorio de Chamilo.

El uso de servicios externos es muy utilizado ya que estos servicios hacen transparente la administración del servidor, así como la seguridad de este. Normalmente estos servicios se proporcionan por una cuota mensual o anual dependiendo como se haya hecho el contrato. Esta modalidad es utilizada por muchas universidades ya que normalmente estas cuentan con servidores los cuales dan el servicio de alojamiento, este servicio puede ser solicitado por el profesor sin costo o en algunas veces por cuotas demasiado bajas.

Finalmente se debe señalar que como se ha mencionado en todo el trabajo, el propósito de este trabajo no es implementar todo el curso de POO, sino que es mostrar el procedimiento para poder crear material didáctico de ciertas secciones del curso así como mostrar cómo se estructuró la información en una plataforma virtual educativa y con base a este procedimiento se cree el material didáctico faltante para usarlo en las aulas de clase. Con toda la base teórica abordada se tendrá la capacidad de hacer uso de este material para el aula de clases, así como de crear nuevo material de los demás capítulos.

- **Creación de usuarios en la LMS**

La creación de usuarios de la plataforma virtual es una tarea muy sencilla, que puede realizarse por el propio usuario o puede solicitarse la aprobación del profesor, en el caso de esta implementación se decidió que los usuarios fueran quienes pudieran registrarse dentro de la plataforma virtual y posteriormente darse de alta en el sistema para poder entrar. Además para el administrador o para el

profesor se proporciona una sección de usuarios donde puede acceder para visualizar los usuarios que se han registrado en el sistema, la imagen de la Figura 20 muestra la interfaz cuando accede a la zona de usuarios en Chamilo.

Foto	Código Oficial ↑	Apellidos	Nombre	Descripción	Grupo	Estado	Activo	Acción
<input type="checkbox"/>	55138	46794	Albert	-	-	-	✓	
<input type="checkbox"/>	2406	BURGOS FLORES	JEFFERSON	-	-	-	✓	
<input type="checkbox"/>	CATERINE	Cerda Tapuy	Mireya Caterine	-	-	-	✓	
<input type="checkbox"/>	prof	Chamilo	Profesor 1	Profesor	-	Profesor	-	
<input type="checkbox"/>	ALUMNOPOO1	Curso	Alumno POO1	-	-	-	✓	
<input type="checkbox"/>	RICARDO143	rodriguez tadeo	yedem	-	-	-	✓	
<input type="checkbox"/>	Robert	T	Robert	-	-	-	✓	

Seleccionar todo - Anular seleccionar todos

Figura 20. Administración de usuarios Chamilo.

La imagen de la Figura 20 muestra las capacidades que tiene el usuario de administrador, ya que le permite agregar alumnos, agregar profesores, gestionar grupos, entre otras tareas.

El siguiente paso consiste en definir que secciones van a estar visibles para los alumnos. Estas tareas pueden hacerse como profesor o administrador, la siguiente ventana muestra la interfaz para poder mostrar u ocultar las secciones que se desean proporcionar a los alumnos.

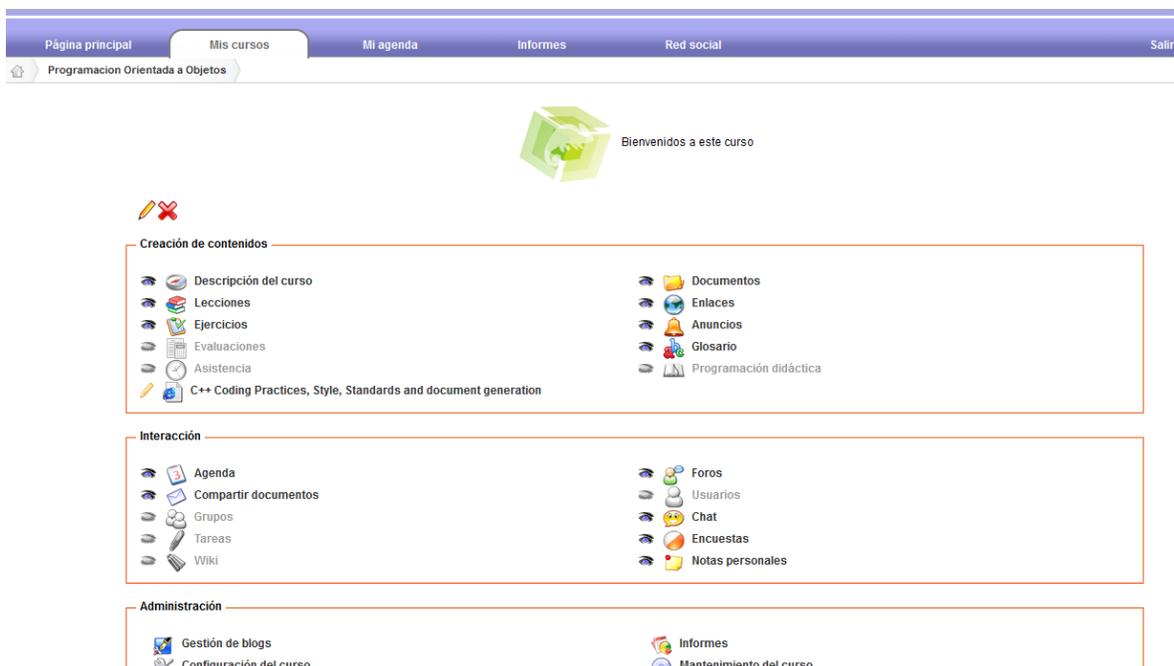


Figura 21. Pantalla Principal Chamilo.

El pequeño ojo situado del lado izquierdo del módulo es el que permite que se muestren o se oculten las secciones que componen Chamilo (véase la Figura 21). La decisión de que secciones se ocultaran o se mostraran dependerá del objetivo de lo que queremos implementar, así si quisiéramos impartir nuestro curso online y realizar evaluaciones a través de la plataforma educativa tendríamos que tener activada la sección de evaluaciones.

- **Descripción del curso y bibliografía**

Una de las partes más importantes de un curso de programación orientada a objetos es la sección donde se definen los objetivos, es decir, donde se describe que se quiere lograr, como se va a lograr y la bibliografía que se proporciona a el alumno. Esta sección se conoce en Chamilo como descripción del curso.

La imagen que se presenta en la Figura 22 representa la sección de la plataforma educativa Chamilo donde se le permite al profesor añadir los objetivos, metodología, los temas que se revisaran, la bibliografía, etc. Esta sección es una de las más importantes ya que es en este punto donde se capta la atención del alumno y donde se resumirá en pocas palabras de lo que se trata el curso, esta sección normalmente es de las primeras que se muestran y de las primeras que el alumno consulta.

Página principal | Mis cursos | Mi agenda | Informes | Red social | Panel de control | Administración de la plataforma | Salir (robtelez)

Programación Orientada a Objetos > Descripción del curso

Descripción general | Objetivos | Temas | Metodología | Materiales | Recursos humanos y técnicos | Evaluación | Apartado personalizado

La Programación Orientada a Objetos

El objetivo principal de esta implementación virtual es proporcionar al alumno herramientas didácticas que complementen un curso de Programación Orientada a Objetos (POO).

A través de este material didáctico el alumno podrá entender algunos de los conceptos más complicados de este paradigma de programación y usarlos en ejemplos prácticos. Entre algunos de estos conceptos se encuentran: el polimorfismo, la herencia, la abstracción de datos y la encapsulación.

El propósito final de este material es que el alumno pueda ocupar en aplicaciones prácticas todos los conceptos de POO adquiridos tanto en el salón de clases así como con el uso de este material didáctico. Estos conceptos le serán de gran utilidad al alumno, al momento de implementar sistemas de software en la vida real y principalmente al implementar sistemas que sean flexibles, reutilizables, robustos y que estén adecuadamente documentados.

Adicionalmente a las lecciones sobre POO se presentan ejercicios de práctica en cada uno de los módulos que forman parte del temario, además se proporcionan referencias adicionales de libros y artículos, estas últimas serán de gran utilidad para las personas que estén interesados en temas muy particulares y que deseen continuar especializándose en el área de programación e ingeniería en software.

La siguiente figura es un mapa mental donde se ilustran los conceptos principales que forman parte de la Programación Orientada a Objetos.

El diagrama es un mapa mental con un nodo central en forma de nube que contiene el texto 'POO'. Desde este nodo central, se extienden tres líneas que conectan con tres rectángulos periféricos: 'CLASES' a la izquierda, 'ABSTRACCIÓN DE DATOS' en la parte superior derecha y 'ENCAPSULACIÓN' a la derecha.

Figura 22. Descripción General.

- **Documentación de los capítulos.**

El paso siguiente es crear los elementos principales del curso, es decir, los contenidos, la creación de los contenidos puede ser usando elementos media como imágenes, video, audio y texto; estos en conjunto logran que se transmita la información con más fluidez y facilidad hacia los alumnos.

Para realizar la creación de contenidos Chamilo proporciona una sección que se denomina Lecciones, estas tienen que ser creadas usando el formato SCORM²⁶. Las lecciones pueden ir acompañadas de ejercicios que se realizan al final de la lección, esto para que el alumno refuerce los conocimientos aprendidos y a través de un puntaje saber el nivel de conocimientos adquirido.

La interfaz para poder crear lecciones es muy sencilla, esta proporciona dos opciones, crear una lección SCORM o importar una lección SCORM ya existente. En la siguiente Figura 23 se ilustra la interfaz de usuario del módulo de Lecciones.

²⁶ **SCORM** (del inglés *Sharable Content Object Reference Model*) es una especificación que permite crear objetos pedagógicos estructurados. Los sistemas de gestión de contenidos en web originales usaban formatos propietarios para los contenidos que distribuían. Como resultado, no era posible el intercambio de tales contenidos. Con SCORM se hace posible el crear contenidos que puedan importarse dentro de sistemas de gestión de aprendizaje diferentes, siempre que estos soporten la norma SCORM(<http://es.wikipedia.org/wiki/SCORM>, consultada en Octubre 2011).

Título	Fecha De Publicación	Fecha De Caducidad	Progreso	Autorización De Opciones
LA PROGRAMACIÓN ORIENTADA A OBJETOS Y OTROS PARADIGMAS Chamilo	miércoles 6 de julio de 2011 23:00		100%	✎ ✂ 👁 📄 📁 ➡ 🔍 ⚙️ ⚠️ ⚡
BREVE DESCRIPCIÓN DE LOS CONCEPTOS BÁSICOS DE POO Chamilo	jueves 7 de julio de 2011 22:23		0%	✎ ✂ 👁 📄 📁 ➡ 🔍 ⚙️ ⚠️ ⚡
EL LENGUAJE DE PROGRAMACIÓN C++ Y LA POO Chamilo	jueves 7 de julio de 2011 22:25		0%	✎ ✂ 👁 📄 📁 ➡ 🔍 ⚙️ ⚠️ ⚡
CONCEPTOS AVANZADOS DE POO Chamilo	jueves 7 de julio de 2011 22:37		0%	✎ ✂ 👁 📄 📁 ➡ 🔍 ⚙️ ⚠️ ⚡
EL LENGUAJE DE MODELADO UML Chamilo	jueves 7 de julio de 2011 22:38		0%	✎ ✂ 👁 📄 📁 ➡ 🔍 ⚙️ ⚠️ ⚡

Figura 23. Interfaz de usuario de Chamilo para las lecciones.

Es necesario recordar que todas las imágenes de la interfaz de Chamilo mostradas en este trabajo fueron tomadas usando el usuario de profesor, ya que en la vista de alumno las imágenes se ven restringidas en muchas características, ejemplo de esto, sería en la sección de Lecciones mostrada en la imagen de la Figura 23, en la vista de alumno sólo se muestran las lecciones y no muestra las opciones para poder crear o importar una lección.

- **Generación de ejercicios.**

El paso siguiente después de realizar las lecciones es generar ejercicios complementarios. Algunas lecciones pueden tener o no ejercicios, y para este caso del material didáctico de POO se decidió colocar ejercicios al final del primer capítulo, esto para ejemplificar el desarrollo de los ejercicios y el potencial que pueden tener para el proceso de enseñanza – aprendizaje.

A continuación en la Figura 24 se muestra una imagen de la interfaz que proporciona Chamilo para poder crear ejercicios.



Figura 24. Interfaz de Usuario para crear ejercicios.

En la Figura 24 se visualizan las diferentes funcionalidades que proporciona Chamilo, algunas de ellas son agregar ejercicios, o importarlos varias fuentes, incluso de un documento Excel. Los ejercicios como se mencionaba anteriormente se pueden añadir a las lecciones para que al finalizar la lección el alumno pueda realizar los ejercicios y obtener un puntaje que refleje el nivel de conocimientos adquirido sobre el tema. Estos ejercicios también pueden ser prácticos como es el caso de los ejercicios de los temas relacionados con POO, en estas secciones se deben colocar ejercicios donde se le solicite al alumno realizar programas.

En la siguiente Figura 25 se muestra la interfaz que proporciona Chamilo para poder crear las preguntas que conformaran el curso. Las preguntas pueden ser de opción múltiple, para contestar en una caja de texto, de relacionar, abiertas, etc.



Figura 25. Interfaz de usuario de Chamilo para poder agregar preguntas a los ejercicios.

- **Agregar ejercicios a lecciones.**

Cuando se está editando una lección se tiene la posibilidad de agregar ejercicios a esta, cabe mencionar que antes de agregar ejercicios a una lección es necesario primero crearlos y posteriormente agregarlos (una vez que se agregan los ejercicios a una lección ya no es posible editarlos, pero si permite eliminarlo para volverlo a crear). Chamilo proporciona una interfaz amigable para poder agregar ejercicios, para esto primero se tiene que editar la lección y posteriormente ir a la sección añadir objeto de aprendizaje (ver Figura 26) para finalmente elegir la serie de ejercicios que se va a agregar a la lección (ver Figura 27).

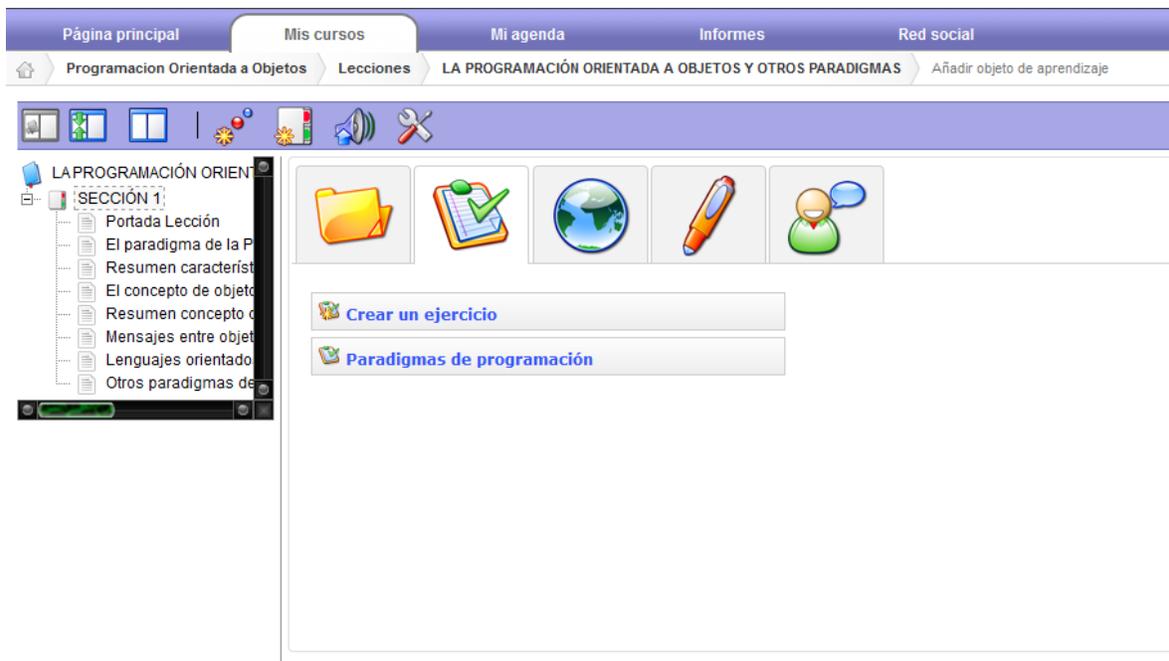


Figura 26. Chamilo, agregar objeto de aprendizaje.

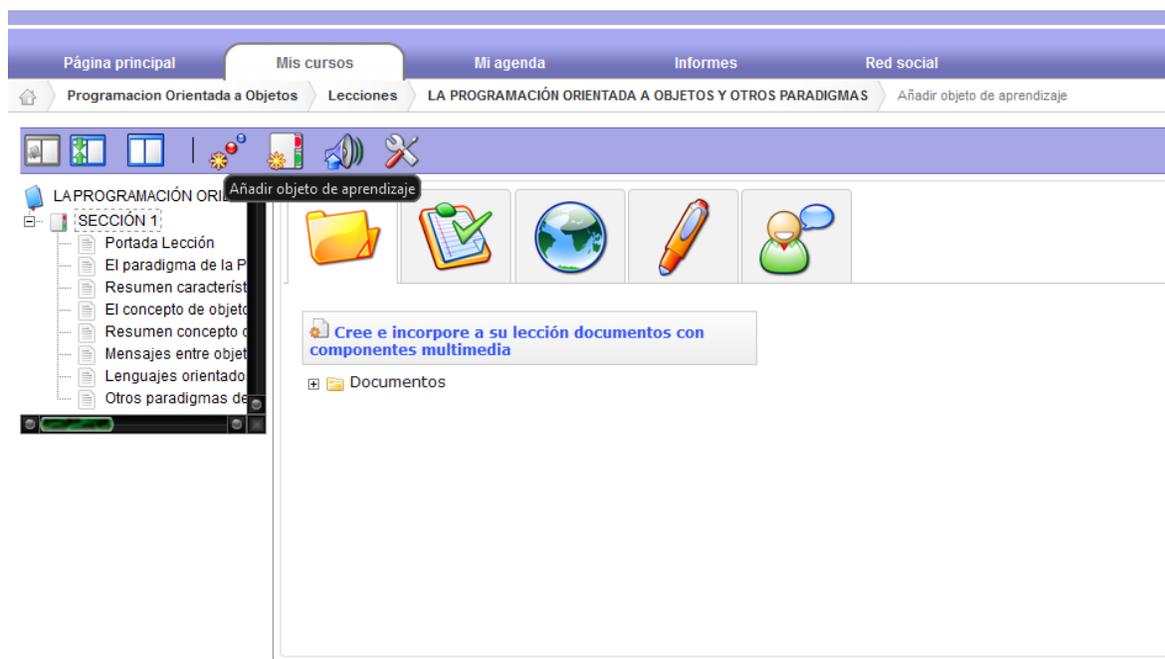


Figura 27. Chamilo, agregar ejercicios a una lección.

Después de haber agregado los ejercicios a una lección, se tiene el material didáctico listo. Es necesario mencionar que la plataforma educativa virtual Chamilo proporciona otras características adicionales para poder realizar el manejo de un curso, algunas como el avance temático, la asistencia, las evaluaciones, etc; sólo que en este material didáctico sólo se consideraron activar los módulos anteriores por cuestiones de simplicidad y por el propósito de este es material didáctico de apoyo y no un curso online de POO en el que se tenga que pasar lista y realizar evaluaciones sobre lo aprendido.

Finalmente se debe mencionar que el material didáctico implementado fue probado por ex alumnos egresados de la carrera de ingeniería en computación, revisados por alumnos de la carrera de pedagogía y por un profesor del CCH que actualmente imparte cursos utilizando plataformas educativas virtuales. Con base a todos los comentarios y sugerencias hechas por los que probaron la aplicación, se hicieron cambios y se agregaron las mejoras pertinentes al material didáctico.

4.6 PRÁCTICAS OCUPANDO CHAMILO

En esta sección se han hecho las practicas con usuarios, así mismo, la capacitación del sistema de Chamilo ha especialistas de la materia de POO. Es necesario mencionar que se consultó y se pidió probar el sistema a personas especialistas en las áreas de pedagogía y a un profesores que utilizan plataformas educativas virtuales para impartir sus clases, así como ingenieros en computación egresados de la carrera. Muchos los comentarios y sugerencias que se hicieron al momento de probar la plataforma fueron integrados.

Para lograr lo anterior se cargó el curso en la pc que funciona como servidor de Ingeniería en Computación, con la siguiente URL:

- ingeniería.aragon.unam.mx/chamilo

Para acceder a este sistema se ocuparon el siguiente usuario y contraseña:

Usuario: robtellez

Password: pPn5GACE

Las modificaciones y comentarios que se hicieron en la fase de pruebas de la plataforma educativa Chamilo fueron las siguientes:

- Las preguntas de la sección de ejercicios no estaban planteadas de manera adecuada por lo que se trabajó con 2 egresadas de la carrera de pedagogía para modificar la estructura de preguntas.
- Algunas de las preguntas de la sección de la sección de ejercicios se tuvieron que separar en preguntas de opción multiple y preguntas abiertas, ya que se me hizo el comentario de que no se podía asignar un puntaje a las preguntas abiertas.
- Como un comentario adicional se me pidió que agregara imágenes representativas de la institución, como el símbolo de la UNAM, para que de esta manera el curso se percibiera institucional.
- Se me solicito dar una pequeña explicación sobre cómo se podían agregar y editar contenidos a un curso, es decir, agregar imágenes, texto y video.

- También se me hicieron comentarios sobre la redacción de los capítulos ya que algunos de ellos se percibían confusos y no se entendía bien lo que se intentaba dar a conocer.
- Se me hicieron otros comentarios muy específicos sobre el material educativo, algunos como agregar imágenes en ciertas secciones o agregar videos que explicaran algún tema en particular.

Como se mencionaba anteriormente se consultaron a algunas personas para probar el curso cargado en esta plataforma educativa virtual, a continuación se presentan galerías de fotos que se tomaron en el transcurso de este proceso.

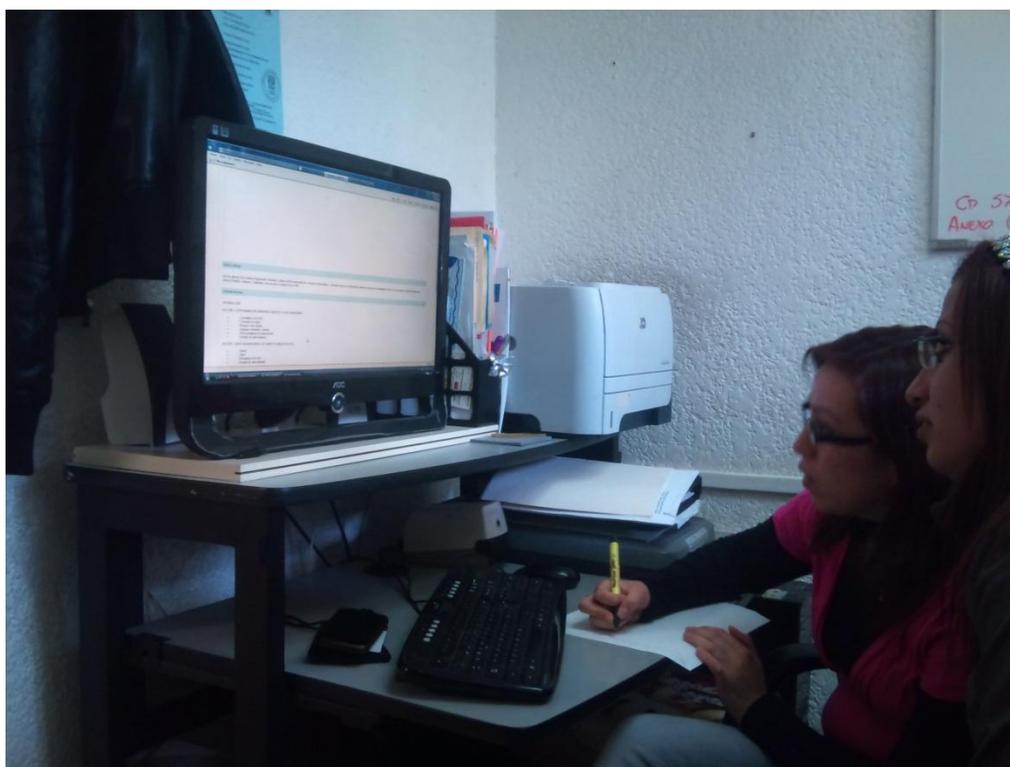


Ilustración 1. Plataforma Chamilo 1

En esta foto de la Ilustración 1 se muestran dos pedagogas que hacían comentarios sobre las mejoras que se podían hacer en el objetivo del curso de Programación Orientada a Objetos.

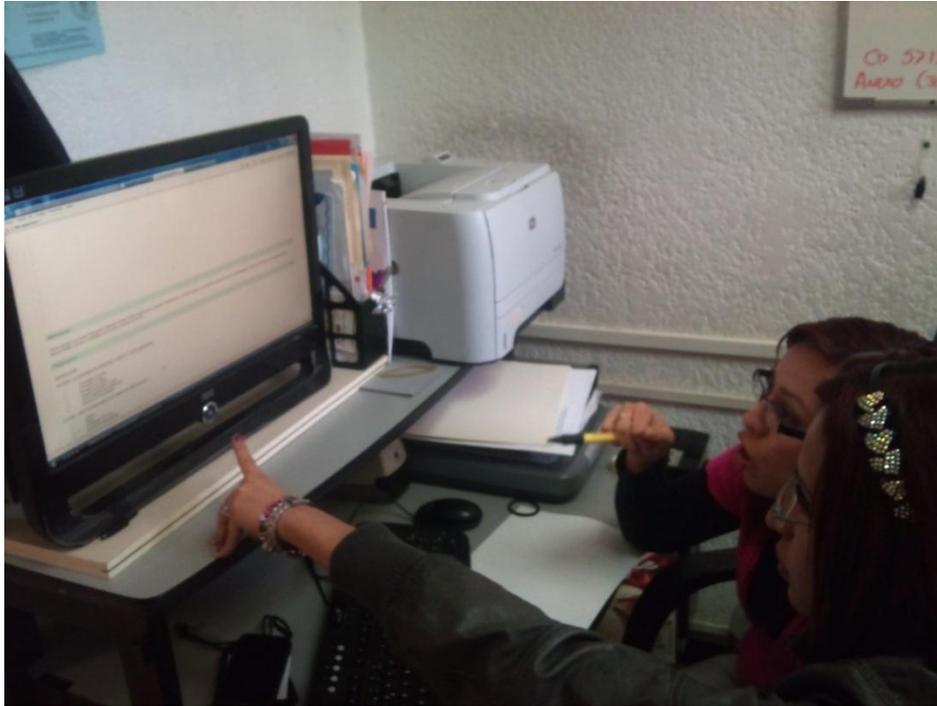


Ilustración 2. Plataforma Chamilo 2

En esta foto de la Ilustración 2 se muestran las mismas pedagogas haciendo comentarios sobre la estructura del temario, principalmente se discutía como es que se había llegado a este temario.

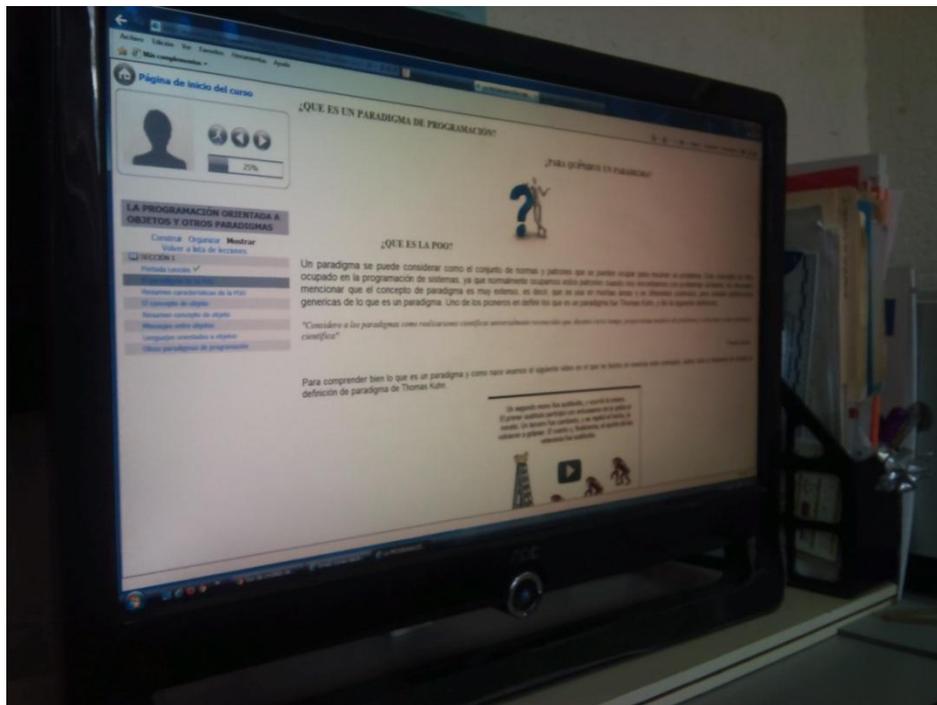


Ilustración 3. Plataforma Chamilo 3

En esta foto de la Ilustración 3 es cuando se comenzó a revisar la primera sección del material de Programación Orientada a Objetos

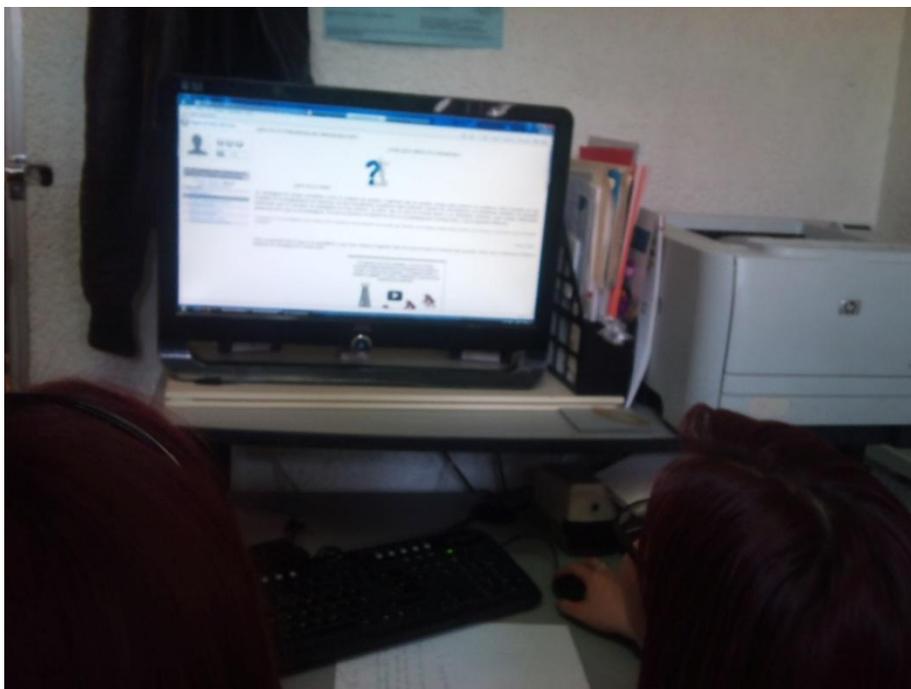


Ilustración 4. Plataforma Chamilo 4

En esta foto de la Ilustración 4 se está haciendo un análisis sobre lo que se quiere dar a entender en la sección de El Paradigma de la POO, la discusión se centraba sobre lo que es un paradigma

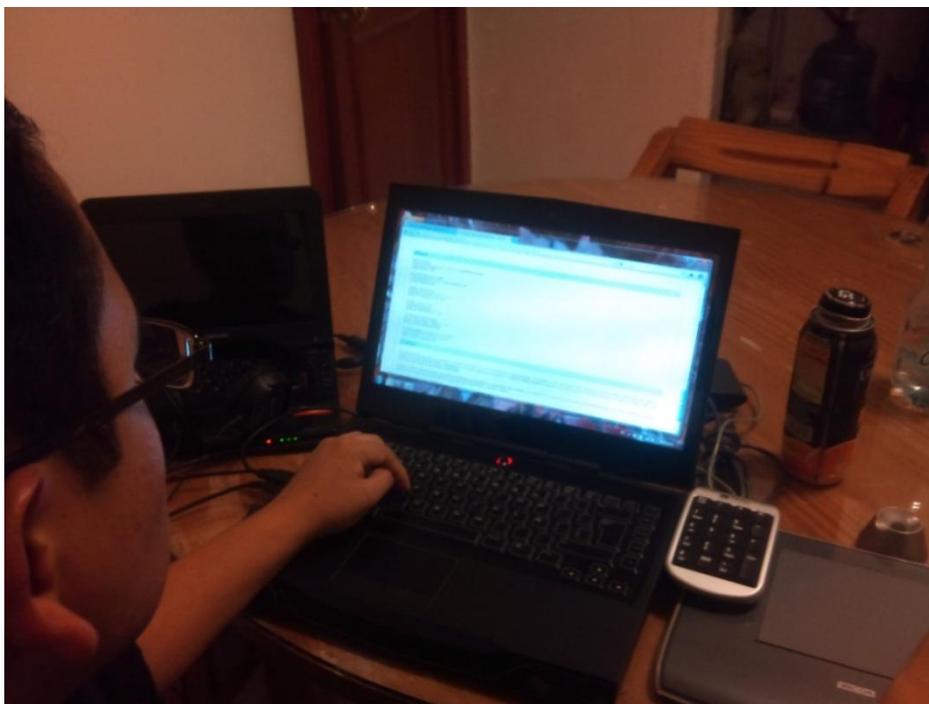


Ilustración 5. Plataforma Chamilo 5

También se consultó la opinión de un profesor de CCH, que es diseñador gráfico, el por su experiencia en el uso de plataformas educativas virtuales dio varias sugerencias sobre el contenido y los cambios que se podían hacer.



Ilustración 6. Plataforma Chamilo 6

En esta foto se están explicando algunas características de la interfaz gráfica de Chamilo así como el contenido del curso de POO.

A continuación se muestran algunas imágenes del material educativo que se encuentra cargado en la plataforma Chamilo de la FES Aragón. Estas imágenes son de diferentes secciones, esto debido a que el material educativo está dividido en capítulos en la sección de lecciones y además contiene una descripción del contenido.

La imagen de la Ilustración 7 y la Ilustración 8 se muestra la descripción del curso de POO así como la bibliografía que se tomó de referencia.

La Programación Orientada a Objetos

El objetivo principal de esta implementación virtual es proporcionar al alumno herramientas didácticas que complementen un curso de Programación Orientada a Objetos (POO).

A través de este material didáctico el alumno podrá entender algunos de los conceptos más complicados de este paradigma de programación y usarlos en ejemplos prácticos. Entre algunos de estos conceptos se encuentran: el polimorfismo, la herencia, la abstracción de datos y la encapsulación.

El propósito final de este material es que el alumno pueda ocupar en aplicaciones prácticas todos los conceptos de POO adquiridos tanto en el salón de clases así como con el uso de este material didáctico. Estos conceptos le serán de gran utilidad al alumno, al momento de implementar sistemas de software en la vida real y principalmente al implementar sistemas que sean flexibles, reutilizables, robustos y que estén adecuadamente documentados.

Adicionalmente a las lecciones sobre POO se presentan ejercicios de práctica en cada uno de los módulos que forman parte del temario, además se proporcionan referencias adicionales de libros y artículos, estas últimas serán de gran utilidad para las personas que estén interesados en temas muy particulares y que deseen continuar especializándose en el área de programación e ingeniería en software.

La siguiente figura es un mapa mental donde se ilustran los conceptos principales que forman parte de la Programación Orientada a Objetos.

```

graph TD
    POO((POO)) --- CLASES[CLASES]
    POO --- ABSTRACCIÓN_DE_DATOS[ABSTRACCIÓN DE DATOS]
    POO --- ENCAPSULACIÓN[ENCAPSULACIÓN]
    POO --- POLIMORFISMO[POLIMORFISMO]
  
```

Ilustración 7. Descripción General

Bibliografía

Bjarne Stroustrup
El Lenguaje de programación C++ (The C++ programming language)
Addison-Wesley, 2002

Harvey M. Deitel, Paul J. Deitel
Como programar en C++ (C++: How to program), 7ed
Pearson Education, 2010

Ceballos, Francisco Javier
Programación Orientada a Objetos con C++
México, Ed. Alfaomega Ra+Ma, 2004

Ceballos, Francisco Javier
Enciclopedia del lenguaje C++
México, Ed. Alfaomega Ra+Ma, 2004

STL tutorial and reference guide
David R. Musser, Gilmer J. Derge, Atul Saini
Addison Wesley Programming Series

The Unified Modeling Language Reference Manual
James Rumbaugh, Ivar Jacobson, Grady Booch
Addison Wesley Professional, 2010

Metodología

En cada una de las secciones que componen el temario encontrará una sección de ejercicios de autoevaluación, el propósito de estos ejercicios es que el alumno por cuenta propia se autoevalúe para saber si comprendió el tema adecuadamente. Adicionalmente en la sección de programación en C++, se proponen a el alumno realizar pequeños programas que pueden ayudar a entender conceptos complejos, tales como la herencia, sobrecarga de operadores o polimorfismo.

Finalmente existen algunas practicas que implican la realización de programas de computadora mas complejos que posiblemente requieran varios de días para su implementación, en estos programas se puede solicitar al alumno ocupar conceptos basicos como avanzados de programación orientada a objetos.

Ilustración 8. Descripción General 2

En la Ilustración 9 se muestra el capítulo de El Paradigma de la POO, cabe mencionar que en algunos capítulos se usan tanto imágenes como videos para dar una explicación más clara de los conceptos que se abordan.

¿PARA QUE SIRVE UN PARADIGMA?



¿QUE ES LA POO?

Un paradigma se puede considerar como el conjunto de normas y patrones que se pueden ocupar para resolver un problema. Esto es muy ocupado en la programación de computadoras para la resolución de ciertos problemas, ya que normalmente se ocupan estos patrones en problemas similares.

El concepto de paradigma es muy extenso y se aplica para muchas áreas del conocimiento no solo de la ingeniería. Uno de los pioneros en definir lo que es un paradigma fue Thomas Kuhn, y dio la siguiente definición.

"Considero a los paradigmas como realizaciones científicas universalmente reconocidas que, durante cierto tiempo, proporcionan modelos de problemas y soluciones a una comunidad científica"

Para comprender bien lo que es un paradigma de forma general y como nace se muestra el siguiente video en el que se ilustra en esencia este concepto según la definición de paradigma de Thomas Kuhn.

como nace un paradigma

El primer sustituto participó con entusiasmo de la paliza al novato. Un tercero fue cambiado, y se repitió el hecho, lo volvieron a golpear. El cuarto y, finalmente, el quinto de los veteranos fue sustituido.

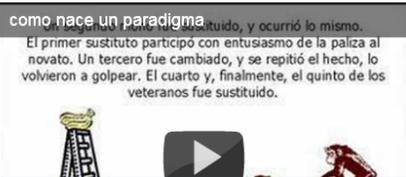


Ilustración 9. Paradigmas de Programación

En la Ilustración 10 e Ilustración 11, se muestra la sección de lecciones del material educativo de POO, en estas lecciones se describe lo que es un objeto y algunos ejemplos que se pueden representar con objetos, posteriormente se explican los mensajes que se pueden mandar entre los mismos objetos.

¿QUE ES UN OBJETO?

Un objeto se puede definir como la instancia de una clase en tiempo de ejecución y una clase a su vez se puede ver como el molde con el cual se puede crear un objeto. El término clase y los componentes de esta se ven a detalle más adelante.

En la POO, el contenedor o el molde de los objetos son las clases, en estas se especifican los métodos y variables miembro pertenecientes al objeto. A partir de una clase se puede crear objetos y dentro de estos se puede tener otros datos como enteros, caracteres, cadenas u otros objetos mucho más complejos.

Este modelo de encapsulación de datos es indispensable en la POO ya que se pueden crear datos complejos que contengan otros más complejos, y como todos estos objetos tienen una interfaz de comunicación no se necesita conocer su estructura ni su funcionamiento interno sino solo su interfaz. En la POO existen otros conceptos como herencia, polimorfismo, sobre carga de operadores; estos conceptos son importantes ya que permiten crear grandes sistemas flexibles, reutilizables y que respetan una arquitectura de sistema.

Como ya se mencionaba anteriormente, el concepto de clase y objeto son los puntos clave para entender lo que es la POO. Para entender el concepto de clase se va a ejemplificar con un carro, este carro tiene características como que es de cierto color, que tiene una marca, puede ser de 4 u 8 cilindros, etc; además tiene elementos como un volante, ruedas, motor, chasis, faros, radio, asientos, etc; y también realiza acciones como encender el motor, avanzar, retroceder, frenar, acelerar, etc. Todo esto puede ser representado en una clase, es decir, la clase puede contener todas las características, elementos y funciones que componen al carro o solo puede contener algunas de estas, esto depende de la funcionalidad que se le quiere dar a la clase.

La figura siguiente ilustra el objeto carro y algunas de las características de este ya mencionadas anteriormente.

OBJETO: CARRO



ATRIBUTOS	ENCAPSULADOS	FUNCIONES
Color: Verde Año: 2000 Trans.: Automática	Cauchos Chasis Motor	Encendido Aceleración Apegado, etc.

Ilustración 10. ¿Qué es un objeto?



Ilustración 11. Mensajes entre objetos

Finalmente en la Ilustración 12, Ilustración 13 e Ilustración 14; se muestran preguntas que forman parte de la primera sección “La Programación Orientada a Objetos y otros paradigmas”, las preguntas se dividieron en 2 que son las de opción múltiple y las preguntas que son abiertas, en el caso de la Ilustración 12 se muestra una pregunta de respuesta abierta. En total se agregaron 12 preguntas para esta sección para las otras secciones se agregaran preguntas y ejercicios de programación que permitan comprender adecuadamente el tema que se está revisando.

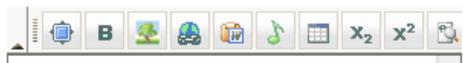
[Página principal](#) | [Mis cursos](#) | [Mi agenda](#) | [Informes](#) | [Red social](#) | [Panel de control](#) | [Administración de la plataforma](#)

Programación Orientada a Objetos > Ejercicios > Paradigmas de programación abiertas

[Volver a la lista de preguntas](#) | [Modificar ejercicio](#)

Paradigmas de programación abiertas

Pregunta 1 : Describa con sus palabras que es un paradigma de programación



[Validar respuesta\(s\)](#)

Ilustración 12. Preguntas abiertas

Página principal Mis cursos Mi agenda Informes Red social Panel de control Administración de la plataforma

Programación Orientada a Objetos > Ejercicios > Paradigmas de programación varias

[Volver a la lista de preguntas](#) [Modificar ejercicio](#)

Paradigmas de programación varias

Pregunta 1 : ¿Cuál de las siguientes sentencias define lo que permite un paradigma de programación?

- Definir un problema con base a lo que se plantea
- Resolver un problema utilizando patrones
- Entender el problema a fondo usando diferentes técnicas
- Encontrar la solución al problema planteado

[Siguiendo pregunta](#)

Ilustración 13. Preguntas de opción múltiple

Paradigmas de programación varias

Pregunta 3 : ¿Cuál de las siguientes sentencias expresa lo que es una interfaz en la POO?

- A través de una interfaz podemos saber de que clase hereda el objeto.
- Una interfaz proporciona una serie de métodos para poder interactuar con un objeto.
- Una interfaz nos permite que dos objetos puedan realizar funciones adicionales a las ya proporcionadas a través de una clase
- Una interfaz es un mecanismo mediante el cual nos comunicamos con otro objeto sin necesidad de conocer sus características internas.

[Siguiendo pregunta](#)

Ilustración 14. Preguntas varias

Cabe mencionar que existe una gran variedad de preguntas que se pueden agregar a un ejercicio, en la Ilustración 15 se muestra una imagen con los tipos de pregunta que existen. Algunas de las preguntas que ya se habían agregado se modificaron de acuerdo a las sugerencias que dieron las pedagogas.

Vista preliminar  Modificar ejercicio



Pregunta	Tipo
1. Describa con sus palabras que es un paradigma de programación	Respuesta abierta
2. Explique con sus propias palabras que es la Programación Orientada a Objetos	Respuesta abierta
3. Defina el concepto de abstracción aplicado en POO	Respuesta abierta
4. Defina con sus propias palabras qué es un componente	Respuesta abierta
5. Complete el espacio en blanco	Rellenar blancos

Ilustración 15. Tipos de preguntas



CAPITULO 5



5. CONCLUSIONES Y RECOMENDACIONES

En este último capítulo se presentan las conclusiones que se obtuvieron a lo largo del desarrollo de este caso práctico así como recomendaciones para mejorar el material didáctico de POO que se implementó.

Ya se mencionaba en secciones anteriores la existencia plataformas educativas virtuales algunas de estas Moodle, eCollege, WebCT, Doqueos, Claroline y Chamilo; esta última fue la que se escogió para este caso práctico. Estas plataformas virtuales educativas dan facilidades tanto para profesores en la transmisión de conocimientos como para alumnos en el aprendizaje de estos.

Estas LMS²⁷ o también conocidas como plataformas educativas virtuales proporcionan características que van desde poder crear lecciones con texto, video y audio, hasta poder realizar evaluaciones a los alumnos de manera virtual, además de poder obtener datos como cuánto tiempo paso el alumno en la plataforma virtual y que días accedió, etc.

En este caso práctico se implementó material didáctico de apoyo para la impartición de la materia de Programación Orientada a Objetos. La importancia de crear este material es la contribución que hace sobre el campo educativo en las áreas de ingenierías, más específicamente sobre la enseñanza de la materia de Programación Orientada a Objetos.

Es necesario recordar que en el capítulo 1 se hizo hincapié sobre el problema educativo que existe en las ingenierías y en la carrera de ingeniería en computación, donde se observó que una de las materias de altos índices de reprobación es esta materia de POO. De esta manera es como este material de apoyo pretende ser una la semilla para que se sigan utilizando las diferentes tecnologías en la educación y principalmente el uso de plataformas educativas virtuales no como sustituto del profesor sino como un complemento para las clases y como un medio de interacción entre el profesor y los alumnos que toman la materia.

Otro punto a considerar en la implementación de este caso práctico es el uso de la plataforma educativa virtual Chamilo, esto debido a que actualmente una de las plataformas educativas más utilizadas en la actualidad es Moodle y a pesar de esto se tuvieron varias razones para preferir usar Chamilo en lugar de Moodle, una de estas es la innovación de Chamilo, ya que es un proyecto nuevo y que actualmente está en desarrollo, considerando esto es aquí donde este trabajo sirve como otra fuente de información sobre la plataforma Chamilo

Además es necesario recalcar que Chamilo es una plataforma educativa innovadora que está retomando muchos problemas de proyectos similares como Moodle. Claroline, Dokeos y está ofreciendo soluciones a estos problemas. Un ejemplo de problema en el que Chamilo contribuyo es disminuir el nivel de complejidad que presentan la mayoría de plataformas educativas virtuales en la

²⁷ Por las siglas en ingles Learning Management Systems.

interfaz gráfica, esto debido a que en otras plataformas es muy complejo y dado que es usado por profesores y alumnos que no están tan familiarizados con las tecnologías aplicadas a la educación, les es muy difícil comprender y utilizar la interfaz gráfica de estas plataformas incluso llegan al punto donde es necesario tomar cursos de bastante tiempo sobre el manejo de la plataforma educativa.

Adicionalmente es necesario comentar que durante la realización de la presente investigación se destaca que se tuvo que utilizar los conocimientos de la carrera de ingeniería en computación tales como:

- a) Programación Orientada a Objetos
- b) Ingeniería en Software
- c) Aplicaciones de lenguajes de programación
- d) Teoría de compiladores
- e) Manejo de sistemas operativos Windows y GNU/Linux.

Considero que el presente caso práctico, me ha permitido ampliar mi experiencia profesional, ya que durante la realización del proyecto me enfrente con problemas con el desarrollo y viabilidad del proyecto. Además de adquirir la experiencia de trabajar con personas de otras disciplinas como pedagogía y diseño gráfico. Todo esto ha ampliado mi percepción sobre el desarrollo de proyectos y sobre la utilidad que tienen las TIC's en las diferentes áreas del conocimiento.

RECOMENDACIONES

Existen varias recomendaciones que se pueden hacer para mejorar el material de apoyo de la materia de POO que se desarrolló, una de ellas es implementar un modelo educativo bien definido sobre el material didáctico. La implementación de un modelo educativo se sugiere que sea en colaboración con un pedagogo. Entre los modelos educativos más usados se encuentra el constructivista, el conductista y el modelo sudbury.

Con respecto a la parte técnica del material implementado, se recomienda que en los temas que se vayan a implementar se agreguen ejercicios al final de la lección, ya que así el alumno reafirma sus conocimientos o práctica los conocimientos adquiridos. Es sumamente recomendable que estos ejercicios vayan incrementando en complejidad ya que así se le permitirá al alumno que se encuentre interesado en la materia especializarse e ir incrementado sus habilidades sobre ciertos temas de programación.

Otra recomendación sobre el material didáctico sería implementar temas adicionales que no se ven en el salón de clases, ya que esto permitirá al alumno que desee especializarse en el área de programación continuar con el estudio de técnicas avanzadas. Algunos de los temas que se sugieren para implementar son los siguientes: patrones de diseño, La STL de C++, buenas prácticas de programación en C++, Programación Genérica, entre otros.

También se recomienda que se agreguen temas relacionados con otros lenguajes de programación como Java o C#, ya que esto permitirá a el alumno darse cuenta de que la resolución de problemas a través de la programación es independiente del lenguaje que se utilice, ya que estos comparten técnicas muy similares y muchos aspectos de la sintaxis.

Sobre la plataforma educativa Chamilo se recomienda seguir muy de cerca la nueva versión (2.0) de Chamilo que próximamente se liberara, ya que esta trae características adicionales de las que tiene ahora y esto permitirá seguir innovando en el uso de las últimas tecnologías para el proceso de enseñanza – aprendizaje.

LIMITACIONES

Existen algunas limitaciones con respecto al uso de la plataforma educativa Chamilo, y sobre todo en la instalación de esta (ver el manual de instalación de Chamilo en el anexo). Es necesario que la instalación de la plataforma Chamilo requiera conocimientos técnicos sobre computación, sobre todo si se piensa instalar la plataforma educativa en un sistema operativo Linux. Es por esta razón que se decidió incluir en el anexo un manual de instalación en los sistemas operativos Windows y Linux. Debido a esto se recomienda realizar la instalación de Chamilo con el apoyo de un administrador de servidores.

Otra de las limitaciones de usar una plataforma educativa es en la cuestión de los costos para su implementación ya que se requiere un equipo de cómputo que funja como servidor para almacenar los cursos creados, así como también es necesario contar con una conexión de banda ancha para que los alumnos puedan acceder sin problemas a los cursos online. Para todo esto se sugiere que se le esté dando soporte técnico continuamente a la plataforma educativa, para verificar asuntos de seguridad y de almacenamiento en la PC servidor.

Para las dudas que puedan surgir sobre este trabajo y asuntos relacionados con la plataforma virtual educativa Chamilo puede referirse al siguiente correo:

Correo electrónico: proftigerx2011@gmail.com

APENDICE

GUÍA DE INSTALACIÓN DE CHAMILO EN UN SISTEMA OPERATIVO WINDOWS Y LINUX

La instalación de la plataforma educativa virtual Chamilo puede ser de 2 maneras. La primera es contratar un servicio que ya proporcione el alojamiento para Chamilo²⁸, por lo que un curso estaría en internet en cuanto el proveedor de acceso y se cargue el curso en la plataforma educativa. La principal ventaja de usar esta opción es que no se tiene que contar con el hardware como una PC que sirva como servidor y una conexión de banda ancha, esto debido a que el proveedor del servicio de almacenamiento ya lo proporciona, la única tarea como profesor sería crear los cursos en la plataforma de manera remota, para esto se debe tomar en cuenta que es necesario el pago por el servicio que el proveedor del servicio de alojamiento brinda.

El procedimiento para tener un curso virtual en la plataforma educativa Chamilo en un servidor de alojamiento externo como beeznest es muy sencillo, sólo es necesario realizar el pago por el servicio para obtener el acceso y seguir las instrucciones que da el proveedor del servicio para cargar el curso.

La segunda opción que se puede tomar en cuenta es hacer la instalación local, para esto se debe de contar con una PC que funja servidor (con suficiente almacenamiento) para poder guardar y dar servicio de conexión a los cursos virtuales que se tendrán alojados, además es necesario tener una conexión de banda ancha con IP fija y para la instalación de Chamilo es necesario contar con conocimientos básicos en la instalación y configuración de PHP, Apache y MySQL Server.

Para tener una idea del hardware que se necesita se va a tomar en cuenta los datos que Chamilo muestra en su página. Está página proporciona un servicio de

²⁸ Uno de los proveedores del servicio de alojamiento para Chamilo es BeezNest y la página web donde se puede contratar este servicio es: <http://www.beeznest.com/es/alojamiento/chamilo-light>

almacenamiento muy limitado para versiones de prueba y se da a conocer que actualmente <http://campus.chamilo.org> tiene las siguientes características en cuestiones de usuarios y cursos:²⁹

- 19,600 usuarios registrados
- 300 usuarios conectados simultáneamente (con un timeframe de 120 segundos)
- 3,500 cursos

El servicio que proporciona la página <http://campus.chamilo.org>, se encuentra alojado en un equipo con las siguientes características:

- Procesador 2 x Intel Xeon Dual-Core 5130
- 2GB de memoria RAM
- 250GB SATA HDD de almacenamiento en disco duro
- X7DBN Motherboard
- 5TB / ancho de banda mensual.

Por la información anterior es posible entender que una computadora actual podría funcionar como servidor siempre y cuando se cuente con un servicio de internet que proporcione la banda ancha necesaria para que los usuarios que se vayan a agregar al curso se puedan conectar remotamente y que tenga el suficiente almacenamiento en disco duro.

En el caso de la instalación de Chamilo de forma local se necesita una PC que sirva como servidor de datos y tener una conexión de banda ancha con una IP fija. A continuación se detallan los pasos generales que hay que seguir para instalar Chamilo localmente.

²⁹ Información obtenida de la página web <http://www.chamilo.org/es/documentacion>, consultada en Diciembre de 2012

INSTALACIÓN DEL SOFTWARE NECESARIO PARA CHAMILO EN WINDOWS

La Instalación se realizó en un sistema operativo Windows 7 de 64 bits en una laptop y además en la PC que funciona como servidor en la FES Aragón, esta última cuenta con un sistema operativo GNU/Linux Debian. Se instaló la versión 1.8 de Chamilo, esta versión de Chamilo se obtiene gratuitamente de la siguiente página web: <http://www.chamilo.org/es/download-es>

1.- El primer paso después de descargar Chamilo es tener instalado un servidor web y para esto se continuará con la descarga de este servidor web, se decidió optar por Apache Web Server, debido a que es uno de los servidores web más utilizados y porque es soportado por Chamilo. El link para la descarga de la versión 2.2.20 de Apache HTTP Server es el siguiente: <http://httpd.apache.org/download.cgi>

Se descargan el instalador de esta versión y se procede con la instalación, la primera ventana que se muestra al ejecutar el asistente de instalación es la de la Ilustración 16.

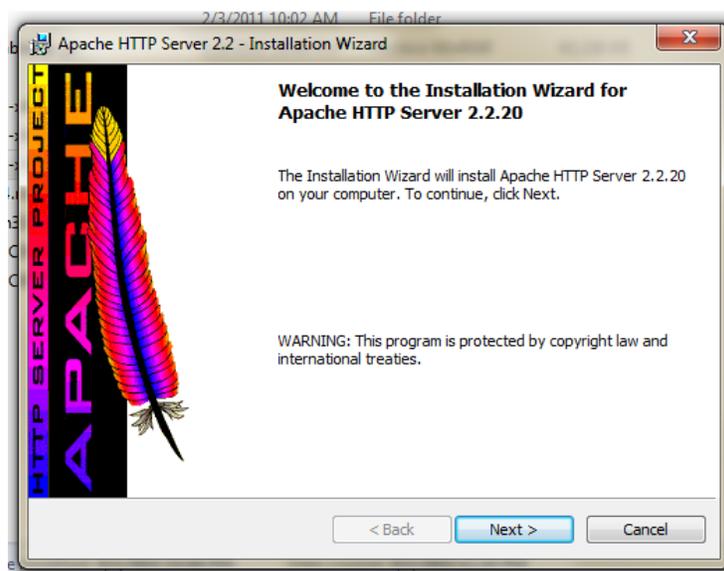


Ilustración 16

Posteriormente solicita datos como el dominio de la red y el nombre del equipo que funcionará como servidor, estos datos pueden ser obtenidos si se solicitan al

administrador del equipo en el que se va a instalar el servicio web. En este caso se dejaron los que ya tenía por default (ver Ilustración 17).

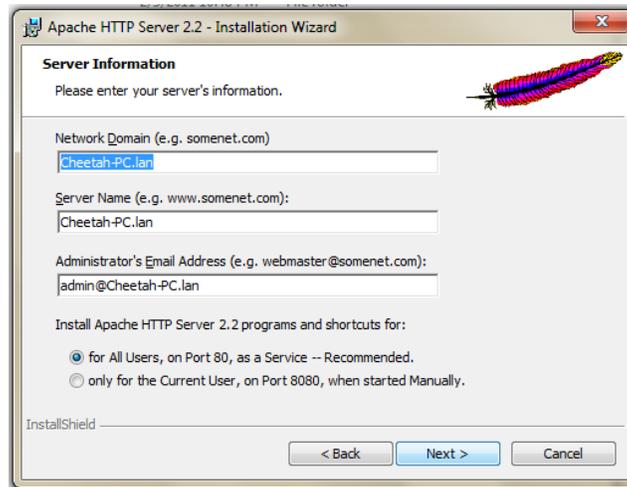


Ilustración 17

En la siguiente ventana vamos a elegir una instalación típica, si se quisiera instalar ciertos componentes en particular se tendría que elegir la personalizada.

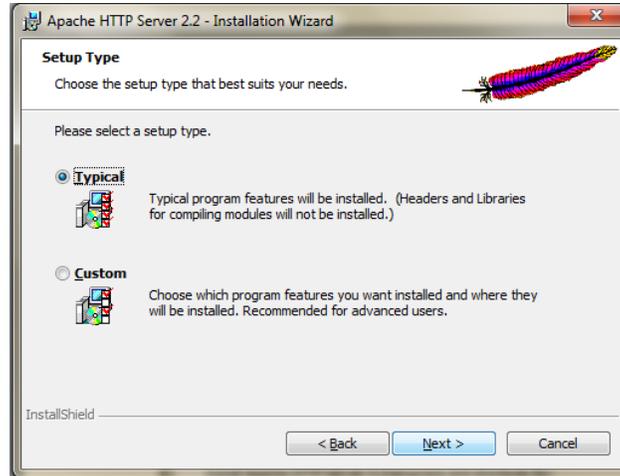


Ilustración 18

El siguiente paso es elegir donde se va a instalar Apache HTTP, normalmente esto lo hace en una ruta por default, pero en este caso se creará el folder Apache2.2 en la ruta "C:/", por lo tanto la ruta de instalación sería "C:\Apache2.2", después de elegir la ubicación en el instalador de Apache HTTP Server

presionamos siguiente y se mostrara la Ilustración 19 y posteriormente al presionar el botón Next la Ilustración 20

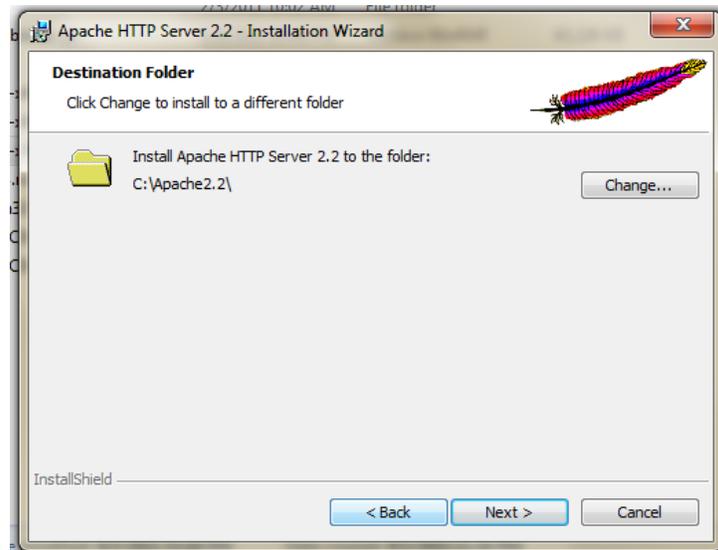


Ilustración 19

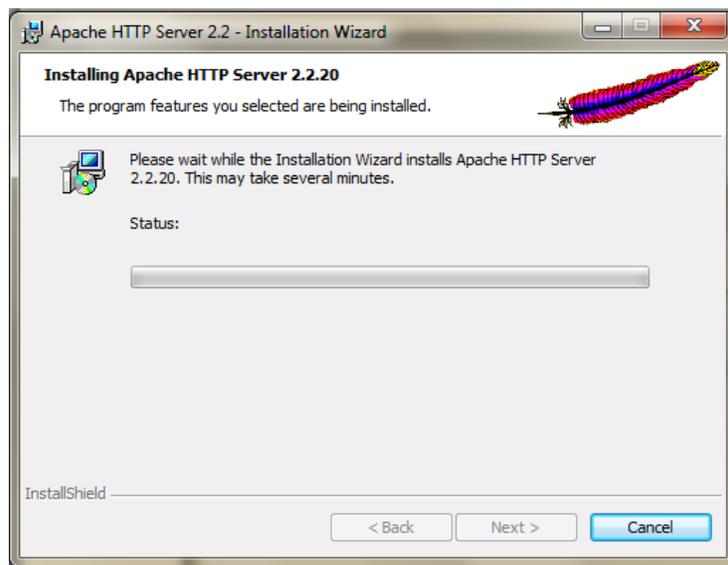


Ilustración 20

Finalmente se muestra la Ilustración 21 indicando que la instalación de Apache HTTP Server ha sido finalizada con éxito.

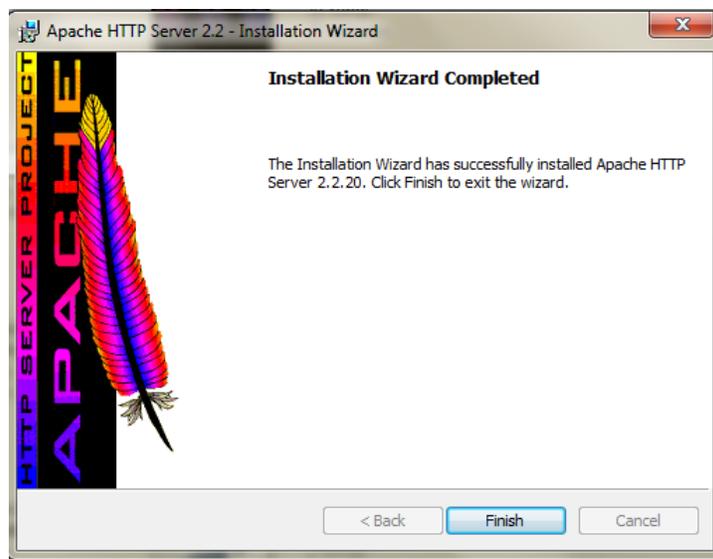


Ilustración 21

Al finalizar de instalar apache es necesario cambiar un parámetro dentro del archivo httpd.conf que se encuentra ubicado en la ruta donde se instaló apache (C:\Apache2.2\conf). Es necesario abrir el archivo con el block de notas y buscar la línea “*DirectoryIndex index.html*” y sustituirla por la siguiente “*DirectoryIndex index.php*”, esto es para que cuando Chamilo cargue una página busque el index.php, si no se hace esto Chamilo buscara una página html y la mostrara por lo que se tendrá que dar manualmente click en el link del archivo index.php. En la Ilustración 22 se muestra el archivo httpd.conf ubicado en la línea que hay que cambiar.

```

#
# Controls who can get stuff from this server.
#
order allow,deny
allow from all

</Directory>

#
# DirectoryIndex: sets the file that Apache will serve if a directory
# is requested.
#
<IfModule dir_module>
  DirectoryIndex index.html
</IfModule>

#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by web clients.
#
<FilesMatch "\.ht$">
  order allow,deny
  deny from all
  Satisfy All
</FilesMatch>

```

Ilustración 22

2.- El segundo paso antes de instalar Chamilo es la instalación de PHP, para esto se va a lanzar el instalador de PHP versión 5.3.4 que se descargó de la siguiente página <http://windows.php.net/download/>.

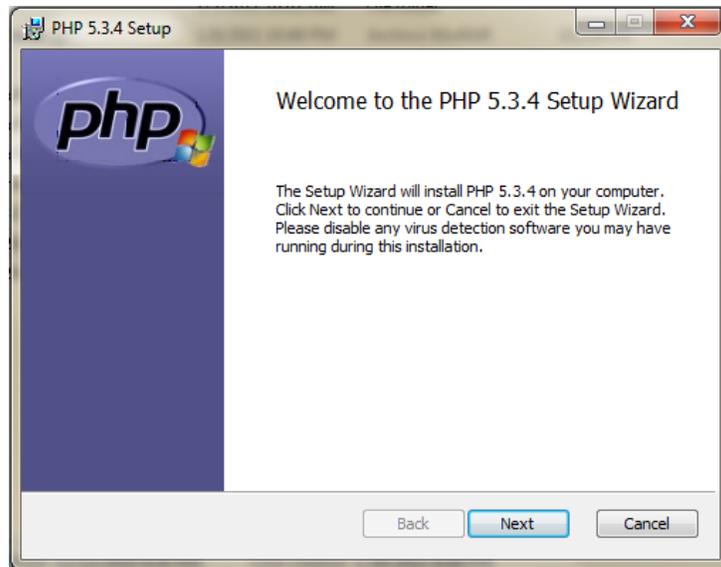


Ilustración 23

Al igual que como lo hicimos con apache vamos a crear el folder PHP en la ruta "C:/" por lo que la ruta donde se instalara PHP será "C:\PHP", esta ruta la escogemos en el wizard de instalación de PHP cuando lo solicite.

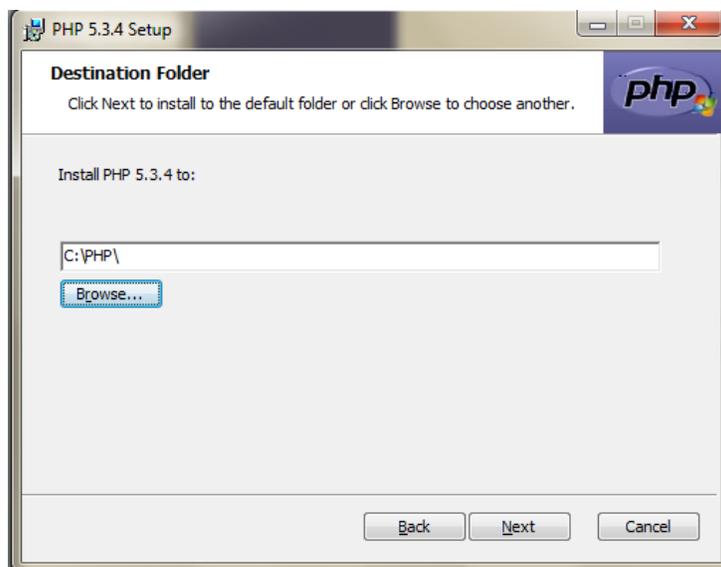


Ilustración 24

A la hora de estar instalando PHP pide un servidor web a establecer, en este caso como ya se instaló Apache se elegirá la opción: Apache 2.2.x Módulo.

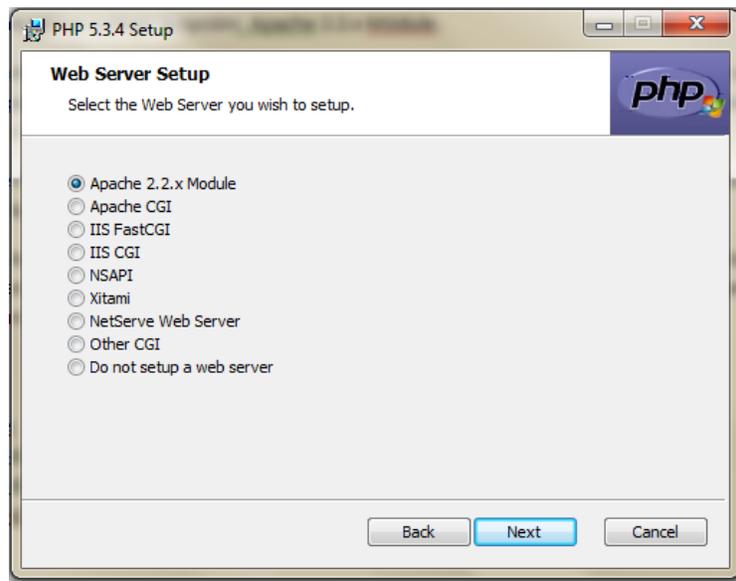


Ilustración 25

Después el instalador pide la ruta donde del directorio de configuración de Apache, por lo que se elegirá a través del wizard la ruta "C:\Apache2.2\conf", tal como se ve en la Ilustración 26.

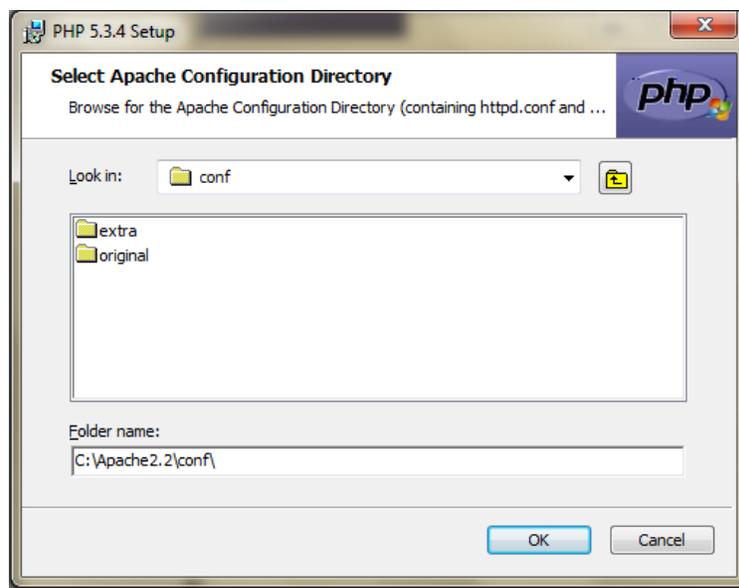


Ilustración 26

Después sólo es necesario dar click en siguiente para continuar con la pantalla de selección de ítems que se muestra en la Ilustración 27.

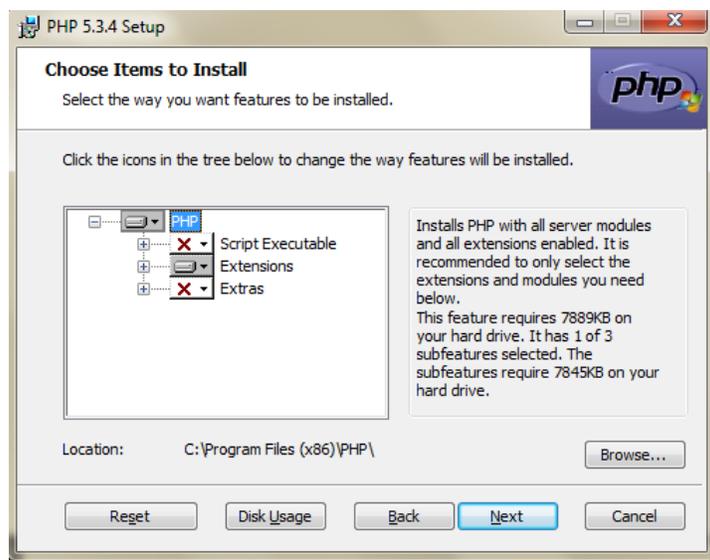


Ilustración 27

En este punto de la instalación de PHP es necesario activar las extensiones de php necesarias, para esto es necesario ir a la sección de extensión en la ventana de instalación y activamos los siguientes módulos (Ilustración 28):

- * gd
- * pcre
- * session
- * standard
- * mysql
- * zlib
- * xsl

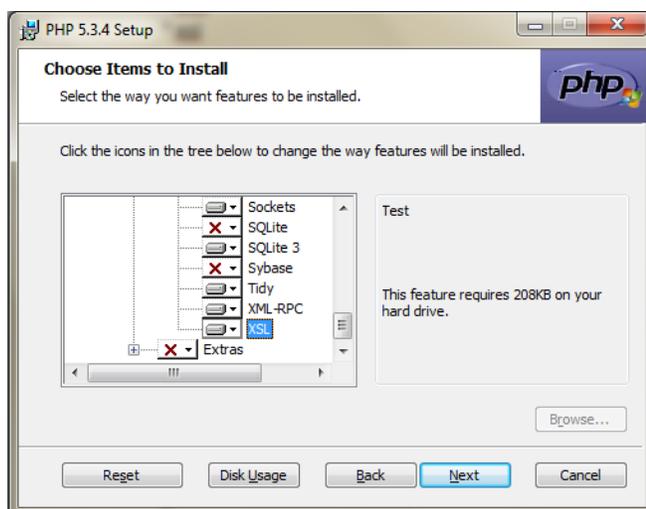


Ilustración 28

Una vez terminada la selección de ítems (puede ser que no todos estén en la lista pero es necesario activar los que estén disponibles) es necesario dar click en el

botón Next lo que el wizard avanzará a la pantalla mostrada en la Ilustración 29 y finalmente al dar click en Install nos mostrara la pantalla de la Ilustración 30, indicando que la instalación de php ha terminado.

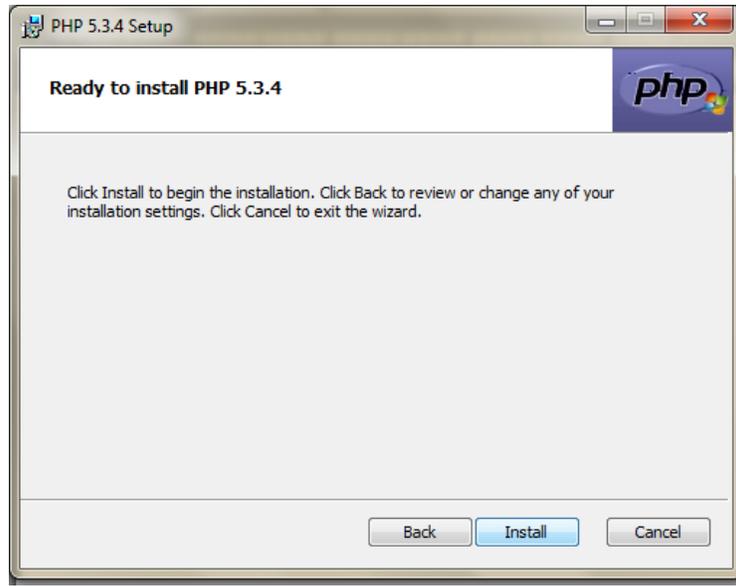


Ilustración 29

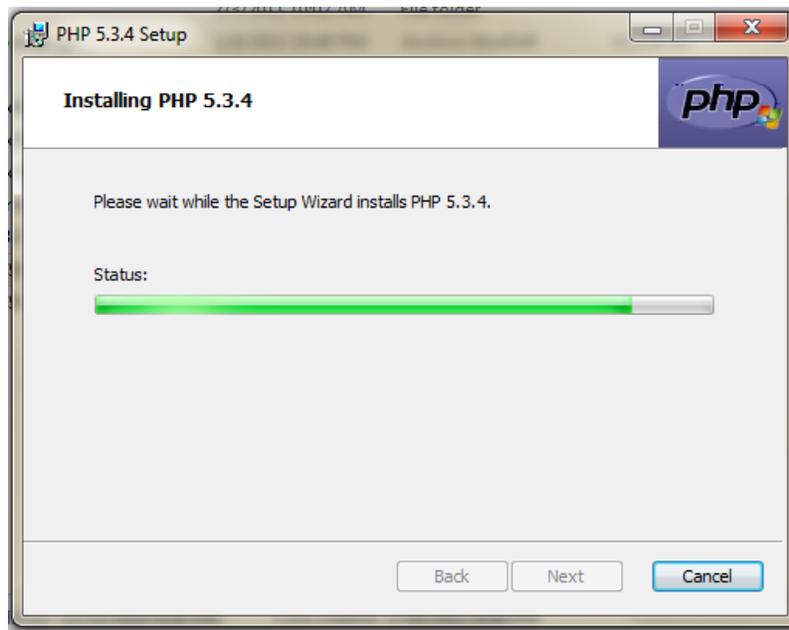
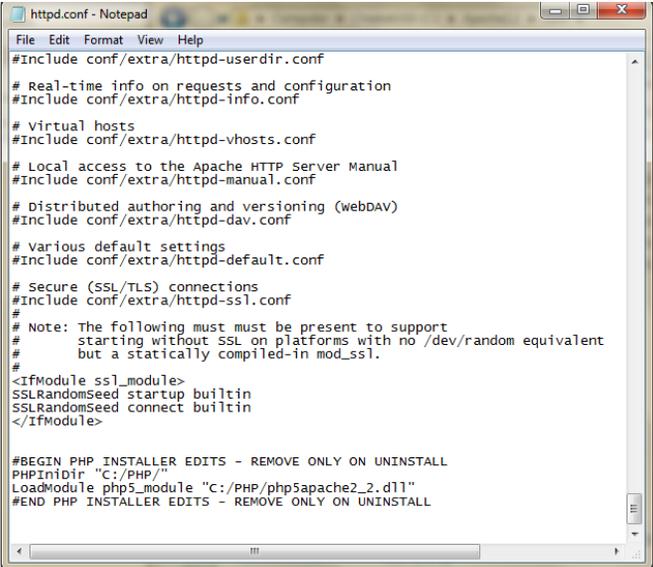


Ilustración 30

Puede ocurrir que php no edite los archivos de apache de manera adecuada, para esto es necesario ir al directorio donde se encuentra instalado Apache en este caso es "C:\Apache2.2\conf" y modificar el archivo "httpd.conf"(ubicado en este mismo directorio), al final del archivo se encuentran las líneas donde se carga php, para esto debemos hay que revisar que las rutas donde carga los módulos de php sean correctas.



```

httpd.conf - Notepad
File Edit Format View Help
#Include conf/extra/httpd-userdir.conf
# Real-time info on requests and configuration
#Include conf/extra/httpd-info.conf
# Virtual hosts
#Include conf/extra/httpd-vhosts.conf
# Local access to the Apache HTTP Server Manual
#Include conf/extra/httpd-manual.conf
# Distributed authoring and versioning (webDAV)
#Include conf/extra/httpd-dav.conf
# Various default settings
#Include conf/extra/httpd-default.conf
# Secure (SSL/TLS) connections
#Include conf/extra/httpd-ssl.conf
#
# Note: The following must must be present to support
# starting without SSL on platforms with no /dev/random equivalent
# but a statically compiled-in mod_ssl.
#
<IfModule ssl_module>
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
</IfModule>
#
#BEGIN PHP INSTALLER EDITS - REMOVE ONLY ON UNINSTALL
PHPIniDir "C:/PHP/"
LoadModule php5_module "C:/PHP/php5apache2_2.dll"
#END PHP INSTALLER EDITS - REMOVE ONLY ON UNINSTALL

```

Ilustración 31.

En caso de que estuviera mal las líneas (ya sea porque la ruta o la .dll no existe) donde se carga Apache es necesario reemplazarlas por lo siguiente:

```
LoadModule php5_module "c:/php/php5apache2.dll"
```

```
# Configurar la ruta de php.ini
```

```
PHPIniDir "C:/php"
```

Después de realizar este cambio sobre el archivo de configuración de Apache es necesario volver a reiniciar el servidor web Apache, esto se hace en el monitor de Apache que normalmente se coloca en la barra de tareas de Windows (Ilustración 32).

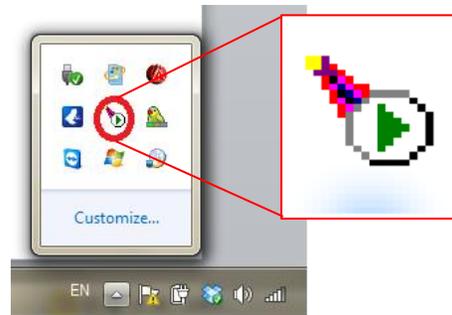


Ilustración 32

3.- El tercer paso es la instalación del programa mySQL server.

Para esto es necesario descargar el instalador de la página web de mySQL server, en este caso se descargó el instalador para Windows de 32 bits ya que son las características del hardware donde se realizó la instalación de prueba, más específicamente se descargó la versión 5.5. El link del instalador es el siguiente: <http://dev.mysql.com/downloads/mysql/>, al ejecutar el instalador nos mostrara las siguientes pantallas (Ilustración 33 e Ilustración 34), aquí es necesario seleccionar instalación típica.



Ilustración 33

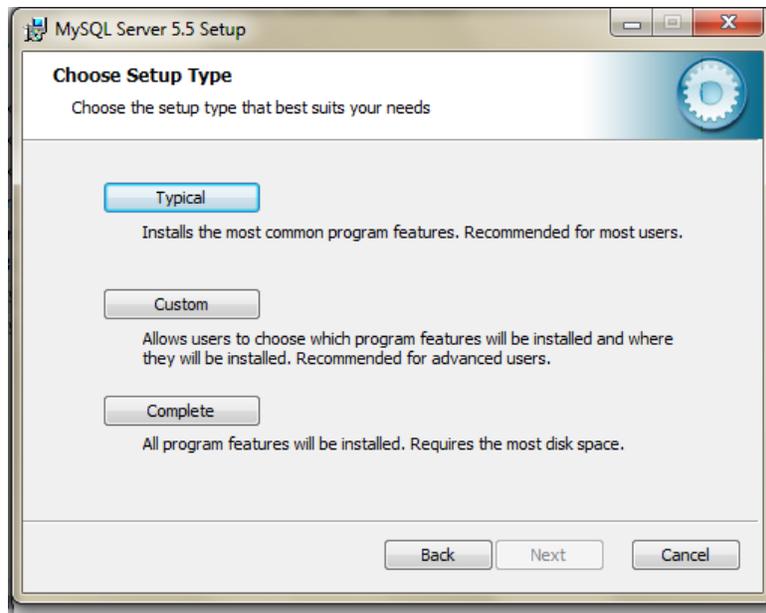


Ilustración 34

Posteriormente se mostraran las pantallas de la Ilustración 35 e Ilustración 36, en estas pantallas sólo se deberá dar click en Install y en Next.

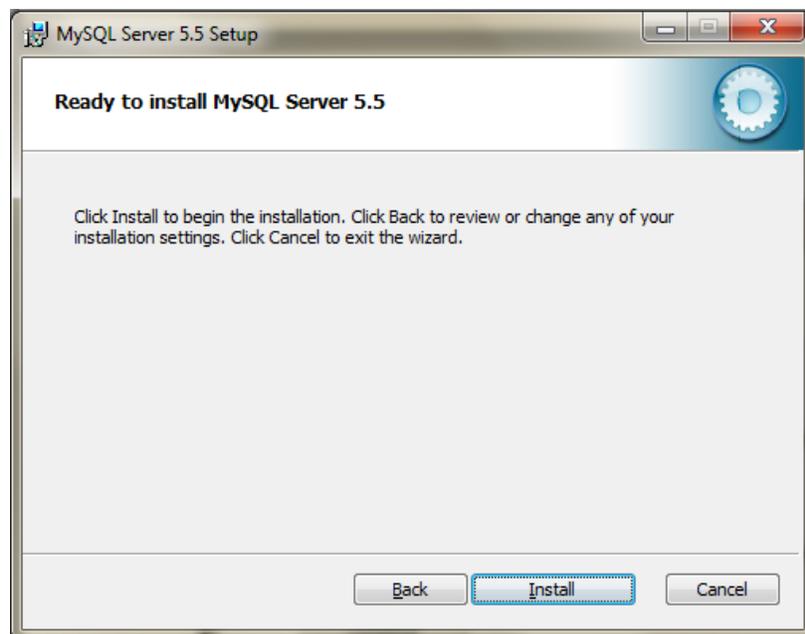


Ilustración 35

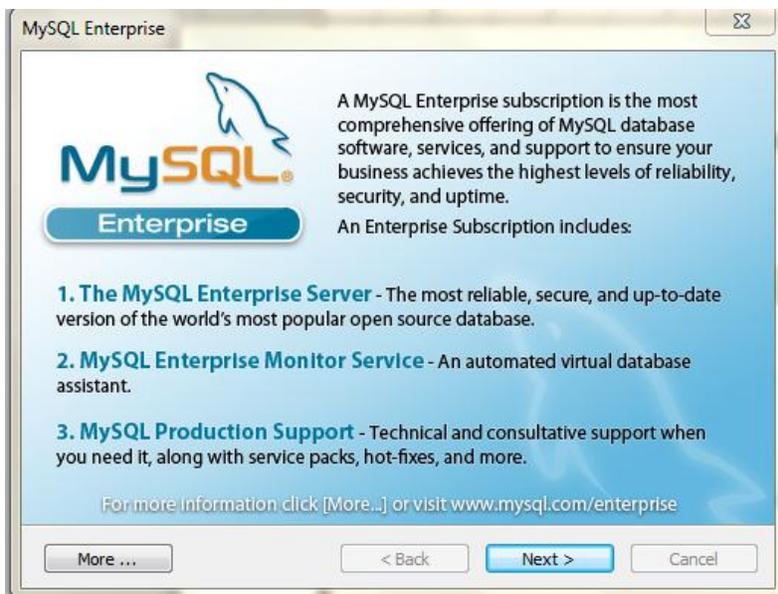


Ilustración 36

Finalmente aparecerá la pantalla de que se ha finalizado el wizard de instalación de MySQL Server.

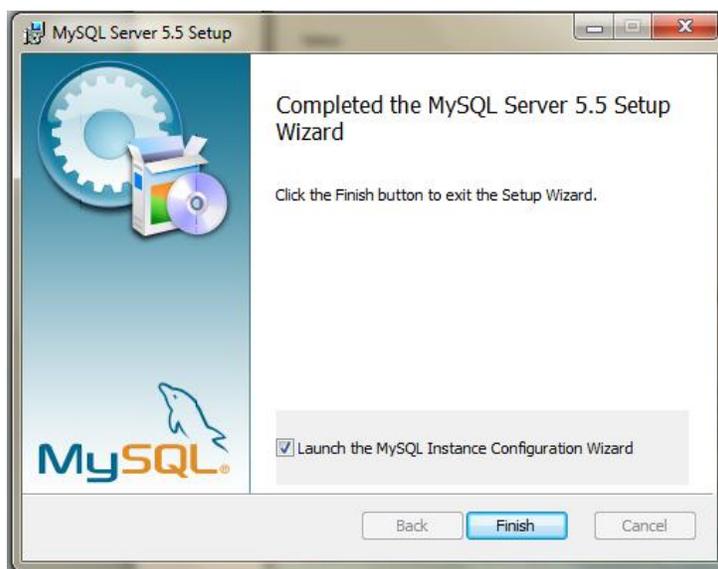


Ilustración 37

Una vez instalado MySQL el wizard mostrara otra pantalla de configuración como en la Ilustración 38, en esta pantalla es necesario click en el botón Next y mostrara la pantalla de la Ilustración 39.



Ilustración 38

En el transcurso de la configuración de MySQL Server, es necesario seleccionar la opción "*Detailed Configuration*" tal y como se muestra en la Ilustración 39 y posteriormente elegir "*Server Machine*", es decir, que la máquina va a funcionar como si fuera un servidor (Ilustración 40)

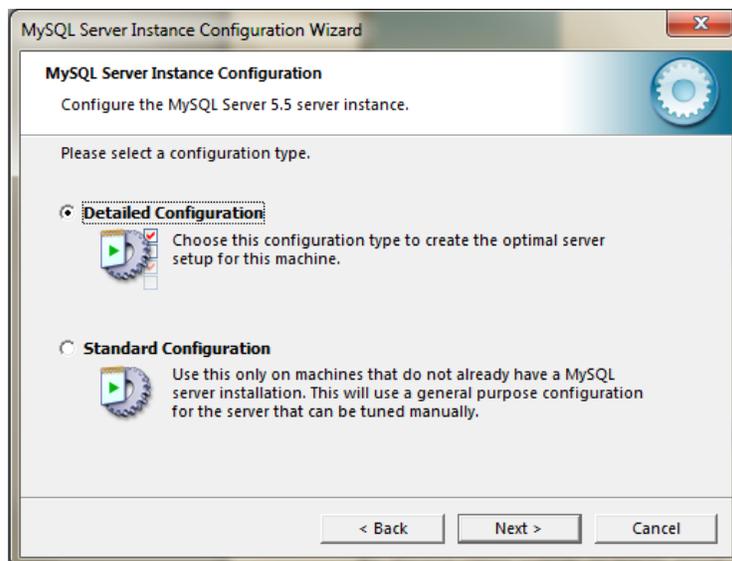


Ilustración 39

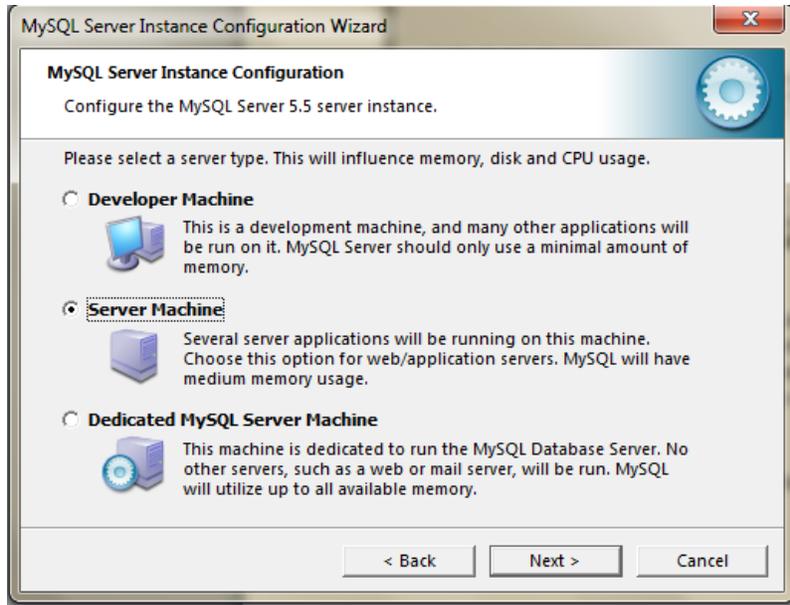


Ilustración 40

Después se escogerá el uso que se le dará a la base de datos, en este caso es necesario elegir la opción “Non-Transactional Database Only” (Ilustración 41), después de elegir esta opción pedirá una opción para elegir la cantidad aproximada de usuarios que accederán a la base de datos, para esto se elegirá la opción “Decision Support” (Ilustración 42).

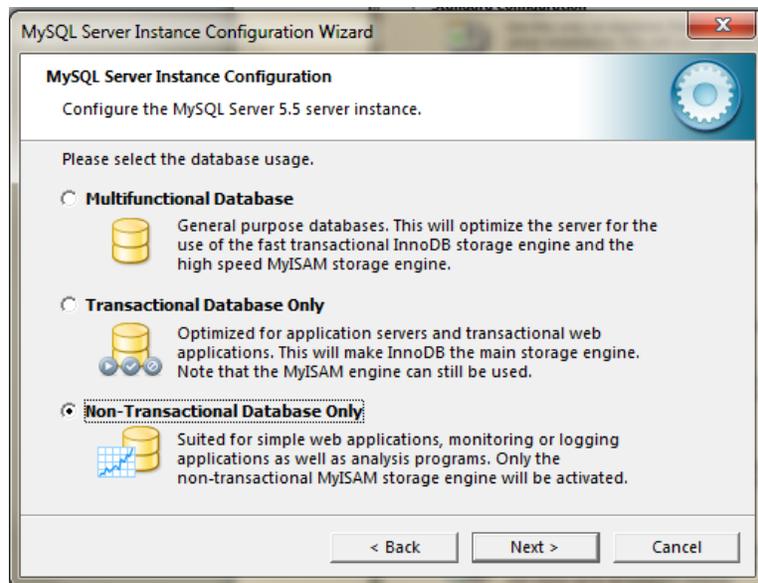


Ilustración 41

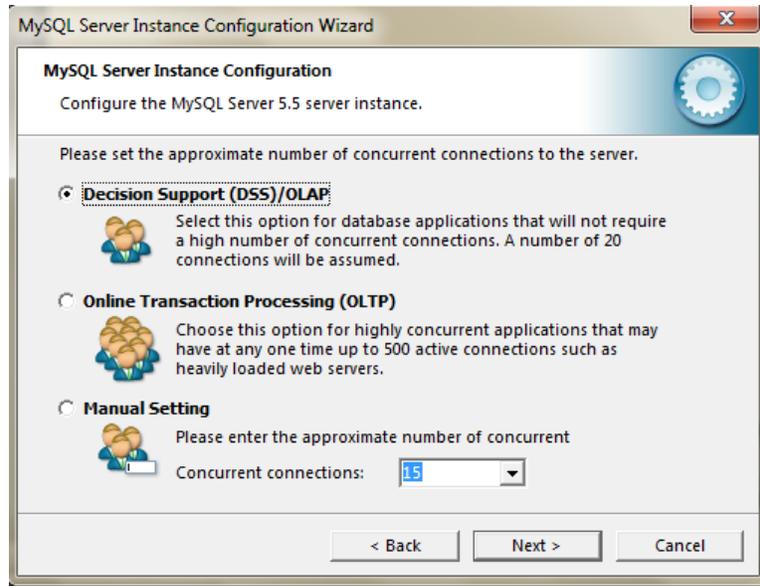


Ilustración 42

Las siguientes dos pantallas se mantendrán con las opciones que vienen por default, es decir, que este activada la opción “Enable TCP/IP Networking” , “Enable Strict Mode” y “Standard Character Set”, tal y como se ve en la Ilustración 43 y en la Ilustración 44.

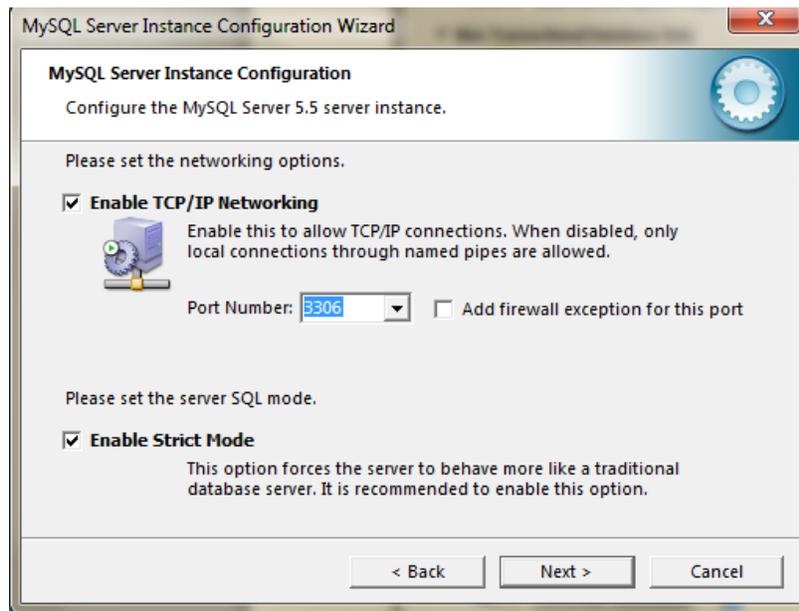


Ilustración 43

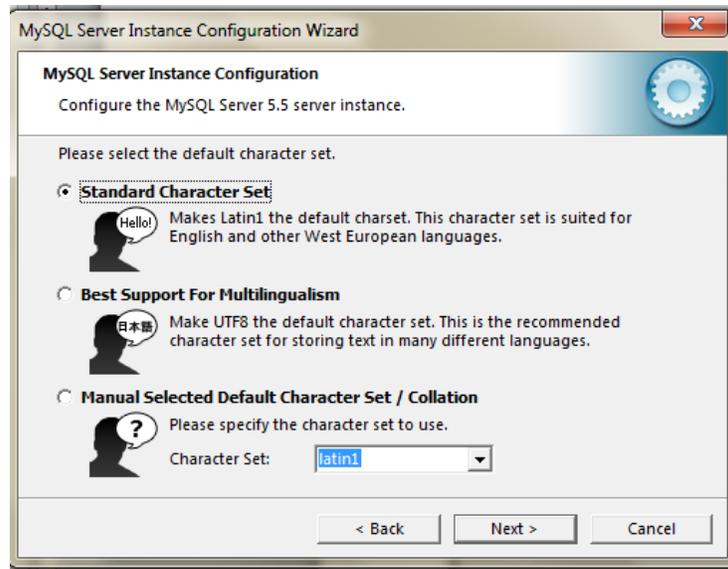


Ilustración 44

En las siguientes pantallas del wizard se solicita que configurar las opciones de Windows, para esto se tendrá que colocarlas tal y como se muestra en la Ilustración 45. En la siguiente pantalla se tendrá que establecer la contraseña para poder acceder como root al servidor MySQL, se muestra una pantalla como la de la Ilustración 46.



Ilustración 45



Ilustración 46

Finalmente ya que configuradas las opciones de MySQL Server es necesario ejecutar el proceso para cargar la configuración ya establecida, para esto se muestra la pantalla de la Ilustración 47, donde sólo se debe dar click en el botón *Execute*. Si todo se ejecutó adecuadamente se mostrara la pantalla de la Ilustración 48 indicando que la ha terminado exitosamente.

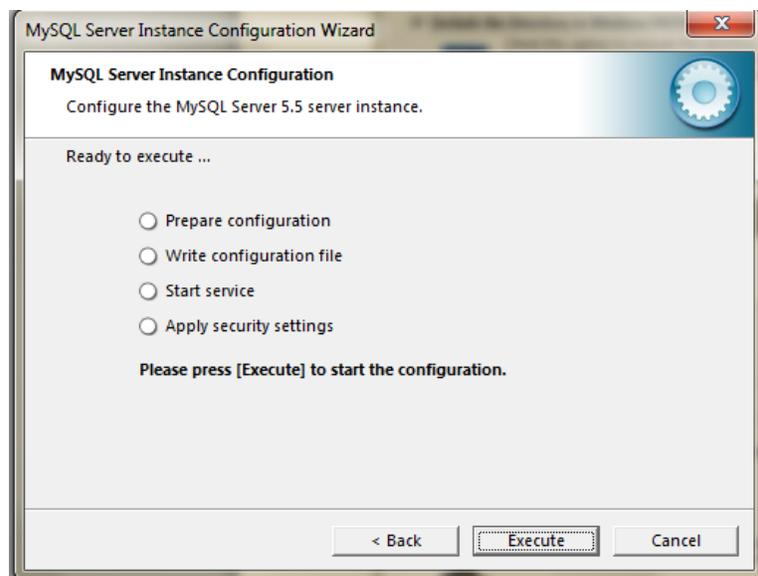


Ilustración 47

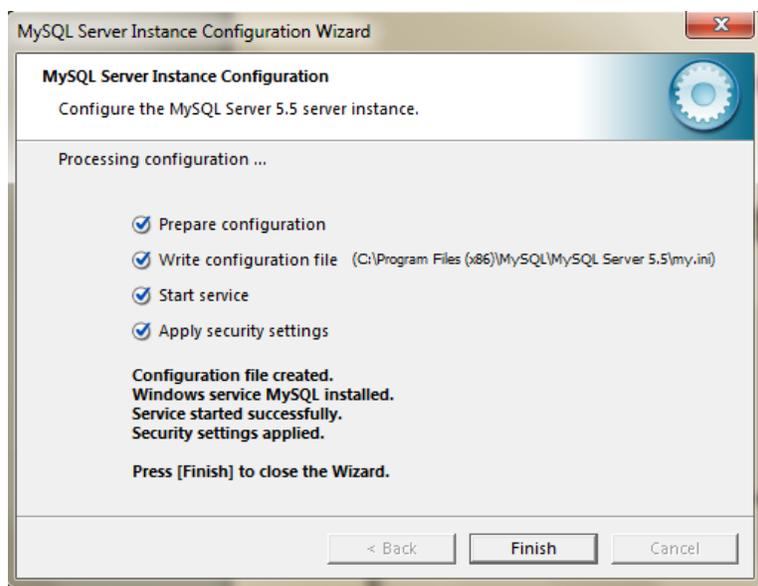


Ilustración 48

Para revisar que MySQL se está ejecutando y que es posible acceder al servicio es necesario abrir una consola de comandos de ms-dos (dar click en inicio, ejecutar y teclear “cmd”) ya abierta la consola de comandos teclear el siguiente comando:

```
C:/ > mysql -u root -p
```

Este comando indica acceder al servicio de mysql con usuario root y que solicite el password. Es aquí donde es necesario ingresar el password que se establecio en la pantalla de la Ilustración 46, si se logra acceder entonces MySQL está instalado adecuadamente, de lo contrario será necesario reinstalar MySQL y asegurarse de la contraseña que se definió durante la instalación.

INSTALACIÓN DEL SOFTWARE NECESARIO EN GNU/LINUX

Para la instalación de Apache, PHP y MySQL en GNU/Linux es necesario descargar los instaladores para este sistema operativo, los links para descargar el software son los mismos que para Windows sólo habría que elegir el código fuente que tiene extensión .gz o .bz y que sea para el sistema operativo GNU/Linux. Una

vez teniendo el código fuente es necesario compilar e instalar los ejecutables en el sistema, para esto se realizan varios pasos:

Para compilar e instalar Apache se ejecutaran los siguientes comandos

```
./configure --prefix=/usr/local/apache \  
    --enable-so \  
    --enable-cgi \  
    --enable-info \  
    --enable-rewrite \  
    --enable-speling \  
    --enable-usertrack \  
    --enable-deflate \  
    --enable-ssl \  
    --enable-mime-magic
```

make

make install

Para compilar e instalar PHP

```
./configure \  
    --with-apxs2=/usr/local/apache/bin/apxs \  
    --with-mysql \  
    --prefix=/usr/local/apache/php \  
    --with-config-file-path=/usr/local/apache/php \  
    --enable-force-cgi-redirect \  
    --disable-cgi \  
    --with-zlib \  
    --with-gettext \  
    --with-gdbm
```

make

make install

Para compilar e instalar MySQL

```
./configure --witout-debug --prefix=/usr/local/mysql
```

make

make install

Para definir las extensiones de los scripts PHP, hay que añadir las siguientes líneas en el fichero de configuración de apache (httpd.conf):

```
AddType application/x-httpd-php3 .php3
```

```
AddType application/x-httpd-php .php
```

```
AddType application/x-httpd-php3 .phtml
```

Ahora ya sólo queda arrancar el servidor, pero primero es necesario copiar el script de arranque en */etc/rc.d/init.d*

```
cp /usr/local/apache/bin/apachectl /etc/rc.d/init.d/apache
```

```
/etc/rc.d/init.d/apache start
```

INSTALACIÓN DE CHAMILO DESDE EL EXPLORADOR.

(YA SEA EN WINDOWS O GNU/LINUX).

El primer paso para instalar Chamilo es descargar la versión 1.8 de Chamilo de la siguiente página web: <http://www.chamilo.org/es/download-es>

Posteriormente descomprimir el archivo zip de Chamilo que se descargó en la carpeta que “htdocs” que se encuentra en el directorio donde se instaló Apache (C:\Apache2.2\htdocs).

Por lo tanto la carpeta de Chamilo se encontraría descomprimida en la siguiente ruta (C:\Apache2.2\htdocs\chamilo-1.8.7.1-stable). Es necesario dar permisos de escritura, lectura y ejecución a los siguientes directorios:

- [chamilo]/main/inc/conf/
- [chamilo]/main/upload/users/
- [chamilo]/main/default_course_document/images/
- [chamilo]/archive/

- [chamilo]/courses/
- [chamilo]/home/

En los puntos anteriores debemos sustituir “chamilo” por el nombre de la carpeta que se asignó en la instalación, en este caso “chamilo-1.8.7.1-stable”.

En Windows para establecer estos permisos se tiene que dar click derecho sobre la carpeta y asignar permisos de lectura, ejecución y escritura. La siguiente Ilustración 49 muestra la ventana donde se asignan permisos en el sistema operativo Microsoft Windows 7.

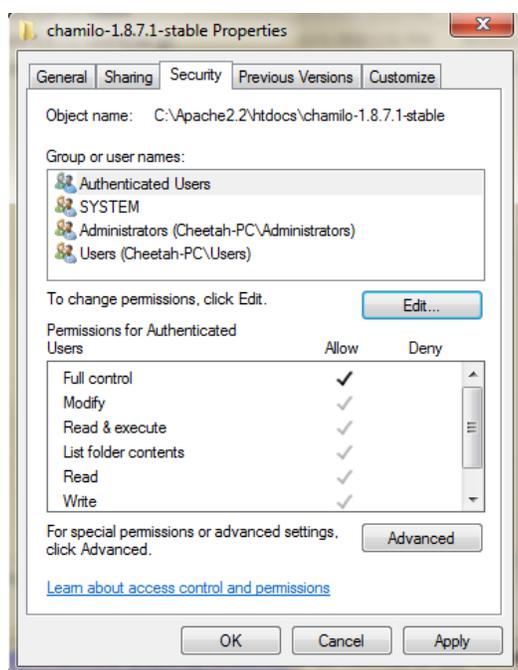


Ilustración 49

Posteriormente vamos es necesario abrir un navegador web y colocar la siguiente dirección. (<http://localhost/chamilo-1.8.7.1-stable/>), el nombre de esta ruta dependerá del nombre que haya colocado a la carpeta, en este caso se llama “chamilo-1.8.7.1-stable”.

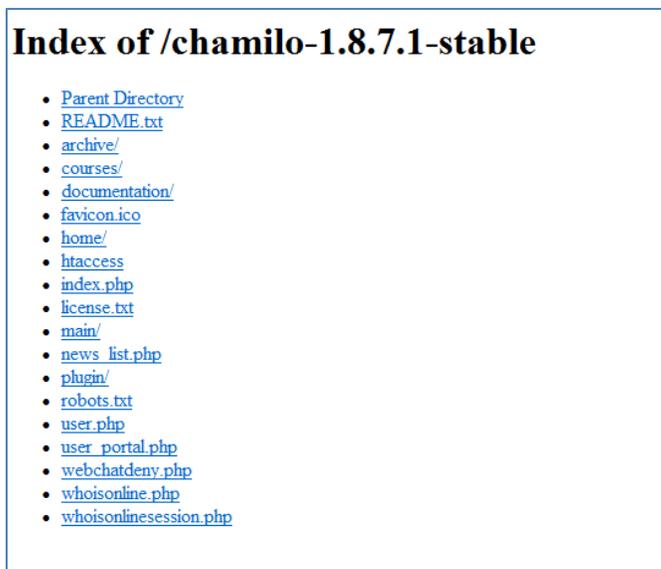


Ilustración 50

El primer paso es dar click en el link de “index.php” como se ve en la Ilustración 50 (Si ya se configuro Apache para que cargara el archivo .php tal y como se indicó anteriormente en la instalación de Apache debería direccionar a esta página automáticamente) y mandara a la ventana donde se muestra el botón para instalar Chamilo.

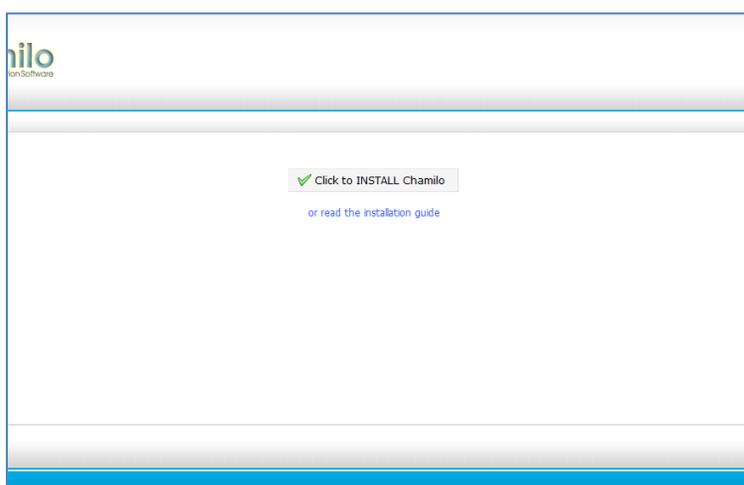


Ilustración 51

Al dar click en el botón para instalar mostrara paso a paso para establecer la configuración de Chamilo, en la Ilustración 52 muestra el paso 1 donde se establece el lenguaje de instalación.



Ilustración 52

Después se muestra la pantalla de los requisitos para instalar Chamilo, en caso de que faltara alguna opción de configuración de php o de otro software instalado, se colocaría de color rojo e indicaría que es necesario activarla.



Ilustración 53

El siguiente paso es dar click en el botón de nueva instalación y mostrara la pantalla para aceptar la licencia, tal y como se muestra en la Ilustración 54.

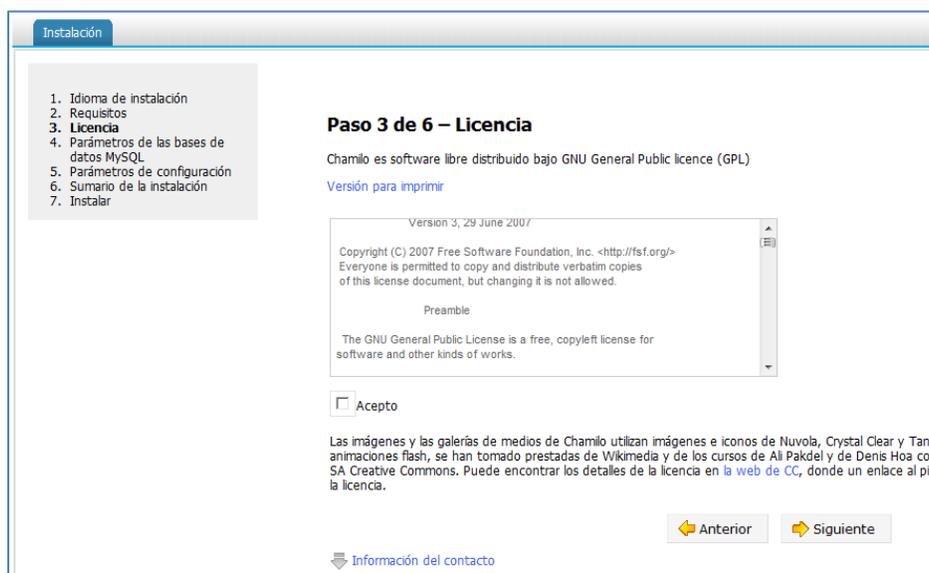


Ilustración 54

Es necesario recordar que cuando se instaló MySQL solicito una contraseña y después una confirmación, es en este punto donde debemos colocar la contraseña (Ilustración 55).

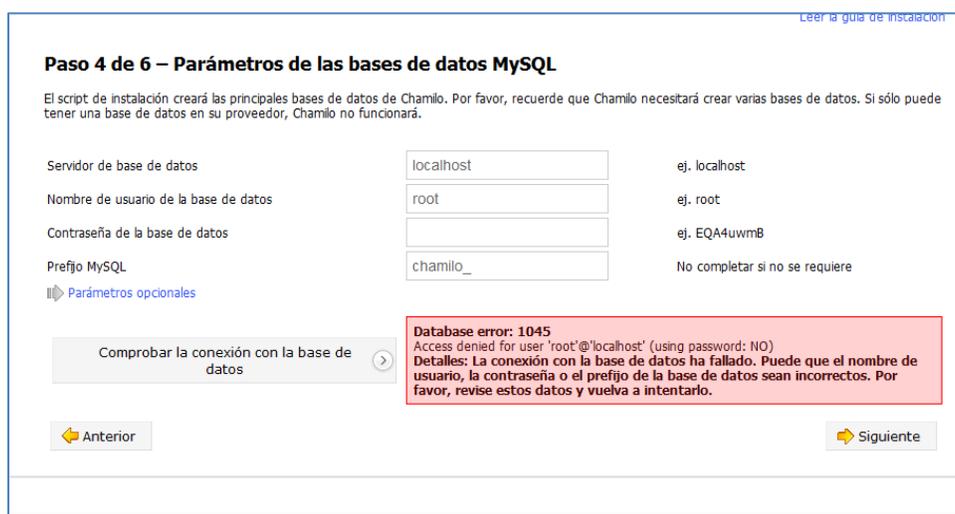


Ilustración 55

Una vez ingresado el password correcto muestra la siguiente pantalla de la Ilustración 56, es necesario anotar estos datos de esta pantalla por si son necesarios más adelante. En este ejemplo como se instaló localmente y se está usando el usuario root por default no se debe de cambiar nada, pero se podrían cambiar estos parámetros cuando sea necesario.

[Leer la guía de instalación](#)

Paso 4 de 6 – Parámetros de las bases de datos MySQL

El script de instalación creará las principales bases de datos de Chamilo. Por favor, recuerde que Chamilo necesitará crear varias bases de datos. Si sólo puede tener una base de datos en su proveedor, Chamilo no funcionará.

Servidor de base de datos	<input type="text" value="localhost"/>	ej. localhost
Nombre de usuario de la base de datos	<input type="text" value="root"/>	ej. root
Contraseña de la base de datos	<input type="password" value="*****"/>	ej. NXJUK8kJ
Prefijo MySQL	<input type="text" value="chamilo_"/>	No completar si no se requiere

[Parámetros opcionales](#)

Database host info: localhost via TCP/IP
 Database server version: 5.5.15
 Database client version: mysqlnd 5.0.7-dev - 091210 - \$Revision: 304625 \$
 Database protocol version: 10

Ilustración 56

Posteriormente el instalador mostrara la pantalla de parámetros de configuración (Ilustración 57), es muy importante anotar en algún lugar el password del administrador de la plataforma, incluso hasta se podría tomar una captura de pantalla y guardar la imagen como referencia.

Paso 5 de 6 – Parámetros de configuración

Los siguientes valores se grabarán en su archivo de configuración `main/inc/conf/configuration.php`: `main/inc/conf/configuration.php`

Idioma principal	<input type="text" value="Spanish"/>
URL de Chamilo (Contenido obligatorio)	<input type="text" value="http://localhost/chamilo-1.8.7.1-stable/"/>
E-mail del administrador	<input type="text" value="admin@Cheetah-PC.ian"/>
Nombre del administrador	<input type="text" value="John"/>
Apellidos del administrador	<input type="text" value="Doe"/>
Teléfono del administrador	<input type="text" value="(000) 001 02 03"/>
Nombre de usuario del administrador	<input type="text" value="admin"/>
Contraseña del administrador (puede que desee cambiarla)	<input type="text" value="hc5hdjzm"/>
Nombre de su plataforma	<input type="text" value="My campus"/>
Acrónimo de la organización	<input type="text" value="My Organisation"/>
URL de la organización	<input type="text" value="http://www.chamilo.org"/>
Método de encriptación :	<input checked="" type="radio"/> md5 <input type="radio"/> sha1 <input type="radio"/> Nada
Permitir que los propios usuarios puedan registrarse :	<input checked="" type="radio"/> Sí Recomendado <input type="radio"/> No
Permitir que los propios usuarios puedan registrarse como creadores de cursos :	<input checked="" type="radio"/> Sí <input type="radio"/> No

Ilustración 57

Más adelante en el paso 6 muestra la pantalla de comprobación antes de instalar como en la Ilustración 58, en ella se verifica que los datos y opciones con que se va a instalar Chamilo están correctas, finalmente dar click en Instalar Chamilo.

Paso 6 de 6 – Última comprobación antes de instalar

Éstos son los valores que ha introducido
Imprima esta página para recordar su contraseña y otras configuraciones

Idioma principal : spanish

Servidor de base de datos : localhost
Nombre de usuario de la base de datos : root
Contraseña de la base de datos : *****
Prefijo MySQL : chamilo
Base de datos principal de Chamilo (BD) : **chamilo_main** (lea la advertencia inferior)
Base de datos de seguimiento. Úsela sólo si hay varias bases de datos. : **chamilo_main** (lea la advertencia inferior)
Base de datos de usuarios : **chamilo_main** (lea la advertencia inferior)
Permitir seguimiento : Sí
Usar Chamilo con una o varias bases de datos : Varios

Permitir que los propios usuarios puedan registrarse : Sí
Método de encriptación : md5

E-mail del administrador : admin@Cheetah-PC.lan
Nombre del administrador : John
Apellidos del administrador : Doe
Teléfono del administrador : (000) 001 02 03
Nombre de usuario del administrador : **admin**
Contraseña del administrador (puede que desee cambiarla) : **hc5hdjzm**

Nombre de su plataforma : My campus
Acrónimo de la organización : My Organisation
URL de la organización : http://www.chamilo.org
URL de Chamilo : http://localhost/chamilo-1.8.7.1-stable/

Atención !
El programa de instalación borrará todas las tablas de las bases de datos seleccionadas. Le recomendamos encarecidamente que realice una copia de seguridad completa de todas ellas antes de confirmar este último paso de la instalación.

[← Anterior](#)

[✓ Instalar Chamilo](#)

Ilustración 58

Finalmente aparecerá la pantalla de parámetros de configuración en la cual se confirma que ya se creó la plataforma y proporciona el link para ir a la plataforma principal de Chamilo (véase la

[Leer la guía de instalación](#)

Paso – Parámetros de configuración

Cuando entra en su plataforma por primera vez, la mejor manera de entenderla es registrarse con la opción 'Profesor (crear un curso)' y seguir las instrucciones.

Aviso de seguridad: Para proteger su sitio, configure main/inc/conf/configuration.php y main/install/index.php como archivos de sólo lectura (CHMOD 444).

[Plataforma que acaba de crear.](#)

Ilustración 59).

[Leer la guía de instalación](#)

Paso – Parámetros de configuración

Cuando entra en su plataforma por primera vez, la mejor manera de entenderla es registrarse con la opción 'Profesor (crear un curso)' y seguir las instrucciones.

Aviso de seguridad: Para proteger su sitio, configure main/inc/conf/configuration.php y main/install/index.php como archivos de sólo lectura (CHMOD 444).

[Plataforma que acaba de crear.](#)

Link a la página principal

Ilustración 59

Finalmente si todo salió correcto, al dar click en el vínculo de “Ir a la plataforma que acaba de crear” nos llevara a la pantalla principal de Chamilo, donde se solicitara el nombre de usuario y password para poder acceder, esto se puede observar en la Ilustración 60.

Ilustración 60

Este proceso sería el que se debería seguir para la instalación de Chamilo, pueden surgir algunos problemas con respecto a la instalación pero esto depende del tipo de máquina en el que se esté instalando y la configuración que en ese momento se tenga.

BIBLIOGRAFÍA

Andy Ju An Wang, K. Q. (2005). *COMPONENT-ORIENTED PROGRAMMING*. WILEY – INTERSCIENCE, A John Wiley & SONS.

BELTRAN LLERA, J. A. (2003). *De la Pedagogía de la Memoria a la Pedagogía de la Imaginación*. FUNDACIÓN ENCUENTRO.

FES Aragón, U. (2011). *CUADRO COMPARATIVO DE LAS MATERIAS CON ALTO INDICE DE REPROBACIÓN 2011-1*.

FES Aragón, U. (2011). *SEGUIMIENTO A MATERIAS CON ALTOS ÍNDICES DE REPROBACIÓN*.

Grady Booch, J. R. (2006). *El lenguaje unificado de modelado: Manual de referencia*. Addison-Wesley.

Harvey M. Deitel, P. J. (2008). *C++. Cómo programar*. Pearson, Prentice Hall.

Margaret A. Ellis, B. S. (1994). *C++ manual de referencia con anotaciones*. Addison-Wesley.

TRANSITO, A. J. (2009). *TUTOR.NET: UNA IMPLEMENTACIÓN DE LA PLATAFORMA MOODLE SOBRE UN REQUERIMIENTO REAL EN EL COLEGIO DE CIENCIAS Y HUMANIDADES NAUCALPAN*. México: UNAM.

MESOGRAFÍA

AL eLearning elearningamericalatina 2003

http://www.elearningamericalatina.com/encuestas/latinoamerica_4.php.

Consultado Noviembre 2011

Chamilo Web Site Creative Commons, 2011.

<http://www.chamilo.org/>.

Consultado Noviembre 2011M

CVSS Current CVSS Score Distribution For All Vulnerabilities 2010.

<http://www.cvedetails.com/vulnerability-list>.

Consultado Septiembre de 2011.

Dokeos Dokeos, 2011

<http://www.dokeos.com/es/productos>.

Consultado Septiembre de 2011.

Leeuwe Marcel Historia del E-Learning

http://www.leerbeleving.nl/wbts/1/history_of_elearning.html.

Consultado Agosto 2011

Moodle

<http://moodle.org/>.

Consultado Agosto 2011

Mundial Foro Económico 2009

<http://www.networkedreadiness.com/gitr/main/fullreport/index.html>.

Consultado Septiembre 2011

Serrano Alfredo Calderon Aprendizaje a distancia 2011.

<http://aprendizajedistancia.blogspot.com/2009/06/el-uso-de-la-tecnologia-es-una.html>.

Consultado Marzo 2011

Tracker Security 2011

<http://securitytracker.com/archives/target/6829.html>.

Consultado Diciembre 2011

VirtualEduca Organización de los Estados Americanos, 2009.

<http://www.virtualeduca.info/ponencias2009/154/SGS.doc>.

Consultado Octubre 2011

Aviles Katrina Con clases en línea, ampliará la UNAM su oferta de educación a distancia

<http://www.elearningworkshops.com/modules.php?name=News&file=article&sid=207> / ed.

Jornada La. - La Jornada, 2006.

Consultado Octubre 2011

Ixel González y Bernardo Mendoza | El Universal

<http://www.eluniversal.com.mx/notas/739219.html>,

Consultado 22 Enero 2011