



Secretaría de Educación Pública
Subsecretaría de Educación

Sistema Nacional de Educación Superior Tecnológica

Educación General de Educación Superior Tecnológica



Instituto Tecnológico de
Tuxtla Gutiérrez

“APROXIMACIÓN CIENTÍFICA A LA INTEGRACIÓN DE LA
MODELACIÓN REUTILIZABLE Y COMPUTACIONAL
DE LAS CIENCIAS EDUCATIVAS”

Licenciatura En Informática

Residencia Profesional

Realizado en:

Universidad Tecnológica de la Mixteca

Presenta:

Vázquez Gómez Blanca Hilda

06270619

TUXTLA GUTIÉRREZ, CHIAPAS

DICIEMBRE DE 2010.



Sistema Nacional de Educación Superior Tecnológica



Dirección General de Educación Superior Tecnológica



Secretaría de Educación Pública
Subsecretaría de Educación

Instituto Tecnológico de Tuxtla Gutiérrez

“APROXIMACIÓN CIENTÍFICA A LA INTEGRACIÓN DE LA
MODELACIÓN REUTILIZABLE Y COMPUTACIONAL
DE LAS CIENCIAS EDUCATIVAS”

Licenciatura En Informática
Residencia Profesional

Realizado en:
Universidad Tecnológica de la Mixteca

Presenta:
Vázquez Gómez Blanca Hilda
06270619

Ing. Ma. Candelaria Gutiérrez Gómez
Asesor Interno

Dr. Iván Antonio García Pacheco
Asesor Externo

Ing. Roberto Cruz Gordillo
Revisor

TUXTLA GUTIÉRREZ, CHIAPAS

DICIEMBRE DE 2010.

ÍNDICE

Introducción	
Justificación	
CAPÍTULO I.- Objetivo General y objetivos específicos	5
1.1 Objetivo General	5
1.2 Objetivos específicos	5
CAPÍTULO II.- CARACTERIZACIÓN DEL ÁREA EN QUE PARTICIPÓ	6
2.1 Universidad Tecnológica de la Mixteca	6
2.2 División de Estudios de Posgrado	7
2.3 Organigrama: Universidad Tecnológica de la Mixteca	9
2.4 Ubicación	10
2.5 Infraestructura del Equipo de Cómputo:	11
CAPÍTULO III.- PROBLEMÁTICA	12
3.1 Problemas a resolver	12
CAPÍTULO IV.- ALCANCES Y LIMITANTES	14
4.1 Metas científicas	14
4.2 Delimitaciones del proyecto	14
4.3 Limitaciones del proyecto	14
CAPÍTULO V.- FUNDAMENTO TEÓRICO	15
5.1 Marco teórico conceptual	15
5.2 Marco teórico específico.	17
CAPÍTULO VI.- PROCEDIMIENTO Y DESCRIPCIÓN DE ACTIVIDADES REALIZADAS.	25
6.1 Etapas de Desarrollo	25
CAPÍTULO VII.- RESULTADOS	28
7.1 Diseño de la base de datos	28
CAPÍTULO VIII.- CONCLUSIONES Y RECOMENDACIONES	29
8.1 Conclusiones	29
8.2 Recomendaciones	30
BIBLIOGRAFÍA	31

INTRODUCCIÓN

El desarrollo de esta residencia profesional se logró dentro del marco del XIX Verano de Investigación Científica organizado por la Academia Mexicana de Ciencias y desarrollado en la Universidad Tecnológica de la Mixteca en el Estado de Oaxaca, con el proyecto CONACyT “APROXIMACION CIENTIFICA A LA INTEGRACION DE LA MODELACION REUTILIZABLE Y COMPUTACIONAL DE LAS CIENCIAS EDUCATIVAS” inscrito en la Convocatoria CB-2008-01 con referencia 106254, bajo la dirección del Dr. Iván Antonio García Pacheco Profesor- Investigador de tiempo completo perteneciente a la División de Estudios de Posgrado de la Universidad antes mencionada (SNI Exp: 43831).

Este trabajo de investigación se compone de ocho capítulos, los primeros tratan sobre los objetivos generales y específicos de esta residencia, el siguiente corresponde a la descripción del área en el cual se desarrolló esta propuesta científica, considera el organigrama de la División de Estudios así como la misión y visión de ésta; el capítulo siguiente hace mención de la problemática a resolver.

El cuarto capítulo, se refiere a los alcances y limitaciones de este proyecto, un capítulo más contiene todo el fundamento teórico que sustenta este trabajo de investigación, finalmente los últimos capítulos corresponden a la descripción de actividades así como el diseño de la base de datos (que se muestra gráficamente), y las conclusiones y recomendaciones.

JUSTIFICACIÓN

En la actualidad existen diferentes modelos educativos que tienen como finalidad brindar los servicios relacionados con el proceso de enseñanza-aprendizaje a cientos de estudiantes. Pero muchos de estos se crearon sin utilizar alguna metodología de desarrollo, y carecen de bases científicas y consideraciones sobre el proceso enseñanza-aprendizaje. Aún así, los modelos creados con bases poco confiables han tenido éxito. Por lo que se vuelven necesarios formalizar la experiencia obtenida con estos modelos mediante métodos formales, para conseguir desarrollar un modelo capaz de reutilizarse con éxito en distintos contextos y niveles educativos. Existe la necesidad de crear un modelo que confronte diferentes factores y permita obtener parámetros determinantes de la calidad de los estudiantes perciben en un modelo tradicional. El objetivo es comparar tentativamente diferentes propuestas.

Para la realización de este proyecto, se han identificado algunos puntos importantes que deben tomarse en cuenta en la creación de modelos educativos basados en modelos computacionales, entre los cuales pueden mencionarse:

- ✓ Generalmente las universidades con instalaciones físicas y modalidades virtuales, así como las universidades con modalidad virtual únicamente (sin instalaciones), no poseen toda la infraestructura que se requiere para soportar servicios Web adecuados para el funcionamiento de sus servicios en línea. Aun en el caso de contar con la infraestructura, hay instituciones que carecen de un sistema lo suficientemente completo, robusto y adecuado para cubrir las funciones esperadas,
- ✓ La estructura y el diseño básico del modelo educativo deben plantearse basados en las características que se observen que cada universidad desea dar a conocer entre sus usuarios.
- ✓ Se hará contar con un modelo pedagógico robusto que soporte el proceso de enseñanza-aprendizaje a distancia, así como la evaluación de expertos que aprueben los elementos y funcionalidades que se hayan considerado.

El enfoque que se dará al proyecto será multidisciplinario, aunque se enfocará más en los aspectos pedagógicos y computacionales. Algunos aspectos serán tomados en cuenta, tales como: elementos psicológicos, proceso cognitivo de enseñanza-aprendizaje a distancia y presencial, aspectos científicos, técnicos y de diseño.

CAPÍTULO I.- OBJETIVOS GENERAL Y ESPECÍFICO

1.1 Objetivo General

El principal objetivo es facilitar y asegurar la integración de los modelos reutilizables y computacionales en la consecución de las necesidades educativas independientemente de la audiencia objetivo. Mediante esta primera aproximación se intenta determinar la factibilidad de aplicar modelos computacionales en el establecimiento de mecanismos que proporcionen principios comunicativos y educativos sólidos que puedan ser reutilizados.

1.2 Objetivos específicos

- ✓ Identificar los aspectos más importantes que determinan el éxito y la calidad de enseñanza de los modelos educativos tradicionales y establecer aspectos de mejora mediante el análisis estructural de los mismos.
- ✓ Desarrollar una notación particular para registrar procesos educativos reutilizables.
- ✓ Establecer las guías para la definición y adaptación de procesos educativos reutilizables.
- ✓ Adaptar un modelo computacional para definir procesos educativos reutilizables.
- ✓ Adaptar el modelo computacional para adaptar procesos educativos reutilizables.
- ✓ Integrar los modelos reutilizable y computacional en un caso real en la División de Estudios de Posgrado de la Universidad Tecnológica de la Mixteca. Una etapa posterior sería generalizar el modelo educativo reutilizable y generar réplicas en el resto de Universidades incorporadas al SUNEО (Sistema de Universidades del Estado de Oaxaca).

CAPÍTULO II.- CARACTERIZACIÓN DEL ÁREA EN QUE PARTICIPÓ

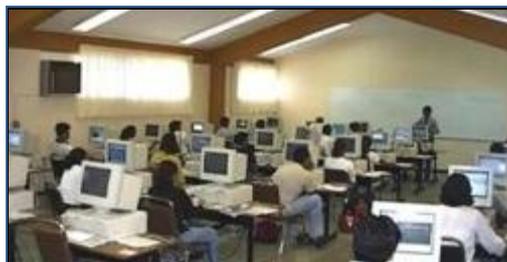
2.1 Universidad Tecnológica de la Mixteca.



La Universidad Tecnológica de la Mixteca es un organismo público descentralizado del gobierno del estado de Oaxaca, fundada por decreto del ejecutivo estatal, el cual fue publicado en el diario oficial, el día 18 de junio de 1990. Su creación responde a la múltiple finalidad de ofrecer oportunidades de formación científica y tecnológica

relevantes y de calidad a quienes deseen proseguir estudios superiores, desalentar la emigración de los jóvenes oaxaqueños, coadyuvar en el desarrollo de proyectos que activen la economía y creen empleo, así como abrir espacios de cultura que hagan extensivos los beneficios del conocimiento a la población en general.

La UTM se crea con un nuevo modelo de universidad cuyos criterios básicos de funcionamiento están referidos a la calidad académica para el desarrollo de sus funciones sustantivas de docencia, investigación, difusión de la cultura y promoción al desarrollo. Dicho modelo comprende una dedicación de tiempo completo a los estudios de licenciatura y de posgrado, por parte de estudiantes y profesores, los cuales interactúan en diversas actividades académicas como las clases teóricas, las prácticas de campo y laboratorio, las actividades programadas en salas de cómputo, biblioteca y laboratorios de idiomas.



Es una política permanente de la universidad la de garantizar las mejores condiciones de infraestructura para el adecuado desempeño académico de estudiantes y profesores, de tal forma que actualmente la UTM cuenta con instalaciones amplias, modernas y completamente equipadas conforme a las áreas de formación profesional y de investigación que realiza.

En los planes y programas de estudio de licenciatura y de postgrado se establece un equilibrio entre la formación teórica y la práctica, se tienen disponibles talleres especializados como los de plásticos, madera y metal, mecánicas, procesamiento de alimentos, electrónica, cerámica y serigrafía, así como laboratorios de física, química y telecomunicaciones, además de salas de idiomas de auto-acceso.



Los estudiantes cuentan con el apoyo del programa de becas de colegiatura, el cual, considerando la situación económica de los mismos y su desempeño académico, exenta total o parcialmente del pago correspondiente a este concepto. Este programa es de amplia cobertura, ya que más del 90% de los alumnos inscritos en la universidad tiene apoyo de este programa, por lo que ningún joven con interés por el estudio, deja de tener acceso a nuestra casa de estudios por razones económicas.

2.2 División de Estudios de Posgrado

La División de Estudios de Posgrado, tiene como función primordial promover los estudios de maestría y posgrado en sus diferentes disciplinas, algunas de las que destacan son:

- Maestría en Administración de Negocios
- Maestría en Ingeniería del Software
- Maestría en Medios Interactivos.

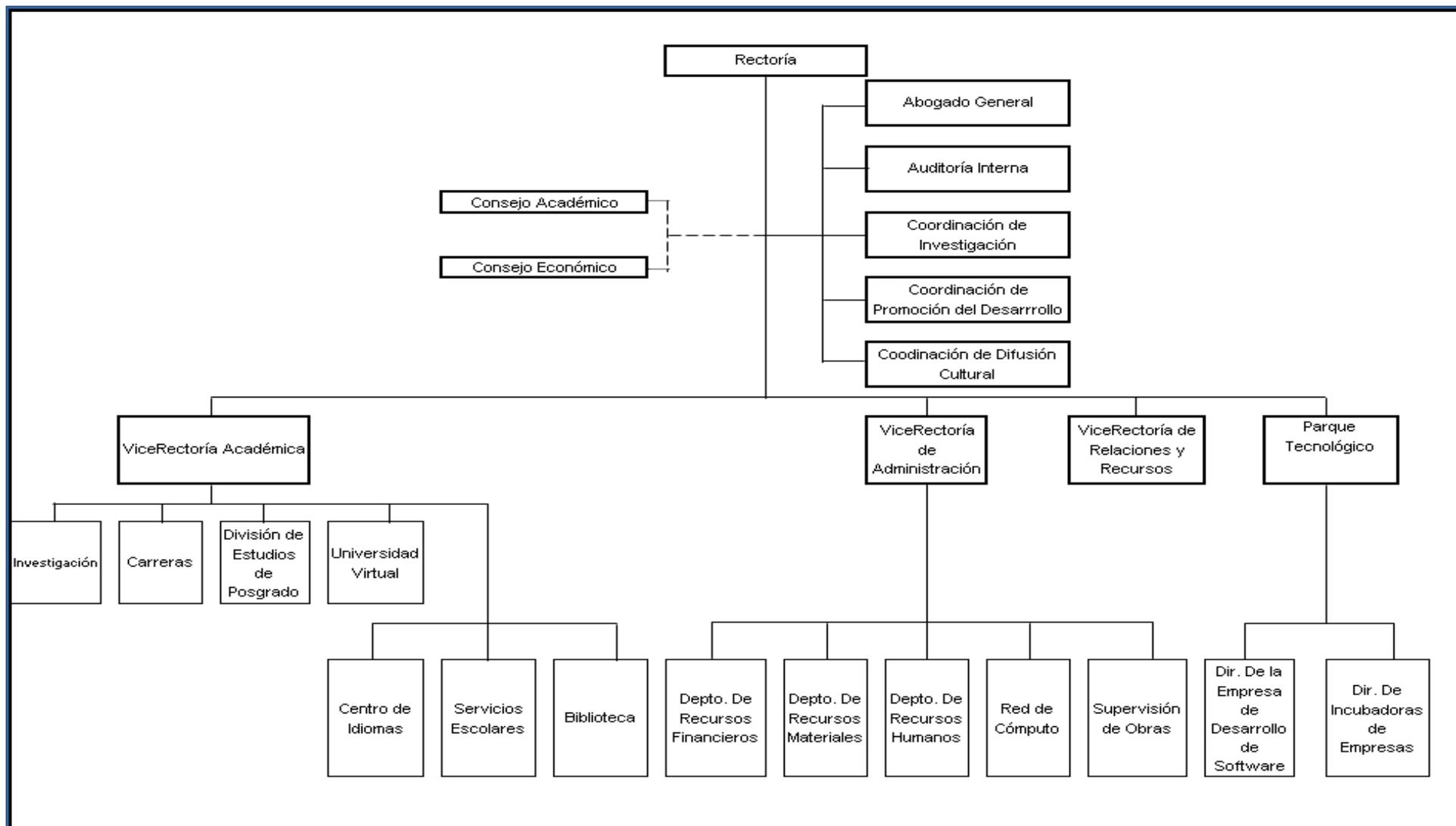


- Posgrado en Electrónica y Computación
- Maestría en Computación.
- Maestría en Sistemas Distribuidos.

Misión: La División de Estudios de Postgrado tiene la misión de ser un centro de excelencia académica en la enseñanza, el avance y la aplicación del conocimiento.

Visión: Ser un líder nacional en educación de postgrado de reconocido prestigio y calidad, en un ambiente de investigación de alto impacto a nivel nacional e internacional.

2.3 Organigrama: Universidad Tecnológica De La Mixteca



2.5 Infraestructura del Equipo de Cómputo:

Para el desarrollo del proyecto se contó con el siguiente equipo:

- División de Estudios de Posgrado (Cubículo de trabajo)
 - Estación de trabajo DELL Precisión T7400
 - Impresora Láser DELL 1320c.
 - Windows XP Profesional

- Laboratorio de Investigación e Ingeniería de Software (LIDIS)
 - Computadoras personales DELL OptiPlex 5200
 - Windows XP Profesional
 - Builder 6.0
 - Enterprise Architect

CAPÍTULO III.- PROBLEMÁTICA

3.1 Problemas a resolver

De acuerdo al proyecto establecido por el Gobierno del Estado de Oaxaca en el año de 1990, para crear un sistema de Universidades Regionales como modelo de excelencia académica, los 7 planteles con 12 campus de estas instituciones educativas que existen en la entidad, representan para los oaxaqueños una alternativa eficaz para la realización de sus estudios profesionales, con niveles de estudio reconocidos en el ámbito nacional e internacional.

A través de estas instituciones educativas se busca impulsar la educación profesional de la sociedad oaxaqueña para que la entidad avance en los indicadores socioeconómicos del país. Además, se desea descentralizar la educación para evitar la salida de los jóvenes de sus lugares de origen, limitar la concentración de los recursos culturales y económicos en las universidades grandes y crear instrumentos culturales de transformación del entorno social.

Cabe mencionar que las Políticas del Programa Nacional de Educación (PRONAE) promovidas en el marco de sus objetivos estratégicos tiene como uno de sus propósitos el impulsar la descentralización educativa para ampliar y consolidar sistemas de educación superior de buena calidad en cada estado, en los que se privilegie el equilibrio de la oferta, la comparación de los programas educativos, la movilidad estudiantil , el reconocimiento mutuo de créditos, la colaboración e intercambio académicos entre instituciones o, en su caso, de los nuevos programas que se abran en las existencias.

Para solventar estos propósitos, las Universidades Regionales han desarrollado un conjunto de procesos educativos con el afán de solventar las carencias tecnológicas. Estos procesos intentan que con las ventajas tecnológicas que ofrecen tanto el software como el hardware, se aprovechen mejor los recursos en la creación de sistemas interactivos de enseñanza que puedan cumplir con el objetivo de formar a los estudiantes que requieran entrenamiento especializado en áreas determinadas. Sin embargo, en la mayoría de ocasiones la tecnología no se ajusta a las necesidades reales y termina siendo un bien “sobrado”.

Actualmente, muchas Universidades destinadas a formar a los estudiantes no cuentan con modelos educativos formales (equipo, profesores, instalaciones, etc.) necesarios para brindar una formación adecuada, es por eso que el desarrollo de un modelo computacional podría representar una solución alternativa para dichos problemas. Sin embargo, dado el limitado presupuesto asignado a las Universidades Públicas, y en particular al estado de Oaxaca, es imposible competir con modelos educativos como los establecidos en la UNAM y el Instituto Tecnológico de Estudios Superiores de Monterrey (ITESM). Tenemos que encontrar la forma de establecer métodos de enseñanza-aprendizaje que puedan reutilizarse en diversos entornos con la formalidad de un modelo computacional.

Independientemente de la Institución educativa que se trate, todos utilizan un recurso que les permite mejorar el modelo educativo tradicional: las computadoras. Esta tecnología puede representar un medio interactivo en el que la comunicación se puede generar tanto desde el emisor como desde el receptor. Es descentralizado y permite transmitir o consultar grandes volúmenes de información, hacer públicos contenidos informativos y acceder a ellos con costos comparativamente mucho más bajos que en otros medios de comunicación de masas y principalmente, está disponible para quien desee usarlas.

De esta forma, un modelo computacional podría constituir un medio propicio para apoyar o desarrollar procesos de aprendizaje mediante la “creación” de nuevas modalidades e acceso al conocimiento que tengan en cuenta las especificaciones de la tecnología. No obstante, muchas de las propuestas actuales educativas presentan claramente un modelo generativo pedagógico que las sustenta y responde a las características propias del medio de manera rutinaria. Según Santágelo “es poco frecuente encontrar que dichas experiencias comiencen por caracterizar la situación de aprendizaje el modelo educativo de referencia y el diseño de estrategias para ese entorno.”

CAPÍTULO IV.- ALCANCES Y LIMITACIONES

4.1 Metas científicas:

- ✓ La creación del modelo educativo reutilizable y las medidas de éxito del estudio. Los avances y resultados alcanzados quedarán a disposición de la comunidad científica mediante un acceso controlado desde el portal de la Universidad Tecnológica de la Mixteca.

4.2 Delimitaciones del proyecto

El desarrollo de este proyecto se enfoca a proporcionar una herramienta gráfica de diseño de SE basada en el enfoque de reutilización de patrones (vistos estos últimos como componentes de un proceso más grande). Las delimitaciones del proyecto son:

- La propuesta se basa en una iniciativa enfocada a proveer elementos gráficos que permitan diseñar componentes reutilizables para cualquier SE.
- La facilidad de validar el producto generado en la UTM nos obliga a limitar a éste entorno académico los resultados o hallazgos alcanzados.
- La herramienta producto de este proyecto, se centra únicamente en las fases de análisis y diseño.

4.3 Limitaciones del proyecto

- La notación gráfica propuesta será la base para desarrollar un conjunto de procesos reutilizables para el desarrollo de SE.
- La notación cubrirá hasta la parte de diseño de un ciclo estándar de desarrollo, la implementación dependerá del programador/desarrollador de un SE.
- El resultado del proyecto será probado en el entorno académico de la UTM por lo que se verán reflejados los contenidos curriculares de aquellos profesores interesados en utilizar una aplicación informática para mejorar el aprovechamiento de su curso.

CAPÍTULO V.- FUNDAMENTO TEÓRICO

5.1 Marco teórico conceptual

Inicialmente se consideran para el modelo de estudio de las demandas de servicios, restricciones y rasgos relevantes de las Universidades Regionales que tengan modelos educativos tradicionales, definiendo aspectos de comparación de al menos 20 sistemas análogos. Todos los datos obtenidos servirán para establecer un análisis estadístico y poder calcular información útil de ellos. Se utilizarán métodos matemáticos para la comparación de los objetos agrupados por sus propiedades, Los atributos más importantes serán seleccionados de acuerdo a un modelo computacional y que permitirá que pueda utilizarse en distintos niveles de enseñanza a nivel superior.

La metodología científica (véase figura 1) consistiría las siguientes fases:

- ✓ *Definir un proceso educativo utilizable.* Incluso si no existe un modelo educativo anterior se tendría que empezar desde este punto. El resultado serán una o más descripciones de un proceso reutilizable, junto con las guías de adaptación. Las descripciones del proceso también se integrarán en una librería de procesos educativos si se ha probado que son aptos para reutilizarse.
- ✓ *Seleccionar un proceso educativo para adoptarlo en un nuevo entorno.* Una vez que el proceso reutilizable está completo, el cargado de su desarrollo está listo para desplegarlo en un entorno real.
- ✓ *Adaptar el proceso reutilizable al entorno.* El proceso educativo reutilizable se adapta para resolver las necesidades educativas específicas y el entorno técnico y educacional de un proyecto educativo.
- ✓ *Decretar como exitoso el proceso en el entorno. (validación de la hipótesis).* El proceso se pone en práctica en el proyecto educativo. Una función de evaluación examina el proyecto para asegurarse de que el nuevo proceso está decretado fielmente.
- ✓ *Refinar el proceso.* De acuerdo con la evaluación anterior la definición del proceso reutilizable se puede refinar y se inserta en al librería de procesos educativos como “buen proceso”.

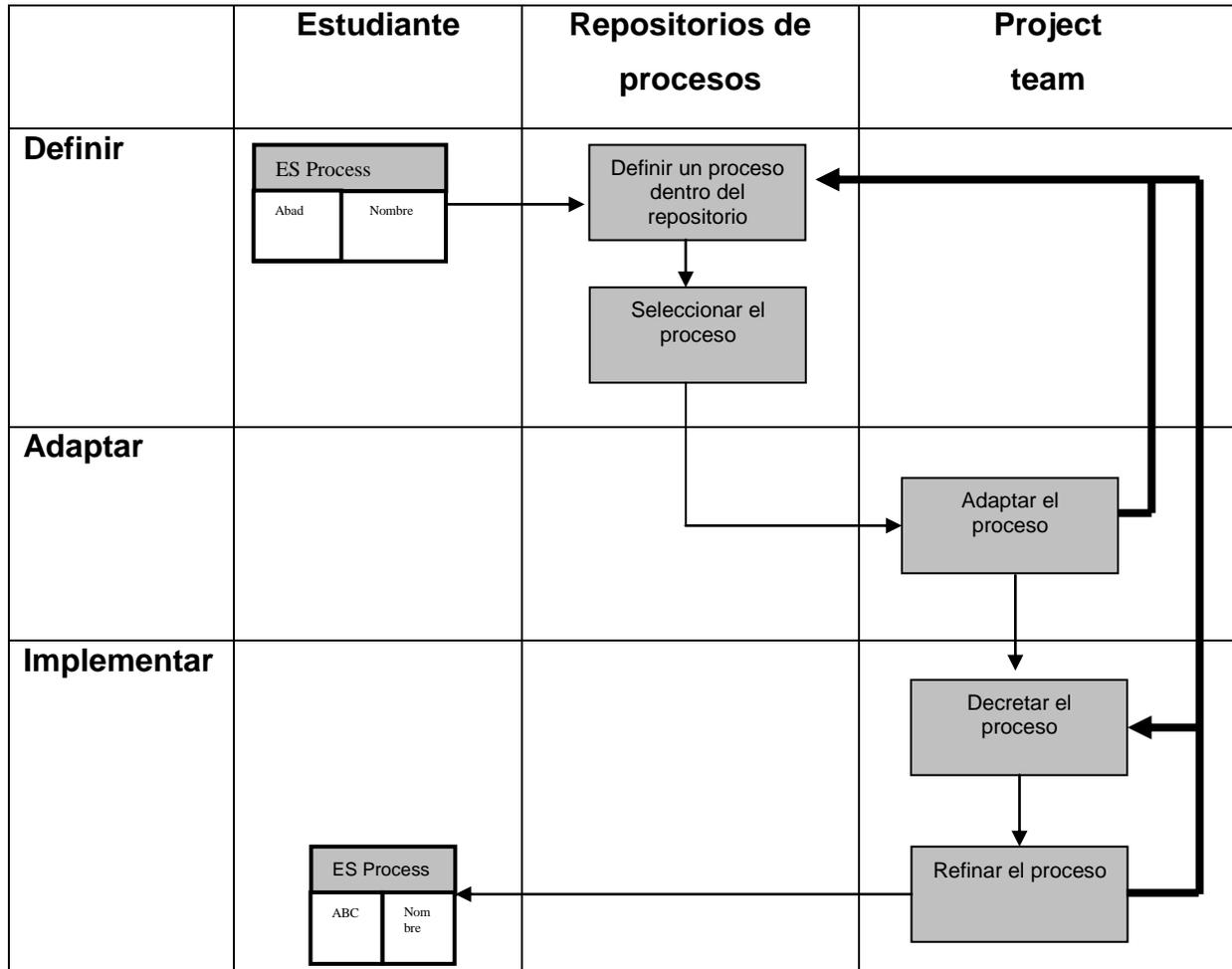


Figura 1. Reutilización de procesos educativos bajo el enfoque de un modelo computacional.

Para este proyecto será validado y refinado el modelo IADIAL desarrollado en la División de Estudios de Postgrado de la Universidad Tecnológica de la Mixteca. IADIAL (Initiating Analyze, Design, Implementating, Assessment, and Learning) refleja los aspectos pedagógicos requeridos para cualquier entorno educativo. Este modelo pedagógico está basado en la Técnica de Resolución de Problemas con modificaciones orientadas a mejorar los resultados del aprendizaje a distancia (véase figura 2).

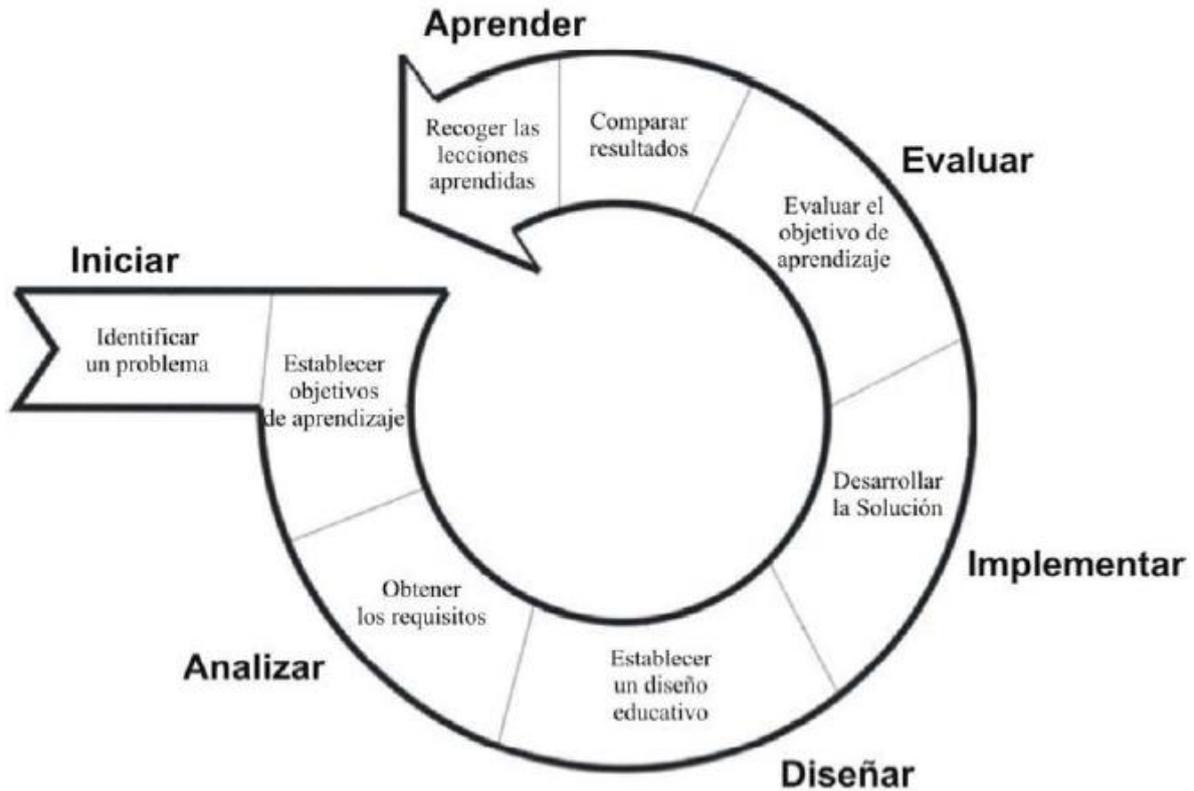


Figura 2.- Modelo Pedagógico IDIAL

5.2 Marco teórico específico.

Una base de datos es una fuente central de datos destinadas a compartirse entre muchos usuarios para una diversidad de aplicaciones. El corazón de una base de datos lo constituye el sistema de administración de bases de datos (DBMS), el cual permite la creación, modificación y actualización de la base de datos, la recuperación de datos y la generación de informes y pantallas. El enfoque de base de datos tiene la ventaja de permitir a los usuarios obtener su propia vista de los datos. Los usuarios no tienen que preocuparse por la estructura real de la base de datos o de su almacenamiento físico. (Kendall, 2005)

Kendall afirma que los objetivos de efectividad de la base de datos son:

1. Asegurar que los datos se puedan compartir entre los usuarios para una diversidad de aplicaciones.
2. Mantener los datos que sean exactos y consistentes.
3. Asegurar que todos los datos requeridos por las aplicaciones actuales y futuras se podrán acceder con facilidad.

4. Permitir a la base de datos evolucionar conforme aumenten las necesidades de los usuarios.
5. Permitir a los usuarios construir su vista personal de los datos sin preocuparse por la forma en que los datos se encuentran almacenados físicamente.

También, Nevado Cabello afirma que, una base de datos es una colección interrelacionada de datos, almacenados en un conjunto sin redundancias innecesarias cuya finalidad es la de servir una o más aplicaciones de la manera más eficiente. Las características que debe tener una base de datos son:

- Independencia lógica de los datos: Permite cambiar la representación lógica del problema sin afectar a los programas de aplicación que los manipula.
- Independencia física de los datos: Permite que la distribución en las unidades de almacenamiento y la estructura física de la información almacenada y la estructura física de la información almacenada sea independientemente de los cambios de la estructura lógica, y por tanto de los procedimientos que manejan las mismas. (Nevado Cabello, 2008)

A finales de los años 60, Edgar Frank Codd definió las bases del Modelo Relacional, donde los datos se estructuran lógicamente en forma de relaciones (tablas). Codd, que era un experto matemático, utilizó una terminología perteneciente a las matemáticas, en concreto de la teoría de conjuntos y de la lógica de predicados. El objetivo fundamental del modelo relacional es mantener la independencia de esta estructura lógica respecto al modo de almacenamiento y a otras características de tipo físico.

La estructura básica del modelo relacional es la relación o tabla. Todos los datos de la base de datos se representan en forma de tabla o relación cuyo contenido varía con el tiempo. Una relación o tabla se representan gráficamente como una estructura rectangular, formada por las filas o columnas. Cada columna almacena información sobre una propiedad determinada de la tabla o relación. A estas columnas también se las denomina atributos. Los atributos pueden aparecer en la relación en cualquier orden. (Nevado Cabello, 2008)

Cada fila de la tabla se denomina tupla, y representa una ocurrencia de la relación. Cada tupla almacena los valores que toma cada uno de los atributos, para cada ocurrencia de la relación. Cada relación o tabla está formada por dos partes:

- Cabecera.- Conjunto fijo de pares atributo:dominio, donde cada atributo pertenece a un único dominio subyacente y todos los atributos son distintos, es decir, no hay atributos que se llamen igual.
- Cuerpo.- Es un conjunto variable de tuplas. Cada tupla es un conjunto de pares atributo:valor, cuyo valor varía con el tiempo. Un dominio D es un conjunto finito de valores homogéneos y atómicos V_1, V_2, \dots, V_n caracterizado por un nombre. Se dice que los valores son homogéneos porque son todos del mismo tipo, y atómicos porque son indivisibles.

Las relaciones se pueden clasificarse por tipos de anomalías de modificación a las cuales son vulnerables. En la década de 1970, los teóricos relacionales investigaron acerca de estos tipos. Cuando alguien encontraba una anomalía, la clasificaba y pensaba en una manera de prevenirla. Cada vez que esto sucedía se mejoraban los criterios de diseño de las relaciones. Estas clases de relaciones, así como las técnicas para prevenir anomalías, se denominan formas normales. Dependiendo de su estructura, una relación puede estar en la primera forma normal, en la segunda o en alguna otra. (Kroenke, 2003)

Primera forma normal: Una relación R se encuentra en primera forma normal (1FN) si y sólo si todos los dominios simples subyacentes contienen sólo valores atómicos. Es decir, que el cruce de una fila con una columna sólo tiene un dato, o sea, que no existen grupos repetitivos.

Segunda forma normal: Una relación R se encuentra en segunda forma normal si y sólo si está en primera forma normal y todos los atributos no clave, dependen funcionalmente de forma completa de la clave primaria.

Tercera forma normal (Boyce- Codd): Se dice que una relación está en tercera forma normal si y sólo si está en segunda forma normal y todos los atributos no clave dependen de manera no transitiva de la clave primaria.

Cuarta forma normal: Una relación R se encuentra en cuarta forma normal, si y sólo si está en forma normal de Boyce-Codd y no existen dependencias multivaluadas.

Quinta forma normal: Para definir la quinta forma normal, primero debemos de definir una dependencia de reunión. Se dice que una relación R satisface la dependencia de reunión (DR) (X,Y,...Z) si y sólo si R es igual a la reunión de sus proyecciones según X, Y,...Z donde X,Y,...Z son subconjuntos de atributos de la relación R. (Nevado Cabello, 2008).

Enterprise Architect combina el poder de la última especificación UML 2.1 con alto rendimiento, interfaz intuitiva, para traer modelado avanzado al escritorio, y para el equipo completo de desarrollo e implementación. Con un gran conjunto de características y un valor sin igual para el dinero, EA puede equipar a su equipo entero, incluyendo analistas, evaluadores, administradores de proyectos, personal del control de calidad, equipo de desarrollo y más, por una fracción del costo de algunos productos competitivos. (Argentina, 2007)

Enterprise Architect es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. EA es una herramienta multiusuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad.

El Lenguaje Unificado de Modelado provee beneficios significativos para ayudar a construir modelos de sistemas de software rigurosos y donde es posible mantener la trazabilidad de manera consistente. Enterprise Architect soporta este proceso en un ambiente fácil de usar, rápido y flexible. Para una mirada rápida al modelado UML en Enterprise Architect vea nuestro tutorial UML y documentos.

Enterprise Architect provee trazabilidad completa desde el análisis de requerimientos hasta los artefactos de análisis y diseño, a través de la implementación y el despliegue. Combinados con la ubicación de recursos y tareas incorporados, los equipos de Administradores de Proyectos y Calidad están equipados con la información que ellos necesitan para ayudarles a entregar proyectos en tiempo.

Las bases de Enterprise Architect están construidas sobre la especificación de UML 2.0. Usa Perfiles UML para extender el dominio de modelado, mientras que la Validación del Modelo asegura integridad. Combina Procesos de Negocio, Información y Flujos de trabajo en un modelo usando nuestras extensiones gratuitas para BPMN y el perfil Erickson-Penker.

EA le ayuda a administrar la complejidad con herramientas para rastrear las dependencias, soporte para modelos muy grandes, control de versiones con proveedores CVS o SCC, Líneas Base por cada punto del tiempo, la utilidad de comparar (DIFF) para seguir los cambios del modelo, interfaz intuitiva y de alto rendimiento con vista de proyecto como un "explorador". (Argentina, 2007)

EA soporta generación e ingeniería inversa de código fuente para muchos lenguajes populares, incluyendo C++, C#, Java, Delphi, VB.Net, Visual Basic y PHP. También hay para CORBA y Python disponibles. Con un editor de código fuente con "resaltador de sintaxis" incorporado, EA le permite navegar y explorar su modelo de código fuente en el mismo ambiente. Para aquellos que trabajan en Eclipse o Visual Studio.Net, Sparks Systems también vende puentes livianos para estas Ideas, permitiéndole modelar en EA y saltar directamente al código fuente en su editor preferido. Las plantillas de generación de código le permiten personalizar el código fuente generado a las especificaciones de su compañía.

EA le ayuda a visualizar sus aplicaciones soportando ingeniería inversa de un amplio rango de lenguajes de desarrollo de software y esquemas de repositorios de base de datos. Ingrese Framework completos desde código fuente o archivos Java .jar - o aún ensambladores binarios .Net! Importando Framework y librerías de código, Ud. puede maximizar la re-utilización y entendimiento de su inversión existente.

EA soporta transformaciones de Arquitectura avanzada dirigida por Modelos (MDA) usando plantillas de transformaciones de desarrollo y fáciles de usar. Con transformaciones incorporadas para DDL, C#, Java, EJB y XSD, Ud. puede rápidamente desarrollar soluciones complejas desde los simples "modelos independientes de plataforma" (MIP) que son el objetivo en "modelos específicos de plataforma" (MEP). Un MIP se puede usar para generar y sincronizar múltiples MIP's proveyendo un aumento de productividad significativo.

C++ Builder es una aplicación Windows que proporciona un entorno de trabajo visual para construir aplicaciones Windows que integra distintos aspectos de la programación en un entorno unificado o integrado. La integración y facilidad de manejo hace que sea una herramienta indispensable para el desarrollo rápido de aplicaciones o RAD (Rapid Application Development). Guarda una gran similitud con el IDE de *Visual Basic*, aunque existen ciertas diferencias.

IDE es el acrónimo de *Integrated Development Environment* o entorno de desarrollo integrado. El IDE de C++ Builder es una aplicación Windows 95 y como tal cumple con los estándares de aspecto, diseño y comportamiento que aconseja Microsoft a los desarrolladores de aplicaciones. En consecuencia, cualquiera que esté familiarizado con el manejo a nivel de usuario de Windows 95 no le supondrá ningún esfuerzo manejarlo con soltura. (Cortijo Bon & Berzal Galia)

El entorno de desarrollo se divide, básicamente, en tres partes. Una serie de ventanas, que pueden estar visibles u ocultas, constituyen la base de C++ Builder. En la parte superior se coloca la ventana principal, que contiene el menú principal, la barra de herramientas (a la izquierda) y la paleta de componentes (a la derecha). Debajo de la ventana principal, y a la izquierda se coloca el inspector de objetos. A la derecha del inspector de objetos está el área de trabajo de C++ Builder, que inicialmente muestra el diseñador de formularios, y escondido u oculto parcialmente tras éste aparece el editor de código. Al abrir una aplicación de Builder consta de los siguientes elementos:

1. **Menú principal:** Permite el acceso a todas las operaciones y posibilita la configuración del programa. En el título del menú principal aparece el nombre de la aplicación (*C++ Builder*) y el nombre del proyecto/grupo de proyectos con el que actualmente se está trabajando. Por defecto, asigna el nombre Project1 al proyecto con el que se va a trabajar, aunque éste deberá cambiarse, lógicamente.
2. **Barra de herramientas:** Permite un acceso rápido a las operaciones que se realizan más frecuentemente. La barra de herramientas tiene como objeto acelerar las operaciones más comunes del menú principal.
3. **Paleta de componentes:** Agrupa a los componentes que pueden incluirse en las aplicaciones. Los componentes constituyen los bloques básicos sobre

la que se construyen aplicaciones Windows con C++ Builder basadas en la VCL (Visual Component Library) y son la base de la VCL. Todos los componentes (visuales y no visuales) están accesible rápida y cómodamente gracias a la paleta de componentes. Podemos decir que es un gran almacén de componentes listos para ser incorporados a una aplicación, seleccionándolo y colocándolo en un formulario. Por conveniencia, todos los componentes disponibles se encuentran organizados en distintas páginas o carpetas según su funcionalidad. Cada pestaña de la paleta de componentes da acceso a un conjunto de iconos que representan a componentes que pueden usarse para diseñar la aplicación.

4. **Diseñador de formularios:** Es una ventana cuadrículada sobre el que se disponen los componentes para diseñar las ventanas que formarán la aplicación. Es una herramienta visual destinada a diseñar y presentar la interfaz de usuario (la apariencia externa) de una aplicación. Un formulario puede ser la ventana principal de un programa, un cuadro de diálogo o cualquier otra ventana.
5. **Inspector de objetos:** Para cambiar las *propiedades* de los objetos que forman la aplicación y seleccionar los *eventos* a los que debe responder la aplicación. Cada componente tiene asociado un conjunto de propiedades y métodos y un conjunto de eventos a los que puede responder. Con el inspector de objetos podremos moldear los componentes de una aplicación según nuestras necesidades, en cuanto a su apariencia (*propiedades*) y funcionalidad (*Eventos a los que puede responder*). En definitiva, podemos modificar las *propiedades* de los componentes y construir los *gestores de eventos* a los que éstos pueden responder.
6. **Editor de código:** Un típico editor de texto multiventana para ver y editar el código de la aplicación. Está perfectamente integrado con el inspector de objetos y el diseñador de formularios. **Inspector de clases:** Es un navegador que muestra las *clases*, *objetos* y *métodos* asociados a la aplicación. Aparece por defecto asociada al editor. Para abrir esta ventana: View | ClassExplorer.
7. **Administrador de proyectos:** Es básicamente un navegador entre los diferentes ficheros que forman la aplicación. No aparece por defecto, y

cuando se abre (View | Project Manager) se muestra como una ventana independiente. (Cortijo Bon & Berzal Galia).

El modelo educativo se describe mediante un sistema que integra fundamentalmente aportes de tres componentes: el Pensamiento de Sistemas (P.S.), el Enfoque Pedagógico Constructivista (E.P.C.) y la Dinámica de Sistemas (D.S.) como metodología guía para la explicación y representación de la estructura y dinámica de los objetos de estudio. La integración de estos tres componentes se da en el escenario de la Práctica Educativa Sistémica (P.E.S.), la cual a su vez se desenvuelve matizada por el contexto cultural en el cual se desarrolla, contexto que podría constituirse en el cuarto elemento del modelo.

En el nivel filosófico se integran en la praxis los componentes ontológico, epistemológico y metodológico. Para el nivel Educativo General la práctica educativa representa el Paradigma de Pensamiento (P.P), el Enfoque Pedagógico (E.P.) y Métodos y Tecnología (M y T). En el nivel Educativo Sistémico la Práctica Educativa Sistémica (P.E.S.) integra el Enfoque Pedagógico Constructivista (E.P.C) al Pensamiento de Sistemas (P.S) y a la Dinámica de Sistemas (D.S.). Finalmente, al nivel operativo el Ambiente Educativo Centrado en procesos de Pensamiento (A.E.C.P), se construye con aportes del constructivismo (Aportes E.P.C.), formas del Pensamiento Dinámico-Sistémico (F.P.) y los útiles informáticos y metodológicos derivados de la Dinámica de Sistemas (Aportes DS). (Andrade Sosa & Parra O, 1998)

Tradicionalmente el aprendizaje escolar fue considerado bajo un enfoque didáctico: maestro/alumno. De esta forma, el proceso educativo fue ingenuamente reducido al encuentro entre dos sujetos, ignorando el resto de las variables intervinientes.

Este enfoque se modifica con la concepción tríadica: docente/alumno/saber en un contexto constituido por el entorno escolar. El sistema de enseñanza se instala también dentro de un sistema social (de acuerdo a Chevallard, los padres, los científicos y la instancia política) en el cual entran en juego aspectos fundamentales del funcionamiento didáctico: las negociaciones, los conflictos, y las decisiones sobre el saber que habrá de enseñarse en la escuela (problema de la transposición didáctica). (Caldeiro, 2005)

CAPÍTULO VI.- PROCEDIMIENTO Y DESCRIPCIÓN DE ACTIVIDADES REALIZADAS.

6.1 Etapas de Desarrollo

A continuación se especifican las etapas de desarrollo del proyecto, las cuales serán divididas más adelante en la planeación del mismo.

Fase 1

Desarrollar una notación para registrar procesos educativos reutilizables.

- ✓ Definir un conjunto de elementos comunes y variantes para el proceso educativo.
- ✓ Desarrollar un marco de trabajo para la definición de procesos educativos reutilizables.
- ✓ Establecer una jerarquía de elementos en tres secciones: descripción del objeto de aprendizaje, descripción del entorno, y descripciones de las interfaces.
- ✓ Establecer una notación gráfica que permita asimilar fácilmente la composición de los procesos educativos.

Establecer las guías para la definición y adaptación de procesos educativos reutilizables.

- ✓ Crear procesos educativos reutilizables que puedan ser utilizados por cualquier entorno de aprendizaje.
- ✓ Crear mecanismos de transferencia de procesos educativos reutilizables.
- ✓ Crear mecanismos de refinación de procesos educativos reutilizables.

Fase 2

Adaptar el modelo computacional para definir procesos educativos reutilizables.

- ✓ Determinar las condiciones operativas bajo las cuales será posible reutilizar los procesos educativos definidos.
- ✓ Analizar los procesos bajo el enfoque del modelo computacional para generar definiciones genéricas reutilizables.
- ✓ Realizar definiciones preliminares de los procesos educativos reutilizables.

- ✓ Realizar la modelación reutilizable detallada para cada proceso educativo definido.

Adaptar el modelo computacional para adaptar procesos educativos reutilizables.

- ✓ Desarrollar las descripciones gráficas que describan textualmente y gráficamente el proceso educativo específico.

Fase 3

Integración del modelo reutilizable y computacional en un caso de estudio real.

- ✓ Comprobar que la integración puede alcanzar los objetivos de aprendizaje establecidos al inicio del proceso.

Durante el tiempo en que se llevó a cabo el desarrollo de esta residencia, como primera actividad a realizar fue el diseño de la base de datos, se llevaron a cabo los diseño lógicos y físicos de la base de datos, así también se aplicó las formas de normalización para disminuir las anomalías y redundancias de los datos.

Es importante recalcar que cuanto se inicia esta residencia ya se tenía contemplada la plataforma de Gestión de procesos, así como la metodología que contenía todo el análisis de esta propuesta científica.

Para desarrollar el diseño de la base de datos, se utilizó la herramienta Enterprise Architect, con la cual se implementó todo el diseño de la misma, para la declaración de variables en esta herramienta se utilizó Postgre y bajo esta técnica se diseño toda la base de datos compuesta por más de 50 tablas, las cuáles integraba desde las tablas alumno, usuarios, proyectos, plantillas, faceta, actor, respuesta, bloque, actividad, profesor, entre otras.

Después de realizar dicho diseño, pasó a proceso de revisión, en la cual se probó que el diseño estaba correcto, después de este análisis y aprobación, se continuó con la fase de programación, en la cual se utilizó la herramienta Builder 6.0 con la cual se inició desde cero. Los íconos y las convenciones para los diagramas de secuencia, ya estaban estipulados por el proceso de MESOFT.

Los siguientes meses fueron dedicados, para la programación y creación de la base de datos; con la plataforma creada se logró la conexión de la base de datos y que se fueran actualizando los registros necesarios.

Al crear la plataforma, sobre la cual los alumnos y profesores trabajaran, fue una dura tarea, ya que se programó y consideró todos los aspectos, desde que se detectara un clic y se abriera las diferentes opciones que tanto el alumno como el catedrático tenía permitido. Así también como el arrastre de objetos, y obtener ventanillas diferentes para un simple clic izquierdo o derecho y que se guardaran los cambios que hacían en una sesión actual y si deseaba el usuario después regresar y continuar trabajando, los objetos que había creado deberían estar en las coordenadas que había colocado en la sesión anterior así como los parámetros estipulados por él, y todos estos cambios se almacenaban en tiempo real en la base de datos.

Para la elaboración de todo el diseño de la base de datos, se trabajaron dos semanas continuas hasta lograr obtener un diseño que no tuviera anomalías ni redundancia de datos.

Los siguientes meses se emplearon para la creación del modelo computacional. Al final se entregó el diseño completo de la base de datos, así cómo el sistema antes planteado.

CAPÍTULO VII.- Resultados

CAPÍTULO VIII.- CONCLUSIONES Y RECOMENDACIONES

8.1 Conclusiones

Esta residencia profesional se concluye de forma exitosa, ya que se cumplió con lo plasmado al principio, que era integrar el análisis así como la metodología de MESOFT y convertirlo en el diseño de bases de datos y lograr el modelo computacional requerido, la siguiente fase es realizar las pruebas con el sector involucrado, y rectificar y analizar que tan eficiente fue el desarrollo de este arduo trabajo.

Se logró la creación del modelo educativo reutilizable bajo el entorno de herramientas computacionales, el objetivo principal de esta residencia profesional se alcanzó ya que es factible técnicamente aplicar modelos computacionales en el establecimiento de mecanismos que proporcionen principios comunicativos y educacionales sólidos que puedan ser reutilizados.

El equipo de trabajo en esta aproximación científica fue excelente ya que cada quién aportó con su granito de arena para que se lograra el cometido de esta residencia profesional.

Como futura licenciada fue una excelente experiencia haber sido partícipe de este trabajo de investigación inscrito en la Convocatoria CB-2008-01 con referencia 106254, dentro del marco del XIX Verano de Investigación Científica organizado por la Academia Mexicana de Ciencias y desarrollado en la Universidad Tecnológica de la Mixteca en el Estado de Oaxaca.

8.2 Recomendaciones

La única recomendación, es que las Universidades sigan impulsando este tipo de actividades, de manera que los estudiantes de nivel licenciatura puedan participar en proyectos que otras Instituciones a través de su grupo de Investigadores estén realizando, y fomentar el trabajo de investigación y porque no “inyectar” esa actitud para realizar estudios de posgrado en un futuro, y tener más maestros y doctores que tanta falta le hace a nuestro gran país México.

BIBLIOGRAFÍA

Andrade Sosa, H. H., & Parra O, C. A. (23 de Octubre de 1998). *RIBIE 98, IV Congreso da Rede Iberoamericana de Informática Educativa*. Recuperado el 02 de Diciembre de 2010, de <http://www.c5.cl/ieinvestiga/actas/ribie98/183.html>

Argentina, S. S. (2007). *SPARX SYSTEMS*. Recuperado el 01 de Diciembre de 2010, de <http://www.sparxsystems.com.ar/products/ea.html>

Caldeiro, G. P. (2005). *¿Qué es el aprendizaje escolar?* Recuperado el 02 de Diciembre de 2010, de <http://educacion.idoneos.com/index.php/283093>

Cortijo Bon, F. J., & Berzal Galia, F. (s.f.). *Curso de C++ Builder*. Recuperado el 2 de Diciembre de 2010, de <http://elvex.ugr.es/decsai/builder/intro/2.html#AYUDA>

Kendall, J. E. (2005). *Análisis y diseño de sistemas*. Pearson Educación.

Kroenke, D. M. (2003). *Procesamiento de bases de datos: fundamentos, diseño e implementación*. México: Pearson Educacion.

Nevado Cabello, M. V. (2008). *Introducción a las bases de datos relacionales*. Madrid: Visión Libros.