GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL
PROCEDIMIENTO DESARROLLO CURRICULAR
GUÍA DE APRENDIZAJE

1. IDENTIFICACIÓN DE LA GUÍA DE APRENDIZAJE

- **Denominación del Programa de Formación:** Tecnólogo en Análisis y Desarrollo de Sistemas de Información.
- **Código del Programa de Formación:** 228106
- **Nombre del Proyecto**: Diseño y construcción de software a la medida para el sector empresarial.
- **Fase del Proyecto**: Análisis
- **Actividad de Proyecto:** AP4. Determinar la estructura lógica del sistema de información.
- **Competencias**:

220501033. Diseñar el sistema de acuerdo con los requisitos del cliente.

220501009. Participar en el proceso de negociación de tecnología informática para permitir la implementación del sistema de información.

240201502. Producir textos en inglés en forma escrita y oral.

240201500. Promover la interacción idónea consigo mismo, con los demás y con la naturaleza en los contextos laboral y social.

- **Resultados de Aprendizaje.**

22050103304. Construir el prototipo del sistema de información, a partir del análisis de las características funcionales del sistema en relación con facilidad de manejo, funcionalidad y experiencia del usuario, apoyado en software aplicado según protocolos de diseño.

22050103306. Diseñar la estructura de datos, a partir del modelo conceptual determinado en el análisis del sistema, utilizando herramientas tecnológicas de bases de datos, según las normas y estándares establecidos.

22050100902. Definir estrategias para la elaboración de términos de referencia y procesos de evaluación de proveedores, en la adquisición de tecnología, según protocolos.

24020150203. Comprender una amplia variedad de frases y vocabulario en inglés sobre temas de interés personal y temas

24020150207. Identificar formas gramaticales básicas en textos y documentos elementales escritos en inglés

24020150002. Desarrollar procesos comunicativos eficaces y asertivos dentro de criterios de racionalidad que posibiliten la convivencia, el establecimiento de acuerdos, la construcción colectiva del conocimiento y la resolución de problemas de carácter productivo y social.

24020150014. Asumir actitudes críticas, argumentativas y propositivas en función de la resolución de problemas de carácter productivo y social.

- **Duración de la Guía:** 302 horas / 1,7 meses.

2. PRESENTACIÓN

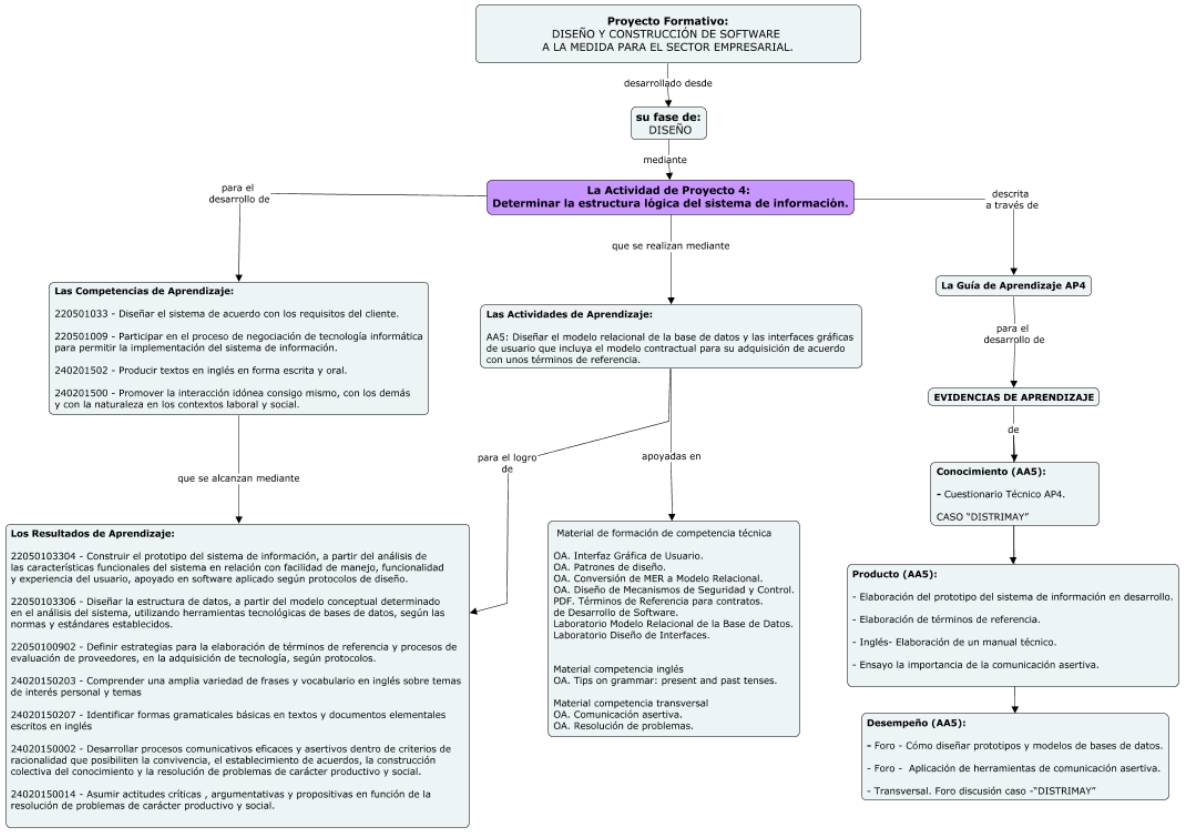
La presente guía de aprendizaje orienta el desarrollo de la actividad de proyecto 4. ***"Determinar la estructura lógica del sistema de información"***, en ella, el aprendiz abordará temáticas que le permitirán de acuerdo con su proyecto formativo, elaborar los aspectos que le permiten determinar cómo se va a construir el sistema de información propuesto en la fase de análisis por medio de los diseños de interfaces, modelos de bases de datos, utilizando patrones de diseño con base en la arquitectura de la solución a implementar.

Adicionalmente, la actividad de proyecto en su visión integradora, contempla el abordaje de contenidos referidos a la comprensión de textos en inglés, tanto en forma escrita como auditiva, y desarrollar procesos comunicativos asertivos y eficaces que mejoren la convivencia, el establecimiento de acuerdos, la construcción colectiva de conocimientos en función de la resolución de problemas de carácter productivo y social.

El desarrollo de la presente actividad de proyecto contempla llevar a cabo la actividad de aprendizaje: "***AA5: Diseñar el modelo relacional de la base de datos y las interfaces gráficas de usuario que incluya el modelo contractual para su adquisición de acuerdo con unos términos de referencia***."

Para lograr lo planteado en la actividad de proyecto, se hace necesario que el aprendiz realice la actividad de aprendizaje establecida siguiendo las indicaciones del instructor, teniendo en cuenta la relación y la secuencia lógica relacionadas con las fases que intervienen en el desarrollo de un sistema de información. Realizando lo anteriormente mencionado, se busca formar un tecnólogo en análisis y desarrollo de sistemas de información con la capacidad de llevar a cabo las actividades que comprenden la fase de diseño, en la cual se determina la estructura lógica del sistema de información.

Para un mejor entendimiento por parte del aprendiz sobre los procesos y conocimientos asociados a esta guía de aprendizaje, en su Fase Diseño, correspondiente a la actividad de proyecto: **"*AP4. Determinar la estructura lógica del sistema de información*"**; se hace necesario estudiar y comprender el **esquema de contenidos de la guía presentada a continuación**, donde se muestran las actividades de aprendizaje a realizar, frente a los contenidos, resultados de aprendizaje, competencias y evidencias a desarrollar.



Dedicación horaria

La Actividad de Proyecto 4 tiene una duración de 302 horas y se proyecta realizar en 1,7 meses, con un promedio de trabajo académico diario de 6 a 8 horas, para el desarrollo de las actividades contará con acompañamiento de instructores de competencia técnica, de inglés y transversales. Es importante que el aprendiz organice su tiempo dada la dedicación horaria que demanda la realización de las actividades y las evidencias que se mencionan a continuación.

Evidencias
AP04-AA5-EV01. Cuestionario Técnico AP4.
AP04-AA5-EV02. Elaboración del prototipo del sistema de información a desarrollar.
AP04-AA5-EV03. Foro - ¿Cómo diseñar prototipos y modelos de bases de datos?
AP04-AA5-EV04. Elaboración de términos de referencia.
AP04-AA5-EV05. Inglés - Elaboración de un manual técnico.
AP04-AA5-EV06. Transversal - Ensayo la importancia de la comunicación asertiva.
AP04-AA5-EV07. Transversal - Foro - Aplicación de herramientas de comunicación asertiva.
AP04-AA5-EV08. Transversal. Estudio de Caso DISTRIMAY.
AP04-AA5-EV09. Transversal. Foro discusión caso DISTRIMAY.

Orientación del Instructor: para el desarrollo de las actividades de aprendizaje y sus correspondientes evidencias, el instructor orientará la revisión y posterior profundización en los contenidos y recursos didácticos (objetos de aprendizaje, videos, enlaces externos, bibliografía, otros) dispuestos en el ambiente virtual. Igualmente, brindará realimentación oportuna sobre los resultados de aprendizaje y competencias en desarrollo.

Así mismo, el instructor, es el encargado de orientar el proceso formativo mediante explicaciones específicas, aclaración de inquietudes, direccionamiento de actividades, recomendaciones para el aprendizaje y desarrollo de ejercicios prácticos; fomentando en el aprendiz una cultura de investigación, de respeto a los derechos de autor y de acción basada en la calidad.

Sesiones en línea: mediante la plataforma virtual el equipo de instructores programarán y realizarán sesiones e línea para dar a conocer el objetivo de la actividad de proyecto, su temática, la dinámica del proceso formativo, las actividades a realizar, las evidencias que se deberán aportar, su forma de entrega y los criterios de evaluación a aplicar, entre otros aspectos, como por ejemplo resolver dudas e inquietudes, explicar temas técnicos específicos, orientar en la búsqueda de información y permitir espacios para sustentación de evidencias.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

El desarrollo de las actividades de aprendizaje que se abordan en la presente actividad de proyecto, se estructura de la siguiente manera: inicialmente, se plantea ejercicio de reflexión inicial basada en preguntas integrales sobre saberes previos aplicables a la actividad de aprendizaje y una actividad de apropiación de conocimiento; se describen cada una de las actividades de aprendizaje junto con las actividades a realizar y las evidencias asociadas, se menciona los recursos requeridos y el material de estudio vinculado.

Descripción de la(s) Actividad(es)

- **Actividades de reflexión inicial**

En la actividad de aprendizaje 3 se elaboró el modelo conceptual del sistema de información en desarrollo, con base en las necesidades o requerimientos del cliente y mediante el uso e interpretación de la información levantada se elaboraron los Diagramas UML de acuerdo a las características de cada uno de ellos. Así mismo, en esta actividad de aprendizaje se realizó el Modelo de la base de datos de acuerdo a las necesidades del sistema de información en desarrollo.

En esta actividad de aprendizaje se utilizan los resultados anteriores para que apoyados en la construcción de prototipos se realice el diseño del sistema de información de acuerdo a los requerimientos funcionales del cliente.

Antes de iniciar esta etapa de diseño, es necesario que el aprendiz se plantee las siguientes preguntas e intente responder las inquietudes acordes con su nivel de conocimientos:

- ¿Cómo se crean prototipos?

- ¿Para qué sirven los prototipos?
- ¿Para el usuario final es importante ver los prototipos?
- ¿Qué son las interfaces de usuario y para qué sirven?
- ¿Qué tipos de patrones de desarrollo de software conoce?
- ¿Qué es la normalización de una base de datos y para qué sirve?
- ¿Qué es un modelo entidad relación y para qué sirve?
- ¿Qué son y para qué sirven los términos de referencia?
- ¿Por qué es importante el manual de usuario?
- ¿Qué debe contener el manual de usuario?
- ¿Cómo reacciona ante situaciones de problemas en el ámbito productivo o social?
- ¿Qué tipos de propuestas realiza que posibiliten la resolución de problemas?

Un diseño correcto del sistema permitirá al analista de sistemas realizar un desarrollo del sistema de información. Por el contrario, un deficiente diseño hará que se presenten rediseños que pueden retrasar el proyecto o aumentar su costo.

La actividad de reflexión inicial ha generado inquietudes, y muy seguramente, suscitado alternativas para el adecuado diseño de los prototipos del sistema de información que en este momento el aprendiz proyecta realizar, con el ánimo de brindar una solución específica a una necesidad o problema en particular; esta posibilidad de introspección, abre el camino hacia un espacio de contextualización en el que se acomete la identificación y rescate de los saberes que el aprendiz ha construido a lo largo de sus particulares vivencias y desempeño laboral.

Es importante tener presente que los conocimientos adquiridos mediante el estudio y aplicación de la presente guía de aprendizaje son pieza clave para determinar la estructura lógica del sistema de información, por tanto, es preciso avanzar en el proceso, teniendo en cuenta que cada elemento estudiado es un requisito en el alcance completo de las competencias que propone el programa de formación, para ello es importante que el aprendiz en formación tenga en cuenta:

- Fortalecer sus conocimientos a través de la consulta de los diferentes materiales de formación materiales y recursos que se proporcionan en el ambiente virtual; igualmente válida como herramienta de apoyo, resulta la obtención de información a partir de búsqueda y consulta propia.
- Construir todas las evidencias requeridas y aplicar el proceso de autoevaluación a través de los instrumentos de evaluación proporcionados, con el fin de verificar el cumplimiento de las exigencias pedidas.
- Publicar las evidencias en la plataforma virtual de aprendizaje con el fin de que sean evaluadas por el instructor, o equipo de instructores que apoyan la ejecución del proceso; como resultado de esta evaluación recibirá la realimentación de logros alcanzados, de tal manera que pueda reflexionar frente a su aprendizaje y avance en el proceso formativo**.**

- **Actividad de apropiación**

Para lograr el desarrollo de las actividades de aprendizaje que se proponen en la actividad de proyecto 4, es necesario que el al aprendiz realice la lectura de los materiales de formación y se apropie de los contenidos y conocimientos dispuestos en los diversos recursos que se encuentran en plataforma virtual, en cada material de formación encontrará una actividad didáctica como instrumento de realimentación que le permitirá verificar el alcance de conocimientos adquiridos, podrá revisar el mapa orientador de contenidos, analizar el glosario de términos relevantes al tema de estudio, cotejar la bibliografía para la exploración y seguimiento de la temática desarrollada y estudiar los contenidos en general.

Material de formación de competencia técnica
Interfaz gráfica de usuario.
Laboratorio. Diseño de Interfaces
Patrones de diseño.
Conversión de MER a Modelo Relacional.
Laboratorio. Modelo Relacional de la Base de Datos.
Diseño de Mecanismos de Seguridad y Control.
PDF. Términos de Referencia para contratos de Desarrollo de Software.

Material competencia inglés
Tips on grammar: present and past tenses.

Material competencia transversal
Comunicación asertiva.
Resolución de problemas.

Durante el proceso de lectura tendrá acompañamiento de los instructores, deberá ponerse en contacto para la programación de las sesiones en línea donde puede aclarar dudas e inquietudes que se presenten respecto del material de formación y las temáticas abordadas dentro de los mismos.

Laboratorio. Diseño Modelo Relacional de una Base de Datos: con este laboratorio se introduce al aprendiz con la herramienta DBdesigner con el fin de construir una base de datos por medio del modelo relacional de la base de datos. El laboratorio se compone de lo siguiente:

a. Seguir los procedimientos y explicaciones proporcionadas en los Videotutoriales para el Diseño del Modelo Relacional de una Base de Datos utilizando la Herramienta DBDesigner.

b. Desarrollar los Ejercicios de ejemplo que se muestran en los videostutoriales utilizando la herramienta DBDesigner.

Se plantean al aprendiz algunos ejercicios que deberá realizar y entregar voluntariamente al instructor para su realimentación. La realización de estos ejercicios que hacen parte del laboratorio tiene como finalidad afianzar los

conocimientos adquiridos y desarrollar mayor comprensión y práctica para alcanzar los resultados de aprendizaje planteados en esta Actividad de proyecto, por tal motivo no son actividades calificables

Laboratorio. Diseño de Interfaces: con este laboratorio se introduce al aprendiz diseñar interfaces de usuario utilizando las herramientas Balsamiq, Mockups y Gomockingbird. El laboratorio se compone de lo siguiente:

a. Revisar los objetos de contenido sobre diseño de interfaz de usuario.
b. Revisar tutorial de interfaces mediante Balsamiq, Mockups y Gomockingbird.
c. Ejercicios de ejemplo que se muestran en los objetos de contenido y tutoriales.

Se plantean al aprendiz algunos ejercicios que deberá realizar y entregar voluntariamente al instructor para su retroalimentación. La realización de estos ejercicios que hacen parte del laboratorio tiene como finalidad afianzar los conocimientos adquiridos y desarrollar mayor comprensión y práctica para alcanzar los resultados de aprendizaje planteados en esta Actividad de proyecto, por tal motivo no son actividades calificables.

Una vez haya apropiado conocimientos y conceptos proceda con la presentación de la siguiente evidencia:

Evidencia (De conocimiento)

AP04-AA5-EV01. Cuestionario Técnico AP4. Con el fin de obtener evidencia de esta actividad de apropiación, con relación a los conocimientos adquiridos por el aprendiz para el logro de los resultados de aprendizaje previstos, mediante la interacción de este con el material de estudio suministrado en cada uno de los materiales de formación dispuestos en la plataforma, se dispone de un cuestionario que determina si el aprendiz maneja adecuadamente los conceptos sobre Diseño de Interfaces de Usuario, Patrones de diseño, Conversión del Modelo Entidad Relación a Modelo Relacional, Diseño de Mecanismos de Seguridad y Control; Tips de gramática en inglés – Tiempos presente y pasado, Comunicaciones asertivas racionales orientados hacia la convivencia y resolución de problemas.

Resultados de Aprendizaje. Todos los RAP que están vinculados a la Actividad de Proyecto 4, a excepción de los RAP de competencias de inglés y transversales.

- 22050103304. Construir el prototipo del sistema de información, a partir del análisis de las características funcionales del sistema en relación con facilidad de manejo, funcionalidad y experiencia del usuario, apoyado en software aplicado según protocolos de diseño.
- 22050103306. Diseñar la estructura de datos, a partir del modelo conceptual determinado en el análisis del sistema, utilizando herramientas tecnológicas de bases de datos, según las normas y estándares establecidos.
- 22050100902. Definir estrategias para la elaboración de términos de referencia y procesos de evaluación de proveedores, en la adquisición de tecnología, según protocolos.

Forma de entrega: en el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / **AP04-AA5-EV01. Cuestionario Técnico AP4.**

- **ACTIVIDAD DE APRENDIZAJE 5**. Diseñar el modelo relacional de la base de datos y las interfaces gráficas de usuario que incluya el modelo contractual para su adquisición de acuerdo con unos términos de referencia.

Con base en el modelo conceptual diagrama entidad relación y los casos de uso que se levantaron en la actividad de proyecto anterior, el aprendiz deberá realizar el modelo relacional de la base de datos, diseñar las interfaces de usuario del sistema de información en desarrollo, así como, elaborar documento de términos de referencia y evaluación de ofertas, que contendrá los siguientes elementos:

- Modelo relacional de la base de datos.
- Interfaces de usuario.
- Términos de referencia.
- Evaluación de ofertas y proveedores.

Uno de los pasos iniciales para la construcción de una base de datos es el modelado de ellos por medio de un Modelo Entidad Relación (MER), modelo que permite por medio de entidades, atributos y relaciones acercarse a una representación preliminar del diseño de una base de datos.

Luego de terminar un MER debe entrarse en el diseño de un modelo lógico, el cual es llamado modelo relacional y en donde se incluyen relaciones lógicas entre los datos. Por último, se llega a la implementación de la base de datos por medio de un Gestor de base de datos (Modelo físico). Con respecto a las interfaces de usuario, estas se convierten en parte esencial del proceso de diseño general del software y estas deben de ser diseñadas para que exista una buena interacción del usuario y una buena presentación de la información.

Por último, el aprendiz debe de conocer los componentes básicos de los términos de referencia en procesos de negociación de bienes informáticos y utilizar plantilla de evaluación de ofertas y proveedores.

A continuación, se describen las actividades a realizar y las evidencias asociadas que el aprendiz debe entregar.

Actividad. Modelo relacional de bases de datos e interfaces del SI.

Con base en las técnicas aprendidas en los materiales de formación “Conversión de MER a Modelo Relacional”, el aprendiz deberá entregar la lista de las tablas, campos, llaves primarias y llaves foráneas de la base de datos del sistema de información a desarrollar.

Teniendo en cuenta las técnicas presentadas en el material de formación “Interfaz gráfica de usuario”, el aprendiz deberá elaborar las interfaces gráficas del sistema de información que se encuentra desarrollando.

Paso 1. Elaboración del Modelo relacional de la base de datos

A partir del modelo entidad relación de la actividad de proyecto 3 y las técnicas revisadas en los recursos de la actividad de proyecto 3, el aprendiz deberá elaborar el modelo relacional, es decir, las tablas, campos, llaves primarias, llaves foráneas del sistema de información a desarrollar.

Lo anterior se deberá entregar en formato de hoja electrónica con la siguiente estructura:

Tabla		**Nombre de tabla**		
No	**Nombre Campo**	**Tipo**	**Longitud**	**Tipo llave**

Paso 2. Elaboración de las interfaces

Con base en los casos de uso que se levantaron en la actividad de proyecto el aprendiz deberá crear las interfaces de usuario para cada uno de ellos.

Las interfaces las podrá hacer el aprendiz en cualquiera de las herramientas revisadas, después de diseñadas deberá exportarlas a un formato de imagen PNG o JPEG. Con este último archivo el aprendiz deberá diligenciar un documento de procesador de texto con la siguiente estructura:

Nombre Caso de uso	
<< Imagen de la interfaz diseñada >>	

Lo anterior se requiere para cada interfaz de usuario diseñada.

Evidencia (De producto)

AP04-AA5-EV02. Elaboración del prototipo del sistema de información a desarrollar. Para cumplir con esta evidencia, es importante que haya realizado la actividad de apropiación referida a la comprensión al material de estudio presentando en esta guía. De acuerdo con las indicaciones de su instructor, posteriormente debe ingresar y entregar la actividad (evidencia) desarrollada en la plataforma.

Resultados de Aprendizaje

22050103304. Construir el prototipo del sistema de información, a partir del análisis de las características funcionales del sistema en relación con facilidad de manejo, funcionalidad y experiencia del usuario, apoyado en software aplicado según protocolos de diseño.

22050103306. Diseñar la estructura de datos, a partir del modelo conceptual determinado en el análisis del sistema, utilizando herramientas tecnológicas de bases de datos, según las normas y estándares establecidos.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- **Formato para desarrollo de evidencia:** AP04-AA5-EV02-Elaboración-prototipo-SI.
- **Instrumento de evaluación:** IE-AP04-AA5-EV02-Elaboración-prototipo-SI.

Forma de entrega

Debe enviar el archivo a través de la plataforma virtual en el área correspondiente indicada por el instructor encargado. Al momento de realizar la entrega tenga en cuenta los siguientes pasos:

En el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / AP04-AA5-EV02. Elaboración del prototipo del sistema de información a desarrollar. En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Actividad. Diseñando prototipos y modelos de bases de datos eficaces que contemplen las necesidades del sistema de información.

Una vez se hayan apropiado los conocimientos provistos en los materiales de la actividad de proyecto 4 y se hayan realizado las evidencias de conocimiento y producto el aprendiz deberá participar en un foro para responder a las siguientes preguntas:

- ¿Por qué cree usted que es importante la creación de prototipos del sistema de información para el usuario final?
- ¿Es posible representar la totalidad de los datos de una empresa con el modelo relacional?
- ¿Sin la realización de prototipos se puede hacer un buen diseño de las interfaces del sistema?
- ¿Cómo manejar los datos no estructurados como fotos y videos?

- ¿Cómo hacer las interfaces de usuario más amigables e intuitivas?
- ¿Cómo hacer interfaces de usuario para personas con limitaciones visuales?
- ¿Cómo aplicar el patrón MVC en el desarrollo de un sistema? presente un ejemplo.
- ¿Es aconsejable permitir la redundancia de datos para aumentar el rendimiento del sistema?
- ¿Cómo debe interactuar el usuario con el sistema informático?
- ¿Cómo se debe presentar la información del sistema informático al usuario?

Posteriormente debe ingresar al foro que se encuentra en el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / AP04-AA5-EV03. Foro - ¿Cómo diseñar prototipos y modelos de bases de datos?, allí encontrará el enlace para participar en el foro temático donde debe socializar sus respuestas con sus compañeros.

Además de hacer su participación, revisar los aportes de al menos dos compañeros a quienes deberá formular realimentación, argumentación o inquietudes sobre sus participaciones y propuestas planteadas.

Evidencia (De desempeño)

AP04-AA5-EV03. Foro - ¿Cómo diseñar prototipos y modelos de bases de datos? De acuerdo con la experiencia del aprendiz en determinar la estructura lógica del sistema de información, se plantea este foro que pretende generar discusión acerca de la construcción del prototipo del sistema de información y el diseño de la estructura de las bases de datos con base en los requerimientos funcionales del sistema.

Resultados de aprendizaje:

- 22050103304. Construir el prototipo del sistema de información, a partir del análisis de las características funcionales del sistema en relación con facilidad de manejo, funcionalidad y experiencia del usuario, apoyado en software aplicado según protocolos de diseño.
- 22050103306. Diseñar la estructura de datos, a partir del modelo conceptual determinado en el análisis del sistema, utilizando herramientas tecnológicas de bases de datos, según las normas y estándares establecidos.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- Formato para desarrollo de evidencia: AP04-AA5-EV03-Foro- Diseño-Prototipo-Base-Datos-SI.
- Instrumento de evaluación: IE-AP04-AA5-EV03-Foro- Diseño-Prototipo-Base-Datos-SI.

Forma de entrega: en el menú principal del programa de formación ingrese a Fase Diseño / Actividad de proyecto 4 / Evidencias / AP04-AA5-EV03. Foro - ¿Cómo diseñar prototipos y modelos de bases de datos?

Actividad. Elaboración de términos de referencia en procesos de negociación de bienes informáticos.

Con base en la lectura del contenido "PDF. Términos de Referencia para contratos de Desarrollo de Software", el aprendiz con base en los requerimientos del sistema de información deberá elaborar los componentes básicos de un documento de términos de referencia que permitan realizar la negociación de bienes informáticos.

Con base en el archivo de evaluación de propuestas proporcionado, el aprendiz deberá evaluar y calificar ofertas y proveedores de acuerdo a las necesidades informáticas de la organización.

Paso 1. Documento de Términos de referencia.

A partir de la lectura y ejemplo propuesto en la Infografía Términos de Referencia para contratos de Desarrollo de Software, el aprendiz deberá elaborar un documento que tenga los términos de referencia con base en los requerimientos del sistema de información y las necesidades de la organización, con la siguiente estructura:

Componentes de un documento de Términos de Referencia:

- Objetivos.
- Fuente de recursos.
- Proponentes habilitados o restricciones.
- Idioma.
- Actividades a realizar.
- Entregables, Hitos o Fechas de entrega y detalle de los entregables.
- Equipo de trabajo.
- Duración del contrato.
- Sede o localización de las labores a realizar.
- Documentos a presentar.
- Proceso para seleccionar al proveedor.
- Forma de pago.
- Confidencialidad de la información.

Paso 2. Evaluación de propuestas.

Con base en el archivo de Excel proporcionado "Evaluación de propuestas" el aprendiz deberá de evaluar y calificar las propuestas de proveedores. A continuación, se detalla el uso de esta plantilla de calificación:

- Encabezado: en esta sección se debe de colocar información referente al nombre de la organización, nombre del proyecto, el bien informático o servicio a contratar, la fecha de elaboración y datos del gerente de proyecto y/o líder técnico en caso de que aplique.

1							**EVALUACIÓN DE PROPUESTAS**												VERSION 1.0
2							CODIGO:												Pag. 1 de __
3	**NOMBRE PROYECTO :**												**EMPRESA**						
4	**I. DATOS DE LA EVALUACIÓN**																		
5	**BIEN O SERVICIO A CONTRATAR:**																	**FECHA:**	
6	**GERENTE DE PROYECTO:**											**LÍDER TÉCNICO:**							

- Evaluación de requerimientos funcionales: Esta sección describe los requerimientos funcionales que debe de cumplir o solucionar el bien informático a adquirir.

	II. EVALUACIÓN DE LOS REQUERIMIENTOS FUNCIONALES																	
	Item		**% Cap.**	**Peso del Item**	**<Nombre Proponente 1>**		**<Nombre Proponente 2>**		**<Nombre Proponente 3>**		**<Nombre Proponente 4>**		**<Nombre Proponente 5>**		**Comentarios**			
1	<Requerimiento funcional 1>																	
2	<Requerimiento funcional 2>																	
3	<Requerimiento funcional 3>																	
4	.....																	
5	<Requerimiento funcional n>																	
6																		
7																		
8																		
9																		
10																		
Sub Totales				**0,00%**	**#**	**#**	**0,00%**	**#**	**#**	**0,00%**	**#**	**#**	**0,00%**	**-**	**0,00%**	**-**	**0,00%**	

  - columna Ítem, se describe cada uno de los requerimientos funcionales
  - columna % Cap. el archivo tiene 4 capítulos (Evaluación de Requerimientos Funcionales, Evaluación de Requerimientos No Funcionales, Evaluación de Aspectos Financieros, Evaluación de Aspectos Generales). En esta columna se debe de especificar el peso o valor de cada capítulo. Por ejemplo, la distribución de esta columna puede quedar de esta manera:
    - Evaluación Req. Funcionales: 40%
    - Evaluación Req. No Funcionales: 30%
    - Evaluación Aspectos Financieros: 15%
    - Evaluación Aspectos Generales: 15%

    Para un total del 100%
  - Peso del Ítem: en esta columna colocar el peso porcentual de cada Ítem dentro del Capítulo. Por ejemplo: requerimiento Funcional 1 tiene un peso del 10%, Requerimiento funcional 2 tiene un peso del 15% y así sucesivamente hasta completar el 100%.

    Este valor porcentual depende de la importancia de que ese requerimiento en particular sea provisto por el proveedor.
  - Nombre del proponente: en esta columna se especifica el nombre de cada uno de los potenciales proveedores.

- Evaluación de requerimientos no funcionales: esta sección describe los requerimientos no funcionales o requerimientos técnicos que debe de cumplir o solucionar el bien informático a adquirir.

III. EVALUACIÓN DE LOS REQUERIMIENTOS NO FUNCIONALES														
	Item	% Cap.	Peso del Item	<Nombre Proponente 1>		<Nombre Proponente 2>		<Nombre Proponente 3>		<Nombre Proponente 4>		<Nombre Proponente 5>		Comentarios
1	<Requerimiento no funcional 1>													
2	<Requerimiento no funcional 2>													
3	<Requerimiento no funcional 3>													
4	.....													
5	<Requerimiento no funcional n>													
6														
7														
8														
9														
10														
11														
	Sub Totales		0,00%	#	#	0,00%	#	#	0,00%	#	#	0,00%	-	0,00%	-	0,00%	

- columna Ítem, se describe cada uno de los requerimientos no funcionales o requerimientos técnicos.
- columna % Cap. el archivo tiene 4 capítulos (Evaluación de Requerimientos Funcionales, Evaluación de Requerimientos No Funcionales, Evaluación de Aspectos Financieros, Evaluación de Aspectos Generales). En esta columna se debe de especificar el peso o valor de cada capítulo. Por ejemplo, la distribución de esta columna puede quedar de esta manera:
  - Evaluación Req. Funcionales: 40%
  - Evaluación Req. No Funcionales o Técnicos: 30%
  - Evaluación Aspectos Financieros: 15%
  - Evaluación Aspectos Generales: 15%

  Para un total del 100%
- Peso del Ítem: en esta columna colocar el peso porcentual de cada Ítem dentro del Capítulo. Por ejemplo: Requerimiento no funcional 1 tiene un peso del 10%, Requerimiento no funcional 2 tiene un peso del 15% y así sucesivamente hasta completar el 100%.
  Este valor porcentual depende de la importancia de que ese requerimiento en particular sea provisto por el proveedor.
- Nombre del proponente: en esta columna se especifica el nombre de cada uno de los potenciales proveedores.

- Evaluación de Aspectos Financieros y Aspectos Generales: estas secciones describen los aspectos financieros de cada proveedor a tener en cuenta en su calificación y aspectos generales como experiencia-casos de éxito, equipo de trabajo, garantías, entre otros.

IV. EVALUACIÓN DE ASPECTOS FINANCIEROS														
	Item	% Cap.	Peso del Item	<Nombre Proponente 1>		<Nombre Proponente 2>		<Nombre Proponente 3>		<Nombre Proponente 4>		<Nombre Proponente 5>		Comentarios
1	Capital de Trabajo (AC - PC)													
2	Relación Precio de Venta / Ventas													
3	Indice de Endeudamiento (PT / AT)													
4	Indice de Liquidez (AC / PC)													
5	Precio de Venta													
	Sub Totales		0,00%	#	#	0,00%	#	#	0,00%	#	#	0,00%	-	0,00%	-	0,00%	
V. EVALUACIÓN DE ASPECTOS GENERALES														
	Item	% Cap.	Peso del Item	<Nombre Proponente 1>		<Nombre Proponente 2>		<Nombre Proponente 3>		<Nombre Proponente 4>		<Nombre Proponente 5>		Comentarios
1	Experiencia Previa - Casos de Éxito													
2	Equipo de Trabajo													
3	Tiempo de Garantía													
4	Servicio de Soporte													
5	<Registre Otro Aspecto General 1>													
6	...													
7	<Registre Otro Aspecto General n>													
	Sub Totales		0,00%	#	#	0,00%	#	#	0,00%	#	#	0,00%	-	0,00%	-	0,00%	
	TOTALES	**0%**		**0,00%**		**0,00%**		**0,00%**		**0,00%**		**0,00%**		
	Orden de Eligibilidad *			**1**		**1**		**1**		**1**		**1**		

Al final, la plantilla realiza la respectiva calificación con base en los datos suministrados generando un Orden de Elegibilidad.

PRODUCTO(S) ENTREGABLE(S)

Se entregarán dos archivos realizados en los siguientes programas que son:

1. Documento en Word con los Términos de Referencia.
2. Hoja electrónica con la plantilla de Evaluación de propuestas diligenciado.

Evidencia (De producto)

AP04-AA5-EV04. Elaboración de términos de referencia. Para cumplir con esta evidencia, es importante que haya realizado la actividad de apropiación referida a la comprensión al material de estudio presentando en esta guía. De acuerdo con las indicaciones de su instructor, posteriormente debe ingresar y entregar la actividad (evidencia) desarrollada en la plataforma.

Resultados de Aprendizaje.

- 22050100902. Definir estrategias para la elaboración de términos de referencia y procesos de evaluación de proveedores, en la adquisición de tecnología, según protocolos.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- **Formato para desarrollo de evidencia:** AP04-AA5-EV04-Elaboración-Términos-Referencia.
- **Instrumento de evaluación:** IE-AP04-AA5-EV04-Elaboración-Términos-Referencia.
- **Documento:** plantilla en Excel para evaluación de propuestas.

Forma de entrega

Debe enviar el archivo a través de la plataforma virtual en el área correspondiente indicada por el instructor encargado. Al momento de realizar la entrega tenga en cuenta los siguientes pasos:

En el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias /**AP04-AA5-EV04. Elaboración de términos de referencia.** En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Actividad. Elaboración de un manual técnico en inglés.

Leer y escribir en una lengua extranjera pueden llegar a ser tareas un poco complejas si no se desarrollan y cultivan las habilidades lingüísticas necesarias que le permitan decodificar e interpretar correctamente la información contenida en un texto para, posteriormente, a partir de esta, construir nuevos conocimientos y significados mediante la producción de textos de mayor complejidad.

Como analista y desarrollador de sistemas de información, una de las labores que muy seguramente deberá realizar es la elaboración de los manuales de usuario que le ayudarán a sus clientes a comprender la forma en la que se debe operar y controlar el sistema de información que usted ha diseñado para ellos. En este sentido, los manuales de usuario se convierten en herramientas considerablemente importantes, ya que proporcionan información de gran relevancia para los usuarios finales sobre "cómo usar el sistema de información".

Un manual estructurado y redactado de manera adecuada puede ser tremendamente beneficioso, en tanto que permite limitar las posibilidades de que los usuarios hagan un mal uso del sistema, lo que puede contribuir en gran medida a ahorrar tiempo de ingeniería y soporte técnico, favoreciendo así la imagen de su compañía o de la compañía que provee el sistema de información.

Para el desarrollo de esta evidencia, usted deberá elaborar en inglés el manual de usuario de la interfaz del sistema de información en el que ha venido trabajando en el área técnica. Para este momento, usted ya debe estar por completar el diseño de dicha interfaz, por lo que cuenta con el insumo suficiente para poder empezar la construcción del manual correspondiente.

La evidencia consiste entonces en planear, estructurar y redactar el manual de usuario de la interfaz del sistema de información en un documento Word, para lo cual deberá seguir las instrucciones que se muestran a continuación:

1. Consultar las pautas para la redacción de manuales que se presentan en el material de formación *Grammar tips: present and past tenses.* En dicho material podrá encontrar recomendaciones generales sobre cómo planear, estructurar y redactar su manual de usuario.
2. Construir un glosario de términos técnicos que considere útiles para la redacción de su manual de usuario.
3. Redactar el manual de usuario teniendo en cuenta lo descrito en las pautas proporcionadas en el material de formación relacionado anteriormente.

Al finalizar, deberá guardar el documento y adjuntarlo para enviarlo mediante el enlace de la evidencia mencionada.

LO QUE DEBE CONTENER LA EVIDENCIA (PARTES DEL INFORME O PASOS)

Para desarrollar la evidencia de elaboración de un manual técnico en inglés, tenga en cuenta las siguientes recomendaciones:

1. Consulte las pautas para la elaboración de manuales de usuario disponibles en el material de formación *Grammar tips: present and past tenses*.
2. Planee la estructura y la secuencia de contenidos que presentará en su manual. Para ello identifique el tipo y las características del usuario al que va dirigido.
3. Identifique todos los pasos o tareas que debe seguir el usuario para operar el sistema de información. Haga una lista de ellas.
4. Defina cómo se presentará gráficamente el manual.
5. Incluya información textual y gráfica en el manual suficiente para poder ilustrar a los usuarios sobre el manejo del sistema de información.

6. Preste atención a las recomendaciones de redacción aportadas y téngalas en cuenta durante la elaboración del manual.
7. Sea claro y concreto con la información que presentará en su manual.
8. Lea detenidamente su manual y haga control de calidad a los contenidos y la redacción del mismo. Trate de identificar durante esta revisión aquellos aspectos que pueden no ser lo suficientemente comprensibles para los usuarios.
9. Guarde el documento, adjúntelo y envíelo a su instructor mediante el enlace dispuesto en plataforma.

PRODUCTO(S) ENTREGABLE(S)

Manual de usuario para la interfaz gráfica del sistema de información diseñada en las actividades del área técnica.

Evidencia (De producto - inglés):

AP04-AA5-EV05. Inglés - Elaboración de un manual técnico. Para el desarrollo de esta evidencia es necesario tener en cuenta las indicaciones anteriormente descritas.

Resultados de aprendizaje:

- 24020150203. Comprender una amplia variedad de frases y vocabulario en inglés sobre temas de interés personal y temas técnicos.
- 24020150207. Identificar formas gramaticales básicas en textos y documentos elementales escritos en inglés.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- Formato para desarrollo de evidencia: AP04-AA5-EV05-Inglés-Elaboración-Manual-Técnico.
- Instrumento de evaluación: IE-AP04-AA5-EV05-Inglés-Elaboración-Manual-Técnico.

Forma de entrega: en el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / AP04-AA5-EV05. Inglés - Elaboración de un manual técnico. En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Actividad. La comunicación asertiva y procesos de comunicación eficaz.

El objetivo de esta actividad es que el aprendiz realice una apropiación de los conceptos presentados en el material de formación llamado “Comunicación Asertiva” y el material complementario dispuesto en plataforma. Después de revisar el material de estudio deberá redactar un ensayo que responda a una pregunta planteada, Dicho ensayo debe contener argumentos sólidos sobre su punto de vista con respecto al tema revisado.

Evidencia (De producto)

AP04-AA5-EV06. Transversal - Ensayo la importancia de la comunicación asertiva. Escrito a manera de ensayo sobre procesos comunicativos eficaces y asertivos que posibiliten la convivencia y el establecimiento de acuerdos, que responda a la pregunta planteada ¿cuál es la importancia de la comunicación asertiva en el desarrollo de sus funciones dentro de una organización? Debe plasmar ideas claras y argumentos que soporten su punto de vista. Para la realización de esta evidencia es necesario que lea con mucha atención el material de formación Comunicación Asertiva y el material complementario. Luego deberá redactar un ensayo que responda a la siguiente pregunta ¿Cuál es la importancia de la comunicación asertiva en el desarrollo de sus funciones dentro de una organización? Para redactar dicho ensayo es necesario que argumente su punto de vista. Debe consultar información adicional al material de formación y utilizar mínimo cuatro citas que le sirvan para sustentar lo que plantea. La extensión mínima del ensayo debe ser mín. 3 páginas máx. 5 páginas.

Resultados de Aprendizaje.

- 24020150002. Desarrollar procesos comunicativos eficaces y asertivos dentro de criterios de racionalidad que posibiliten la convivencia, el establecimiento de acuerdos, la construcción colectiva del conocimiento y la resolución de problemas de carácter productivo y social.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- **Formato para desarrollo de evidencia:** AP04-AA5-EV06.Transversal-Ensayo-comunicación-asertiva.
- **Instrumento de evaluación:** IE-AP04-AA5-EV06.Transversal-Ensayo-comunicación-asertiva.

Forma de entrega

Debe enviar el archivo a través de la plataforma virtual en el área correspondiente indicada por el instructor encargado. Al momento de realizar la entrega tenga en cuenta los siguientes pasos:
En el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / **AP04-AA5-EV06. Transversal - Ensayo la importancia de la comunicación asertiva.** En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Actividad. Aplicación de herramientas de la comunicación asertiva

El objetivo de la siguiente evidencia es que el aprendiz realice una apropiación y aplicación de los temas revisados en el material de formación “Comunicación asertiva” y el material complementario.

Evidencia de desempeño

AP04-AA5-EV07. Transversal - Foro - Aplicación de herramientas de comunicación asertiva. Participación en el foro mediante la entrega de documento con tres situaciones problémicas, con su respectivo abordaje y las dos que

seleccionó de sus compañeros, deberá justificar por qué razón seleccionó esos aportes de sus compañeros. Para la realización de esta evidencia en necesario que lea con mucha atención el material de formación "Comunicación Asertiva" y el material complementario. Luego deberá:

1. Para ello deberá plantear ***tres situaciones problemáticas hipotéticas*** que se puedan presentar en su entorno de trabajo y en las cuales pueda aplicar las herramientas de la comunicación asertiva.
2. Es necesario que describa claramente la situación, está deberá contener los actores principales y la descripción de los hechos.
3. Deberá explicar paso por paso cómo resolvería cada una de las situaciones que planteó utilizando las herramientas de la comunicación asertiva.
4. Deberá socializar dichas situaciones con su respectiva manera de abordarla en el Foro: Comunicación asertiva.
5. Realimentará utilizando la rúbrica TIGRE los aportes de tres compañeros en el foro: Comunicación asertiva.
6. Luego seleccionará dos situaciones de sus compañeros que desde su punto de vista cumplen con lo solicitado y hará un compendio en un documento.
7. El documento contendrá sus tres situaciones y las dos que seleccionó de sus compañeros, deberá justificar por qué razón seleccionó esos aportes de sus compañeros. (deberá contener: portada, contenido y conclusiones)
8. Hacer entrega en plataforma dentro del mismo foro del documento relacionado con el anterior punto.

Resultados de Aprendizaje.

- 24020150002. Desarrollar procesos comunicativos eficaces y asertivos dentro de criterios de racionalidad que posibiliten la convivencia, el establecimiento de acuerdos, la construcción colectiva del conocimiento y la resolución de problemas de carácter productivo y social.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- **Formato para desarrollo de evidencia:** AP04-AA5-EV07-Aplicar-herramientas-comunicación-asertiva.
- **Instrumento de evaluación:** IE-AP04-AA5-EV07-Aplicar-herramientas-comunicación-asertiva.

Forma de entrega

Debe enviar el archivo a través de la plataforma virtual en el área correspondiente indicada por el instructor encargado. Al momento de realizar la entrega tenga en cuenta los siguientes pasos:
En el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / **AP04-AA5-EV07. Transversal - Foro - Aplicación de herramientas de comunicación asertiva.** En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Actividad. Análisis de situaciones críticas para dar soluciones oportunas

Esta actividad le permite al aprendiz analizar situaciones críticas para dar soluciones oportunas a problemáticas presentadas en el ámbito productivo y social. En el marco de un caso donde se describe una situación se hace necesario identificar todos los elementos que se presentan en la problemática, con el fin de proponer las soluciones

a implementar para dicha situación. Por tal motivo es importante revisar los materiales complementarios y el material de formación "Resolución de Problemas" en los cuales se muestra información pertinente para el desarrollo de esta actividad.

Evidencia (De producto)

AP04-AA5-EV08. Transversal. Estudio de Caso DISTRIMAY. Para la realización de esta evidencia es necesario que lea con mucha atención el material de formación "Resolución de Problemas" y el material complementario, deberá tener en cuenta el siguiente procedimiento para su desarrollo y entrega.

1. Como técnica inicie leyendo cuidadosamente el problema, apóyese en el material de formación allí encontrará unas orientaciones sobre este aspecto, que le ayudarán a Identificar el problema. Acá puede emplear un documento Word o Excel para que vaya registrando las variables encontradas en la comprensión de la problemática y así pueda visualizar y llegue a la identificación del problema.
2. Continúe con la siguiente etapa, que consiste en Plantear el problema, es importante que tenga claramente identificada la problemática, para que pueda plantear y más adelante diseñar y definir la(s) estrategia(s) a seguir. De lo contrario se enfocará en rutas que no lo llevaran a las soluciones del problema. Para plantear el problema deberá formularse una pregunta que lo lleve a generar posibles soluciones.

En este paso el aprendiz define esa pregunta problemática.

3. Cuando esté en la etapa de Diseño de la(s) Estrategia(s) que considera pertinentes para la resolución de la pregunta problemática, para que pueda diseñar esas estrategias, es importante primero definir:

   Cuál va a ser el objetivo general y los objetivos específicos.

4. Definir la Estrategia. Una vez tenga definido el objetivo general y los específicos, es el momento crucial para definir la Estrategia más acertada o efectiva que lo llevará a lograr el resultado esperado, para lograr definir esas estrategias el aprendiz debe analizar cuáles son las mayores debilidades que presenta la situación y cuáles son sus fortalezas que le permitirán llegar a la solución del problema**.** Por lo tanto, deben analizar muy bien estos interrogantes ¿cuándo se hará?, ¿quién lo llevará a cabo?, ¿cómo se ejecutará?, ¿con qué recursos se cuenta? y al dar respuesta a estos interrogantes podrá construir sus estrategias de solución a la problemática

   ¿Cómo lo llevará a cabo?
   ¿Cuándo se hará?
   ¿Quién lo llevará a cabo?
   ¿Cómo se ejecutará?
   ¿Con qué recursos cuenta?

5. **Diseñar la Planeación,** programando cuales son las actividades, los roles que va a cumplir cada actor, en qué momento o tiempo, de qué recursos dispone. Todos los parámetros para diseñar la planeación los encuentra especificados en el material de formación "Resolución de Problemas".

6. En este momento ya empieza a desarrollar el **Plan de Acción,** que consiste en definir los lineamientos a seguir para definir la problemática planteada**,** recuerde que el plan no solo es el hacer, debe tener en cuenta otros aspectos y variables que son fundamentales para su desarrollo, los cuales se encuentran descritos en el material de formación "Resolución de Problemas" Es muy importante estar atento al cumplimiento de la planeación para evitar desviarse de la ruta definida a seguir.

7. Por último, viene la etapa de la **Evaluación de los Resultados** obtenidos, donde va a observar las mejoras de la situación problemática presentada inicialmente, también va a afianzar más los conocimientos y muy seguramente aprenderá de las experiencias vividas durante este proceso.

   Para realizar esta Evaluación de Resultados el Aprendiz debe leer y seguir los lineamientos descritos en el Material de formación "Resolución de Problemas".

Resultados de Aprendizaje.

- 24020150014. Asumir actitudes críticas, argumentativas y propositivas en función de la resolución de problemas de carácter productivo y social.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

- **Formato para desarrollo de evidencia:** AP04-AA5-EV08-Caso-estudio-Distrimay.
- **Instrumento de evaluación:** IE-AP04-AA5-EV08-Caso-estudio-Distrimay.

Forma de entrega

Debe enviar el archivo a través de la plataforma virtual en el área correspondiente indicada por el instructor encargado. Al momento de realizar la entrega tenga en cuenta los siguientes pasos:
En el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / **AP04-AA5-EV08. Transversal. Estudio de Caso DISTRIMAY.** En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Actividad. Análisis problemas y discusión de soluciones sobre situaciones críticas.

Continuando con el estudio de caso DISTRIMAY, se plantea la discusión sobre el mismo, dicha participación se realizará en foro de discusión entre aprendices e instructor, es importante que una vez se hayan apropiado los conocimientos provistos en los materiales de la actividad de proyecto 4 y se hayan realizado las evidencias de conocimiento y producto de Competencia Transversal, el aprendiz participe en el Foro discusión caso -DISTRIMAY donde exprese sus opiniones y argumente objetivamente los criterios presentados para la solución del Caso "Distrimay", además opine sobre los argumentos expresados por los compañeros, manifestando si está o no de acuerdo y porqué.

La discusión o debate en el Foro debe iniciarse a partir de la problemática a resolver planteándose esta pregunta:

¿El señor Analista frente a la situación planteada que analizó, propuso las soluciones pertinentes para el Caso Distrimay? Argumente sus respuestas basado en los conocimientos que adquirió a través del material de Formación y las consultas que realizó en otras fuentes confiables.

Evidencia de desempeño

AP04-AA5-EV09. Transversal. Foro discusión caso DISTRIMAY. Para la realización de esta evidencia es necesario haber analizado en caso propuesto en la evidencia anterior, posterior a ello deberá participar en el foro de discusión llamado Foro discusión caso DISTRIMAY, tenga en cuenta que podrá interactuar con sus compañeros opinando y debatiendo sobre los criterios y argumentos presentados por cada uno de ellos.

Resultados de aprendizaje:

- 24020150014. Asumir actitudes críticas, argumentativas y propositivas en función de la resolución de problemas de carácter productivo y social.

PRODUCTO(S) ENTREGABLE(S)

1. Ingrese al menú principal del programa de formación en: Fase Diseño / Actividad de proyecto 4 / Evidencias / AP04-AA5-EV09. Transversal. Foro discusión caso DISTRIMAY.
2. Participe en el Foro según las instrucciones presentadas a continuación.
3. Responda las siguientes preguntas, las cuales son tema central del debate.

Situación 1

a. ¿Considera usted que el señor “Analista” identificó correctamente el problema?
b. ¿Cree usted que el señor “Analista” planteó bien el Problema?

Situación 2

c. ¿Cree usted que el señor “Analista” diseñó correctamente la Estrategia a seguir?
d. ¿Considera que la estrategia definida por el “Analista” es la más coherente?

Situación 3

e. ¿En su opinión el señor “Analista” implementó bien el Plan de Acción?
f. ¿Cómo califica usted los resultados obtenidos con la estrategia empleada?

4. Después lea cuidadosamente las publicaciones de sus compañeros (as) y comente respetuosamente dos (2) que llamen su atención.

5. Argumentar porque está o no de acuerdo con sus aportes. De esta forma, se generará una dinámica de construcción colaborativa de conocimiento de gran valor.

Resultados de Aprendizaje.

- 24020150002. Asumir actitudes críticas, argumentativas y propositivas en función de la resolución de problemas de carácter productivo y social.

Documentos asociados a la evidencia: siga las pautas y orientaciones consignadas en los siguientes documentos para el adecuado desarrollo de la evidencia.

Formato para desarrollo de evidencia: AP04-AA5-EV09-Transversal-Foro-Dscusion-caso-DISTRIMAY.
Instrumento de evaluación: IEAP04-AA5-EV09-Transversal-Foro-Dscusion-caso-DISTRIMAY.

Forma de entrega

Debe enviar el archivo a través de la plataforma virtual en el área correspondiente indicada por el instructor encargado. Al momento de realizar la entrega tenga en cuenta los siguientes pasos:
En el menú principal del programa de formación ingrese a: Fase Diseño / Actividad de proyecto 4 / Evidencias / **AP04-AA5-EV09. Transversal. Foro discusión caso DISTRIMAY.** En caso de dificultades comuníquese oportunamente con el instructor a cargo.

Ambiente Requerido

Para el desarrollo de las actividades y evidencias es necesario contar con acceso a plataforma virtual para revisión de contenidos, ámbito laboral para identificación de necesidad o problema a solucionar, ambientes de formación del SENA cuando sea necesario.

Materiales

Los materiales de estudio se encuentran en plataforma virtual, en cada material de formación encontrará una actividad de didáctica como instrumento de realimentación que le permitirá verificar el alcance de conocimientos adquiridos, podrá revisar el mapa orientador de contenidos, analizar el glosario de términos relevantes al tema de estudio, cotejar la bibliografía para la exploración y seguimiento de la temática desarrollada y estudiar los contenidos en general. Recuerde que la revisión de los materiales de estudio es fundamental para su proceso formativo.

Herramientas informáticas necesarias para realizar las actividades propuestas:

- Procesador de Texto y Hoja electrónica.
- Herramienta DBdesigner (versión gratuita de diseño de 2 modelos de bases de datos y 30 tablas por cada modelo).
- Herramientas Balsamiq con plugin Mockups (versión gratuita por 30 días).
- Herramienta Gomockingbird (versión gratuita).

4. ACTIVIDADES DE EVALUACIÓN

Evidencias de aprendizaje	Criterios de evaluación	Técnicas e instrumentos de evaluación
EVIDENCIA DE CONOCIMIENTO AP04-AA5-EV01. Cuestionario Técnico AP4.	Elabora el prototipo, representando la funcionalidad del sistema de información, para dar respuesta a las necesidades de la organización, según protocolos de diseño. Diseña la estructura de la base de datos, que cumpla con las necesidades del sistema de información, utilizando herramientas tecnológicas, de acuerdo con el modelo definido y siguiendo normas técnicas. Identifica los pasos para la elaboración de licitaciones y contratos informáticos, detallando las cláusulas fundamentales, calidad, entrega, garantías, licencias y planes de capacitación, de acuerdo con las políticas de la empresa y con la ley de contratación. Determina los deberes y derechos en una negociación de bienes informáticos, derechos de autor y licencias, que permitan realizar la negociación en tecnología, siguiendo los parámetros jurídicos y políticas de la organización. Aporta elementos en la construcción colectiva del conocimiento según la naturaleza del problema.	Enlace a prueba de conocimiento en LMS.
EVIDENCIA DE PRODUCTO AP04-AA5-EV02. Elaboración del prototipo del sistema de información a desarrollar.	Elabora el prototipo, representando la funcionalidad del sistema de información, para dar respuesta a las necesidades de la organización, según protocolos de diseño. Diseña la estructura de la base de datos, que cumpla con las necesidades del sistema de información, utilizando herramientas tecnológicas, de acuerdo	Formato para desarrollo de evidencia: AP04-AA5-EV02-Elaboración-prototipo-SI Instrumento de evaluación: IE-AP04-AA5-EV02-Elaboración-prototipo-SI

	con el modelo definido y siguiendo normas técnicas.	
EVIDENCIA DE PRODUCTO AP04-AA5-EV04. Elaboración de términos de referencia.	Diseña, elabora y diligencia formatos para evaluar y calificar ofertas y proveedores, de acuerdo a las necesidades informáticas de la organización, siguiendo las políticas de la empresa en los procesos de negociación en tecnología. Identifica los pasos para la elaboración de licitaciones y contratos informáticos, detallando las cláusulas fundamentales, calidad, entrega, garantías, licencias y planes de capacitación, de acuerdo con las políticas de la empresa y con la ley de contratación. Determina los deberes y derechos en una negociación de bienes informáticos, derechos de autor y licencias, que permitan realizar la negociación en tecnología, siguiendo los parámetros jurídicos y políticas de la organización.	Formato para desarrollo de evidencia: AP04-AA5-EV04-Elaboración-Términos-Referencia Instrumento de evaluación: IE-AP04-AA5-EV04-Elaboración-Términos-Referencia Documento: plantilla en excel para evaluación de propuestas
EVIDENCIA DE DESEMPEÑO AP04-AA5-EV03. Foro - ¿Cómo diseñar prototipos y modelos de bases de datos?	Elabora el prototipo, representando la funcionalidad del sistema de información, para dar respuesta a las necesidades de la organización, según protocolos de diseño. Diseña la estructura de la base de datos, que cumpla con las necesidades del sistema de información, utilizando herramientas tecnológicas, de acuerdo con el modelo definido y siguiendo normas técnicas.	Formato para desarrollo de evidencia: AP04-AA5-EV03-Foro-Diseño-Prototipo-Base-Datos-SI Instrumento de evaluación: IE-AP04-AA5-EV03-Foro- Diseño-Prototipo-Base-Datos-SI
EVIDENCIA DE PRODUCTO AP04-AA5-EV05. Inglés - Elaboración de un manual técnico.	Puede relacionar textos en inglés con imágenes o con títulos que le sean adecuados. Traduce del inglés documentos técnicos sencillos en tiempo presente y pasado.	Formato para desarrollo de evidencia: AP04-AA5-EV05-Inglés-Elaboración-Manual-Técnico

		Instrumento de evaluación: IE-AP04-AA5-EV05-Inglés-Elaboración-Manual-Técnico
EVIDENCIA DE PRODUCTO AP04-AA5-EV06. Transversal - Ensayo la importancia de la comunicación asertiva.	Se comunica fácilmente con los miembros de la comunidad educativa según protocolos y normas de convivencia institucional. Establece procesos comunicativos asertivos que posibilitan la convivencia en los contextos social y productivo de acuerdo con las competencias ciudadanas. Establece acuerdos mediante el uso de procesos comunicativos, racionales y argumentados orientados hacia la resolución de problemas según normas y protocolos institucionales. Aporta elementos en la construcción colectiva del conocimiento según la naturaleza del problema.	Formato para desarrollo de evidencia: AP04-AA5-EV06.Transversal-Ensayo-comunicación-asertiva Instrumento de evaluación: IE-AP04-AA5-EV06.Transversal-Ensayo-comunicación-asertiva
EVIDENCIA DE PRODUCTO AP04-AA5-EV07. Transversal - Foro - Aplicación de herramientas de comunicación asertiva**.**	Se comunica fácilmente con los miembros de la comunidad educativa según protocolos y normas de convivencia institucional. Establece procesos comunicativos asertivos que posibilitan la convivencia en los contextos social y productivo de acuerdo con las competencias ciudadanas. Establece acuerdos mediante el uso de procesos comunicativos, racionales y argumentados orientados hacia la resolución de problemas según normas y protocolos institucionales. Aporta elementos en la construcción colectiva del conocimiento según la naturaleza del problema.	Formato para desarrollo de evidencia: AP04-AA5-EV07-Aplicar-herramientas-comunicación-asertiva Instrumento de evaluación: IE-AP04-AA5-EV07-Aplicar-herramientas-comunicación-asertiva
EVIDENCIA DE CONOCIMIENTO	Analiza de manera crítica las situaciones pertinentes que contribuyen a la resolución de problemas, según los	Formato para desarrollo de evidencia: AP04-AA5-EV08-Caso-estudio- Distrimay

AP04-AA5-EV08. Transversal. Estudio de Caso DISTRIMAY.	requerimientos de los contextos productivos y sociales. Propone alternativas creativas, lógicas y coherentes que posibiliten la resolución de problemas según la demanda del contexto social y productivo.	Instrumento de evaluación: IE-AP04-AA5-EV08-Caso-estudio-Distrimay
EVIDENCIA DE DESEMPEÑO AP04-AA5-EV09. Transversal. Foro discusión caso DISTRIMAY.	Argumenta y acoge objetivamente los criterios que contribuyen a la Resolución de Problemas según requerimientos del proceso formativo en función de las demandas concretas de los contextos productivos y sociales.	Formato para desarrollo de evidencia: AP04-AA5-EV08-Caso-estudio -Distrimay Instrumento de evaluación: IE-AP04-AA5-EV08-Caso-estudio-Distrimay

5. GLOSARIO DE TÉRMINOS

- **Analizar**: es una acción intelectual de los seres humanos y que implica la realización de un análisis detallado sobre determinado asunto de interés, para conocer sus características, estado, cualidades o extraer conclusiones.
- **Argumentar**: es la expresión oral o escrita de un raciocinio que se emplea para demostrar o probar una proposición o para convencer a una persona sobre algo que se afirma o se niega.
- **Autenticación**: verificación que se realiza a la identidad del usuario.
- **Biography (noun):** It is the story of a person's life written by another person.
- **Campo**: es la representación de un atributo en un modelo relacional el cual contiene su nombre, tipo de dato y descripción de si es clave primaria o llave foránea.
- **Claridad**: exponer ideas concretas y definidas, con frases bien construidas y terminología común, al alcance de los destinatarios. Cuando hay que emplear palabras que puedan generar dudas al receptor, lo mejor es explicar a profundidad para que puedan ser comprendidas.
- **Clave Foránea**: atributo que es pasado a una tabla, según la regla de transformación, se distingue en un modelo relacional porque tiene el estereotipo <fk> y en un esquema relacional porque despliega una flecha hacia la tabla de la cual proviene.
- **Clave Primaria**: atributo único que no se repite, es identificado en un modelo relacional porque está subrayado o tiene el estereotipo <pk> y en un esquema relacional se encuentra subrayado.
- **Coherencia**: construir los mensajes de forma lógica, encadenando ordenadamente las ideas, remarcando lo que son hechos objetivos y lo que son opiniones, bien sea del orador o de otras personas.
- **Coherente**: cosa o persona que tiene relación lógica con otra cosa o entre sus partes. Persona que actúa en consecuencia con las ideas que expresa.
- **Component (noun):** It is one of the parts of something, especially a machine.
- **Comunicación**: la palabra deriva del latín communicare, que significa “compartir algo, poner en común”. Por lo tanto, la comunicación es un fenómeno inherente a la relación que los seres vivos mantienen cuando se encuentran en grupo. A través de la comunicación, las personas o animales obtienen información respecto a su entorno y pueden compartirla con el resto.
- **Comunicación Asertiva**: consiste en saber transmitir de manera efectiva y precisa un mensaje ante

cualquier situación, incluyendo las que representan un reto. Requiere de control emocional y claridad en las ideas.

- **Comunicación No verbal**: se entiende por comunicación no verbal aquellos medios que complementan los mensajes verbales como las expresiones faciales, los gestos corporales y las distancias espaciales que se establecen entre los interlocutores.
- **Comunicación Verbal:** es el uso de las palabras para la interacción entre los seres humanos, el lenguaje propiamente dicho, expresado de manera hablada o escrita.
- **Concisión**: utilizar las palabras justas y huir de la palabrería.
- **Control (noun/ verb):** 1. It referes to the power to make a person, organization, or object do what you want. 2. A switch or piece of equipment that you use to operate a machine or vehicle:
- **DAO**: Data Access Object. Software intermedio que permite el acceso a una base de datos.
- **Design (noun/verb):** It is the way in which something is planned and made.
- **Encapsulamiento**: Propiedad de la programación orientada a objetos en la cual sólo se hacen visibles algunos atributos y métodos de estos últimos.
- **Encripción**: es un proceso para convertir la información a un formato más seguro. En otras palabras, los datos que están en un formato claro, o sea entendible, se convierten mediante un proceso matemático a un formato encriptado o codificado, o sea ininteligible. Una vez que llegan a su destino, se decodifican para poder ser legibles de nuevo, se desencriptan.
- **Esquema Relacional:** expresar el modelo relacional por medio de un esquema de palabras y asociaciones entre ellas (flechas) que representan las relaciones.
- **Estrategia**: es una serie de acciones meditadas o un plan ideado para proyectar o dirigirse hacia un fin determinado, un asunto u operación y definiendo un conjunto de reglas que aseguran una decisión óptima en cada momento.
- **Firma digital**: proporcionan un medio de protección de la autenticidad e integridad de los documentos electrónicos.
- **Herencia**: cualidad de la programación orientada a objetos en la cual estos últimos tienen los atributos y métodos de una clase superior.
- **Heurística**: (hallar, inventar) es el conjunto de métodos o técnicas para resolver problemas. Es el arte de los seres humanos, en procura de buscar estrategias, criterios, métodos que le permitan resolver problemas a través de la creatividad, pensamiento divergente o lateral, también se apoya en la experiencia propia del individuo y en la de los demás, para hallar la solución más viable al problema.
- **Information architecture (noun):** It is the practice of deciding how to arrange the parts of something to be understandable.
- **Input (noun/verb)**: It is information that is put into a computer.
- Interaction (noun): It refers to the way in which two or more things combine and have an effect on each other.
- **Interface (noun):** 1. It refers to the place or area at which different things meet and communicate with or affect each other. 2. A device or program enabling a user to communicate with a computer.
- **Interfaz de usuario**: la interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.
- **ISO27001**: Sistema de Gestión de Seguridad de la Información
- **JDBC:** Java Database Connectivity. Software intermedio que permite el acceso a bases de datos.
- **Layout (noun):** It is a preliminary plan for the arrangement of written material, photographs or other

artwork

- **Modelo Entidad Relación:** modelo que consta de entidades, atributos y relaciones y permite modelar conceptualmente una base de datos relacional.
- **Modelo Entidad Relación Extendido:** modelo que permite modelar nuevos tipos de relaciones, que no incluye el MER normal, tales como agregación, generalización, exclusión e inclusión.
- **MVC**: Model-View-Controller. Tipo de arquitectura que divide la interacción entre el usuario y el sistema en tres capas denominadas Modelo-Vista-Controlador.
- **Naturalidad**: tal vez lo más difícil de lograr porque requiere una expresión viva y espontánea, lo que no implica vulgaridad o descuido. Es la prueba del dominio del lenguaje y el camino para lograr la naturalidad, precisamente por una cuidadosa preparación de la intervención. Vale la pena resaltar que solo con preparación y con ensayo, se puede asegurar que el mensaje llegue a sus destinatarios de forma precisa y fácilmente comprensible.
- **Navigation (noun):** The art or science of directing the course of a ship, aircraft, spacecraft, etc.
- **Objetivo General:** cumple la función de orientador del proyecto o proceso, limitado a los recursos disponibles, susceptible de ser alcanzado, evaluable, expresado de forma afirmativa, clara y precisa.
- **Objetivo Específico:** son resultados cuantificables, medibles mediante indicadores y verificables, de esta manera permiten hacer seguimiento y evaluación del grado de cumplimiento de los efectos que se quieren conseguir con los objetivos específicos.
- **ODBC**: Open DataBase Connectivity : Software intermedio que permite el acceso a bases de datos.
- **Patrón de diseño**: solución probada y documentada a un problema de diseño de software.
- **Plan de Acción**: prioriza las estrategias a seguir, los programas y acciones que se deben llevar a cabo para para alcanzar los objetivos estratégicos y metas propuestas.
- **Polimorfismo**: propiedad de la programación orientada a objetos que permite que un mismo objeto se comporte de manera diferente de acuerdo a los parámetros de entrada.
- **Prototipos**: un prototipo en software es un modelo del comportamiento del sistema que puede ser usado para entenderlo completamente o ciertos aspectos de él y así clarificar los requerimientos. Un prototipo es una representación de un sistema, aunque no es un sistema completo, posee las características del sistema final o parte de ellas.
- **Reflexivo**: considera y evalúa detenidamente las situaciones antes de actuar. Persona meditabundo, pensativo, introspectivo, prudente, sensato.
- **Relación**: es una asociación que existe entre dos entidades, en un MER se distinguen tres tipos de relaciones: uno a uno (1..1), uno a muchos (1..M) y muchos a muchos (M..M).
- **Sencillez**: tanto en la forma de construir el mensaje, como en las palabras empleadas.
- **SGBD**: sistema de gestión de bases de datos.
- **Tabla**: es la representación de una entidad en un modelo relacional, la cual incluye atributos.
- **Trabajo Independiente:** corresponde al tiempo que el aprendiz debe dedicar en su proceso de formación de manera independiente al tiempo que normalmente recibe con acompañamiento directo del instructor.
- **Trabajo con acompañamiento directo:** corresponde al tiempo directo que el instructor debe dedicar al aprendiz en el proceso formativo, este acompañamiento se enfoca en el seguimiento, la realimentación, orientación y evaluación.
- **Transformación**: referente a la acción o procedimientos mediante el cual algo se altera o modifica o cambia de forma.
- **User (noun)**: A user is someone who uses a product, machine, or service.

6. REFERENTES BIBLIOGRÁFICOS

Bennet, Simon. McRobb, Steve. Farmer, Ray (2006). Análisis y diseño orientado a objetos de sistemas usando UML. McGrawHill, Madrid.

Brönstrup, S., Celsi, E. y Ribeiro, A. (2009). Comunicación, lenguaje y comunicación organizacional. Bogotá, Colombia: Red signo y pensamiento.

Cambridge Dictionary | Free English Dictionary, T. (2017). Cambridge Dictionary | Free English Dictionary, Translations and Thesaurus. Dictionary.cambridge.org. Recuperado de http://dictionary.cambridge.org/us/

Castro, A. (2014). Comunicación organizacional: técnicas y estrategias. Bogotá, Colombia: Autor.

CHANG, RICHARD; KELLY, KEITH, Resolución de Problemas, Argentina, Granica, 1994.

CONSTANTINO, CATHY A.; SICKLES MERCHANT, CHRISTINA, Diseño de sistemas para enfrentar conflictos. Una guía para crear organizaciones productivas y sanas, Barcelona, Granica, 1997.

Costas, J. (2014). Seguridad Informática. RA-MA Editorial. Recuperado de http://site.ebrary.com.bdigital.sena.edu.co/lib/senavirtualsp/reader.action?docID=11038505

Díaz, S. (2014). Una mirada a la racialidad desde la comunicación organizacional. La Habana, Cuba: Editorial universitaria.

Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley. USA.

Gramma E., Helm R., Johnson R., Vlissides John, (1994). Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley. USA

Henry F. Korth , (2006). Fundamentos de Bases de Datos. McGraw-Hill.

How to Write User Manuals. (2017). wikiHow. Recuperado de http://www.wikihow.com/Write-User-Manuals

Kendall, K. E., & Kendall, J. E. (2005). Análisis y Diseño de Sistemas (Sexta ed.). México: Pearson Educación.

Marco Lógico de la Investigación Científica. Dr. Normand Eduardo Aruad Senén. Mtra. Cristina Vásquez Ruiz. Agosto (2014)

Mercovich, E. (s.f.). Diseño Iterativo de Interfaces. Recuperado el 8 de Julio de 2013, de GAIASUR: http://www.gaiasur.com.ar/infoteca/siggraph99/diseno-iterativo-y-prototipado-rapido.html

Pacheco, R. (2006). Competencias claves para la comunicación organizacional. Quito, Ecuador: ProQuest ebrary.

Pressman, R. (2010). Ingeniería del software, un enfoque práctico (Séptima edición). Bogotá: McGraw-Hill.

Sayago, L. (2009). Investigación en comunicación organizacional. Córdoba, Argentina: El cid editor.

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Rita Rubiela Rincón Badillo	Líder Expertos Temáticos	Centro Industrial de Mantenimiento Integral- CIMI - Regional Santander.	Junio de 2017
	Edgar Eduardo Vega	Experto Temático		
	Nelson Mauricio Silva	Experto Temático		
	Edward Beltrán Lozano	Experto Temático		
	César Hernández Flórez	Experto Competencia Transversal - Emprendimiento	Centro de Servicios Empresariales y Turísticos - Regional Santander.	Junio de 2017
	Laura Salomé Ortiz Moreno	Experta Competencia Transversal - Ética y comunicación		
	Ingrid Carolina Flórez Urzola	Competencia Transversal Inglés	Centro de Gestión de Mercados, Logística y Tecnología de la Información - Regional Distrito Capital.	Junio de 2017
	Rosa Elvia Quintero Guasca	Asesor Pedagógico	Centro Industrial de Mantenimiento Integral- CIMI. Regional Santander.	Junio de 2017
	Claudia Milena Hernández Naranjo	Asesor Pedagógico		

8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	**Cargo**	**Dependencia**	**Fecha**	**Razón del Cambio**
Autor (es)					No aplica- guías generadas a partir de un formato nuevo en 2017.



FAVA - Formación en Ambientes Virtuales de Aprendizaje

SENA - Servicio Nacional de Aprendizaje.

Estructura de contenidos

Pág.

Introducción 3
Mapa de contenido 4
Desarrollo de contenidos 5
1. Principios de diseño de las interfaces de usuario 5
2. Tipos de interfaz de usuario 5
2.1. Interfaces de lenguaje natural 5
2.2. Interfaces de pregunta y respuesta 5
2.3. Interfaces con menús 6
2.4. Interfaces con formularios (entrada/salida) 6
2.5. Interfaces de lenguaje de comandos 6
2.6. Interfaces gráficas del usuario (GUI) 6
3. El proceso de diseño de la interfaz de usuario 7
3.1. Análisis del usuario 7
3.2. Prototipo de la interfaz de usuario 8
3.3. Evaluación de la interfaz 8
4. Herramientas software para el diseño de interfaces 10
4.1. Adobe dreamweaver 10
4.3. Wirefreams on the fly mockingbird 12
5. Ejemplos diseño interfaces casos de uso 13
Glosario 17
Bibliografía 18
Control del documento 19

INTERFAZ GRÁFICA DE USUARIO

Introducción

En el proceso general del diseño y desarrollo de software hay un elemento muy importante que es el diseño de la interfaz de usuario. Las grandes empresas contratan personas expertas para ello, pero en la mayoría a los desarrolladores les toca asumir la responsabilidad de diseñar la interfaz de usuario, así como del diseño de software que implementa la interfaz. Un diseño cuidadoso de la interfaz de usuario es parte fundamental del proceso de diseño general del software. Si un sistema software debe alcanzar su potencial máximo, es fundamental que su interfaz de usuario sea diseñada para ajustarse a las habilidades, experiencias y expectativas de sus usuarios previstos.

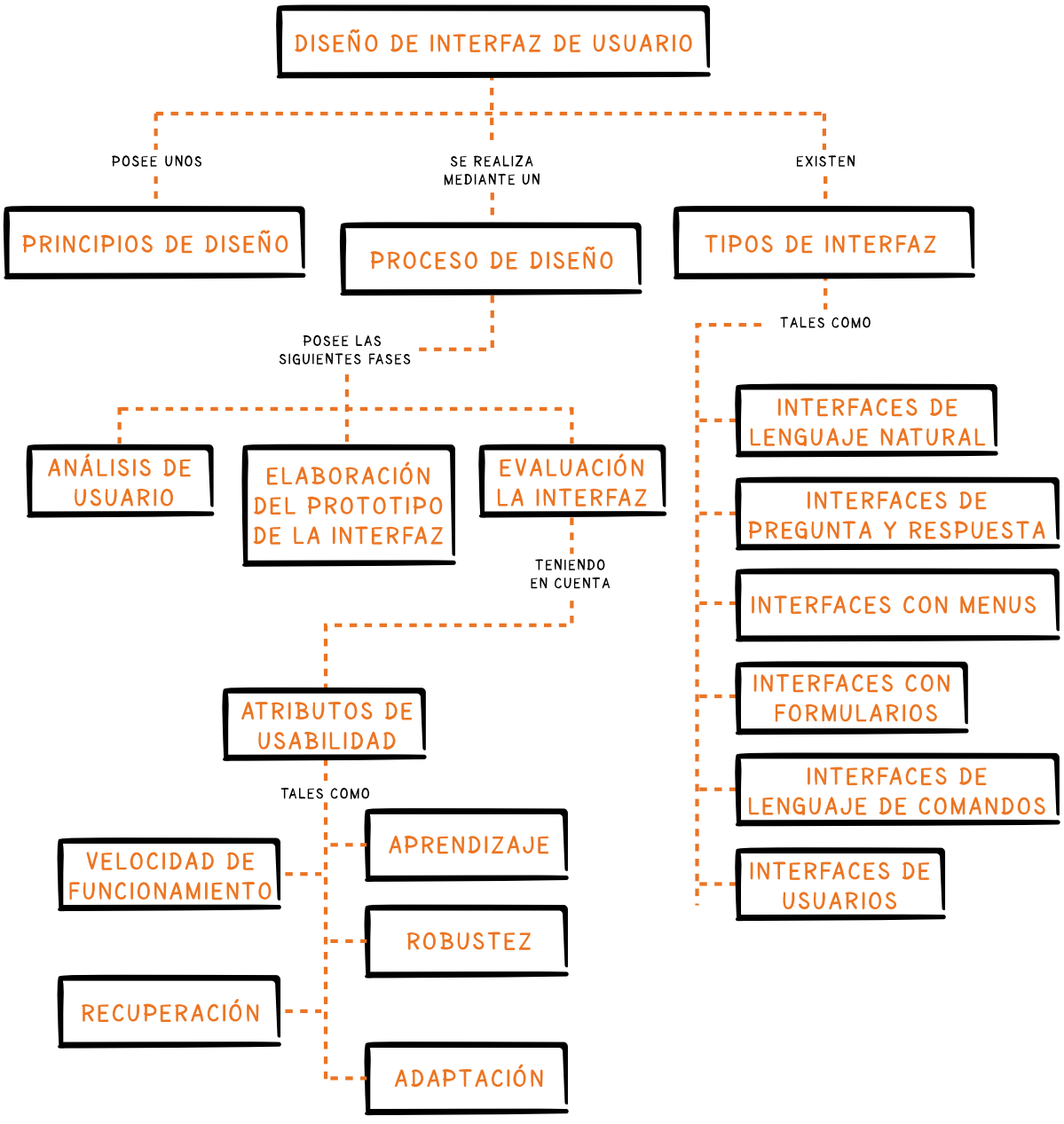


Antes de iniciar el proceso de diseño de la interfaz de usuario, es bueno hacernos las siguientes preguntas:

- ¿Cómo debe interactuar el usuario con el sistema informático?
- ¿Cómo debe presentar la información del sistema informático al usuario?

Una muy buena interfaz de usuario debe integrar los dos elementos relacionados en las preguntas, la interacción del usuario y la presentación de la información.

Mapa de contenido



Desarrollo de contenidos

1. Principios de diseño de las interfaces de usuario

Principio	Descripción
Familiaridad del usuario	La interfaz debe utilizar términos y conceptos obtenidos de la experiencia de las personas que más utilizan el sistema.
Uniformidad	Siempre que sea posible, la interfaz debe ser uniforme en el sentido que las operaciones comparables se activen de la misma forma.
Mínima Sorpresa	El comportamiento del sistema no debe provocar sorpresa a los usuarios.
Recuperabilidad	La interfaz debe incluir mecanismos para permitir a los usuarios recuperarse de los errores.
Guía de Usuario	Cuando ocurran errores, la interfaz debe proporcionar retroalimentación significativa y características de ayuda sensible al contexto.
Diversidad de Usuarios	La interfaz debe proporcionar características de interacción apropiadas para los diferentes tipos de usuarios del sistema.

Figura 1. Principios de diseño de las interfaces de usuario.
Fuente: SENA.

2. Tipos de Interfaz de usuario

2.1. Interfaces de lenguaje natural



Estas interfaces son las “ideales” y utópicas a la vez, ya que en teoría debe permitir la interacción del usuario en su lenguaje natural, como el español o inglés, sin requerir de ninguna otra habilidad especial. El problema es que el lenguaje natural es ambiguo y las interfaces son limitadas.



2.2. Interfaces de pregunta y respuesta

En esta interfaz se le presenta una pregunta al usuario, da su respuesta y puede continuar con la siguiente pregunta, normalmente dependiendo de la respuesta anterior.

2.3. Interfaces con menús

En esta interfaz se le presentan una serie de opciones al usuario y dependiendo de su elección se ejecuta una acción. Los menús pueden estar anidados y llevar de una opción a un nuevo menú.



El usuario necesita saber qué tarea debe ser realizada. En este tipo de interfaz es importante ser consistente, respecto al lugar en que se colocan los menús, teclas de rápido acceso, nombres comunes para ciertas tareas, etc.

2.4. Interfaces con formularios (entrada/salida)



Estas interfaces tratan de asimilar las formas impresas que normalmente utiliza el usuario, ayudándole mostrando dónde colocar qué información. Algunas formas facilitan el trabajo llenando valores por omisión.

En estas interfaces habrá que considerar si el usuario es experimentado, en cuyo caso esperará que las interfaces sean más eficientes que las formas impresas tradicionales.

2.5. Interfaces de lenguaje de comandos



En esta interfaz se permite al usuario controlar la aplicación por medio de comandos, conjuntos de teclas, frases, o combinación de las anteriores. Por lo anterior, el usuario tiene mayor control y poder de decisión sobre la aplicación, pero requieren capacitación para que el usuario memorice los comandos y reglas de utilización.

2.6. Interfaces gráficas del usuario (GUI)

Estas son las interfaces más sofisticadas y complicadas, que ofrecen mayor funcionalidad y flexibilidad al usuario.



3. El proceso de diseño de la interfaz de usuario

El diseño de interfaces es un proceso interactivo, que en general, se puede resumir en las siguientes fases:

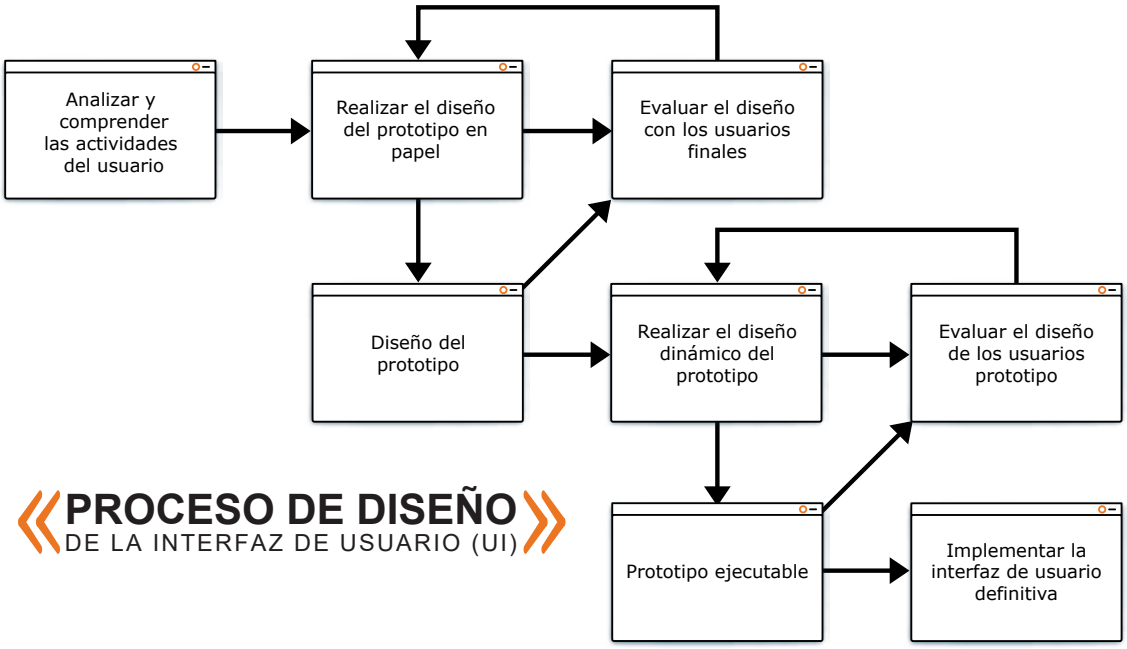


Figura 2. Poceso de diseño de la interfaz de usuario (UI).
Fuente: SENA.

3.1. Análisis del usuario

Se realiza un análisis cuidadoso para conocer y clasificar al usuario, su tarea y el ambiente.

Sino se entiende lo que los usuarios quieren hacer con el sistema, no se podrá llevar a cabo un diseño real y eficaz de la interfaz de usuario. Para desarrollar esta comprensión, se pueden utilizar técnicas como el análisis de tareas, estudios etnográficos, entrevistas de usuarios y observaciones.

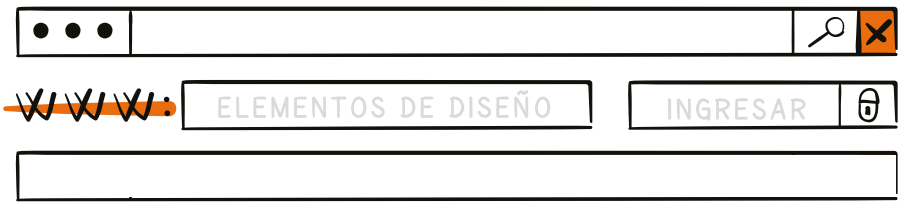


Figura 3. Representación análisis del usuario.
Fuente: SENA.

3.2. Prototipo de la interfaz de usuario

En esta fase se elaboran formatos de pantalla donde se incluyan menús, ubicación de íconos, texto descriptivo, nombres de ventanas, etc. Pueden generarse prototipos de preferencia con herramientas constructoras de GUIs (graphical user interfaces). También en esta fase se realiza la implementación.

Para el diseño de la interfaz se recomienda realizar prototipos evolutivos o exploratorios con la implicación de los usuarios finales. Implicar al usuario en el proceso de diseño y desarrollo es un aspecto fundamental del diseño centrado en el usuario (Nrmal y Draper, 1986), un criterio de diseño para sistemas operativos.

El propósito de realizar prototipos es permitir a los usuarios adquirir una experiencia directa con la interfaz.

Cuando se están construyendo los prototipos se debe realizar en dos etapas así:

- Iniciar con prototipos en papel, maquetas de los diseños de las pantallas y mostrarlos a los usuarios finales.
- Después se perfecciona el diseño y se desarrollan los prototipos automatizados y se ponen a disposición de los usuarios para realizar pruebas y simulación de actividades.

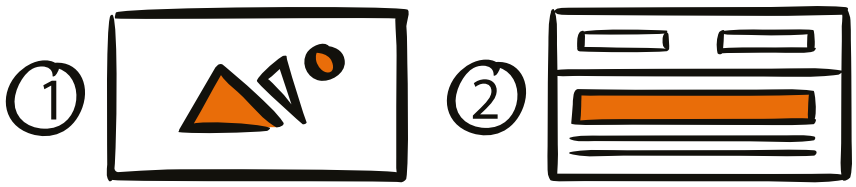


Figura 4. Ejemplo de creación de prototipos.
Fuente: SENA.

3.3. Evaluación de la interfaz

Se realizan pruebas y validaciones de las interfaces previas a la liberación.

La evaluación de la interfaz es el proceso de evaluar la forma en que se utiliza una interfaz y verificar que cumple los requerimientos del usuario. La evaluación se debe llevar a cabo contra una especificación de la usabilidad basada en atributos de la usabilidad.

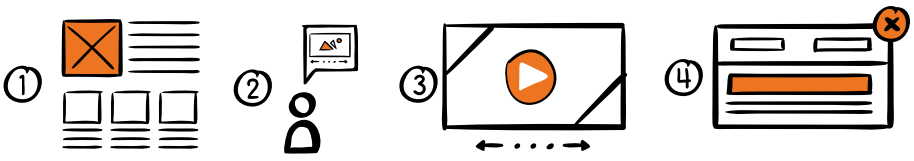
Cabe resaltar que el usuario debe estar involucrado en el proceso de diseño para facilitar que acepte el producto final.

Atributo	Descripción
Facilidad de Aprendizaje	Rapidez con que un usuario aprende a utilizar un sistema con el que no ha tenido contacto anteriormente.
Eficiencia	Que el usuario pueda alcanzar un alto nivel de productividad al usar el sistema.
Retención en el tiempo	Que el usuario pueda recordar fácilmente como se utiliza un sistema cuando ha dejado de utilizarlo por un tiempo.
Tasa de error	Se refiere a la cantidad y gravedad de errores que comete el usuario. Al cometer un fallo el sistema debe informar a los usuarios y ayudar a solucionarlos.
Satisfacción	Hace referencia, si los usuarios se sienten cómodos y satisfechos al utilizar el sistema.
Control	Los usuarios deben tener el control sobre el sistema, y no a la inversa.
Habilidades	El sistema debe respetar al usuario, es decir, el usuario debe sentir que su experiencia se ve potenciada y complementada por el sistema.
Privacidad	El sistema debe ayudar a los usuarios a proteger su información.
Flexibilidad	Posibilidades de intercambio de información entre los usuarios y el sistema.
Robustez	Nivel de apoyo del Sistema al Usuario.
Velocidad de Utilización:	Tiempo que tarda un usuario en realizar una actividad, una vez ya conoce el sistema.

Figura 5. Atributos para evaluación de la interfaz.
Fuente: SENA.

Existen algunas técnicas sencillas para la evaluación de interfaces que pueden identificar deficiencias específicas en el diseño tales como:

- Cuestionarios que recopilan información de lo que opinan los usuarios de la interfaz.
- La observación de los usuarios cuando trabajan con el sistema y piensan en voz alta, de como tratan de utilizar el sistema para llevar a cabo alguna tarea.
- Instantáneas de vídeos del uso típico del sistema.
- La inclusión de código en el software que recopila información de los recursos más utilizados y de los errores más comunes.



4. Herramientas software para el diseño de interfaces

En la actualidad existen muchas herramientas software para el diseño de interfaces. Dentro de ellas encontramos herramientas web y herramientas de aplicaciones Windows. Las herramientas de desarrollo de software como visual studio y netbeans, incluyen elementos propios para el diseño de interfaces.

A continuación se relacionan tres (3) herramientas posibles a utilizar:

4.1. Adobe Dreamweaver:

Esta herramienta es licenciada y viene incluida en la suite de adobe para el diseño de sitios web.

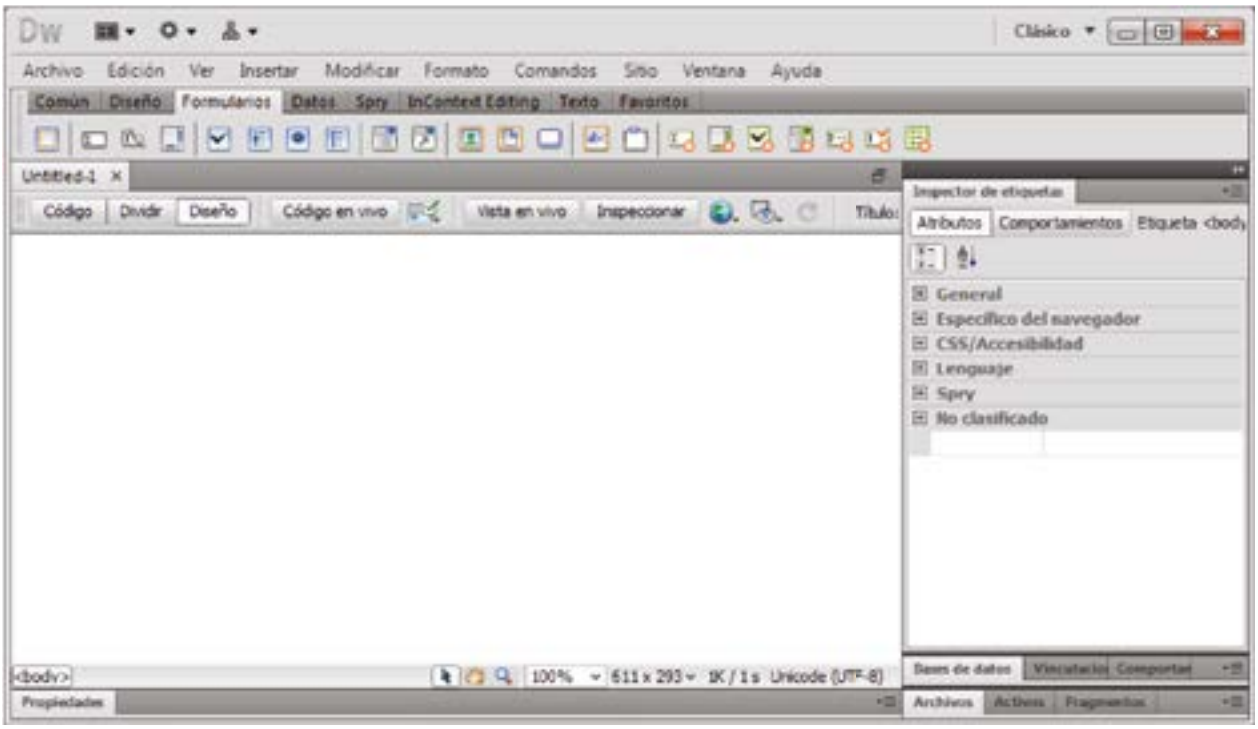


Figura 6. Captura de pantalla del software Adobe Dreamweaver.
Fuente: SENA.

4.2. Balsamiq Mockups:

Es una herramienta web que se puede descargar desde internet por un periodo de prueba.

Sitio de descarga: http://www.balsamiq.com

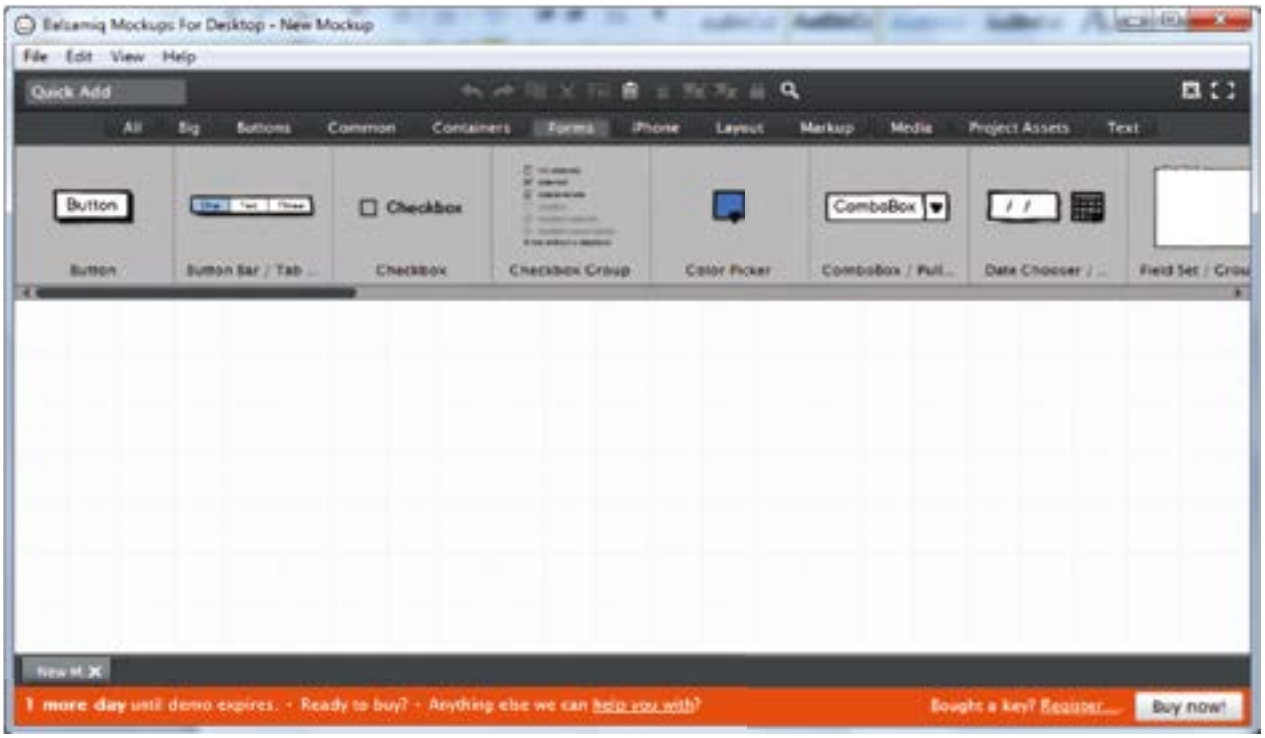


Figura 7. Captura de pantalla de la herramienta Balsamiq Mockups.
Fuente: SENA.

La herramienta permite descargar las interfaces a diferentes formatos como PNG o PDF.

4.3. Wirefreams on the fly mockingbird:

Herramienta web para el diseño de interfaces, se pueden crear las interfaces directamente desde el navegador.

Se debe ingresar a la siguiente dirección: https://gomockingbird.com/



Figura 8. Captura de pantalla de la herramienta Wirefreams onthe fly mockinbird.
Fuente: SENA.

5. Ejemplos diseño interfaces casos de uso

Se tiene como referencia un sistema de un centro odontológico.

Para iniciar se debe realizar la interfaz de usuario para el siguiente caso de uso:

Caso de uso: autenticar usuario.
Sistema: gestión odontológica.

Caso de Uso	**Autenticar usuario**	
Descripción	El comportamiento del Sistema debe validar el usuario que ingresa al sistema.	
Precondición	El personal debe estar registrado en el sistema para que al ingresar sus datos de login y password puedan ser validados.	
Secuencia Normal	Paso	Acción
	1	El personal médico al ingresar al sistema debe ingresar con un login y password asignados por el administrador.
	2	El sistema valida el login y password.
	3	Si los datos son válidos, el sistema muestra las opciones correspondientes, dependiendo del tipo de usuario.
Pos condición	El empleado ingresa al sistema.	
Excepciones	Paso	Acción
	3	Si los datos de login y password no son válidos, el sistema informa al personal médico para que se revise y actualice la información.
	3	Si el login y password no son válidos el sistema no permite ingresar.

Figura 9. Descripción de ejemplo de caso de uso autenticar usuario.
Fuente: SENA.

Pasos para realizar la interfaz mediante prototipos:

1. Crear un prototipo en una hoja de papel.



Figura 10. Crear un prototipo en una hoja de papel.
Fuente: SENA.

2. Luego se procede a realizar utilizando una herramienta software. Para ello se utiliza Balsamiq Mockups. En el documento de apoyo llamado, crear interfaces Balsamiq se muestra el paso a paso de la creación de la siguiente interfaz.



Figura 11. Crear un prototipo en Balsamiq Mockups.
Fuente: SENA.

Caso de uso: registrar paciente.
Sistema: gestión odontológica.

Caso de Uso	**Autenticar usuario**	
Descripción	El comportamiento del sistema deberá describir el paso a paso del caso de uso cuando el personal encargado de gestionar datos del paciente inicie el ingreso de estos.	
Precondición	El paciente no se encuentra registrado en el sistema y tiene la documentación necesaria para poder ser ingresado al sistema.	
Secuencia Normal	Paso	Acción
	1	El sistema muestra al personal médico una interfaz para que se ingresen los datos del nuevo paciente (identificación, nombres, apellidos, dirección, teléfono, correo, genero, tipo de sangre, nombre acudiente).
	2	El personal médico ingresa los datos del paciente.
	3	El sistema valida inicialmente si el documento de identidad del nuevo paciente no existe en el sistema, si no existe registra el nuevo paciente.
	4	El sistema informa al personal médico que el paciente ya ha sido ingresado al sistema.
Pos condición	El paciente queda registrado en el sistema.	
Excepciones	Paso	Acción
	3	Si el sistema detecta que ya existe un paciente registrado con ese documento de identidad, procede a informar al personal médico, estos deben modificar y/o actualizar si es necesario.
	3	Si ya existe la persona en el sistema se procede a cancelar el registro del paciente y se termina el caso de uso.

Figura 12. Descripción de ejemplo de caso de uso registrar paciente.
Fuente: SENA.

Pasos para realizar la interfaz mediante prototipos:

1. Crear un prototipo en una hoja de papel.



Figura 13. Crear un prototipo en una hoja de papel.
Fuente: SENA.

2. Luego se crea la interfaz utilizando una herramienta software desde gomockingbird. com. En el documento de apoyo llamado, crear Interfaces Gomockingbird se muestra el paso a paso de la creación de la siguiente Interfaz.

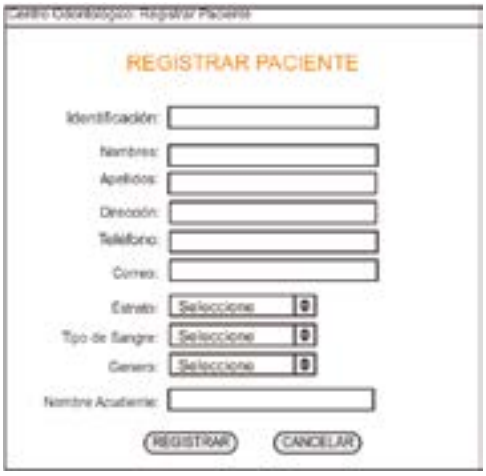


Figura 14. Crear un prototipo en Gomockingbird.
Fuente: SENA.

Glosario

Formularios: un formulario es un documento, diseñado con el propósito de que el usuario introduzca datos estructurados en las zonas del documento destinadas a ese propósito para ser almacenadas y procesadas posteriormente.

GUI: la interfaz gráfica de usuario, conocida también como GUI (del inglés graphical user interface) es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

Interfaz de usuario: la interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.

Menús: es una serie de opciones que el usuario puede elegir para realizar determinada tarea.

Prototipos: un prototipo en software es un modelo del comportamiento del sistema que puede ser usado para entenderlo completamente o ciertos aspectos de él y así clarificar los requerimientos. Un prototipo es una representación de un sistema, aunque no es un sistema completo, posee las características del sistema final o parte de ellas.

Usabilidad: se refiere a la facilidad con que las personas pueden utilizar una herramienta particular o cualquier otro objeto fabricado por humanos con el fin de alcanzar un objetivo concreto. La usabilidad también puede referirse al estudio de los principios que hay tras la eficacia percibida de un objeto.

Bibliografía

Kendall, K., Kendall, J. (2005). Análisis y Diseño de Sistemas. México: Pearson Educación.

Mercovich, E. (2003). Diseño Iterativo de Interfaces. Buenos Aires.

Sommerville, I. (2005). Ingeniería del Software. Madrid: Pearson Educación.

Control del documento

CONSTRUCCIÓN
OBJETO DE
APRENDIZAJE

INTERFAZ GRÁFICA DE USUARIO	
Centro Industrial de Mantenimiento Integral - CIMI Regional Santander	
Líder línea de producción:	Santiago Lozada Garcés
Asesores pedagógicos:	Rosa Elvia Quintero Guasca Claudia Milena Hernández
Líder expertos temáticos: **Experto temático:** **Experto temático:**	Rita Rubiela Rincón Badillo César Marino Cuellar Chacón - V1 Rita Rubiela Rincón Badillo - V2
Diseño multimedia:	Jesús Antonio Vecino Valero
Programador:	Francisco José Lizcano Reyes
Producción de audio:	Víctor Hugo Tabares Carreño

Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de la licencia que el trabajo original.

creative commons
BY NC SA

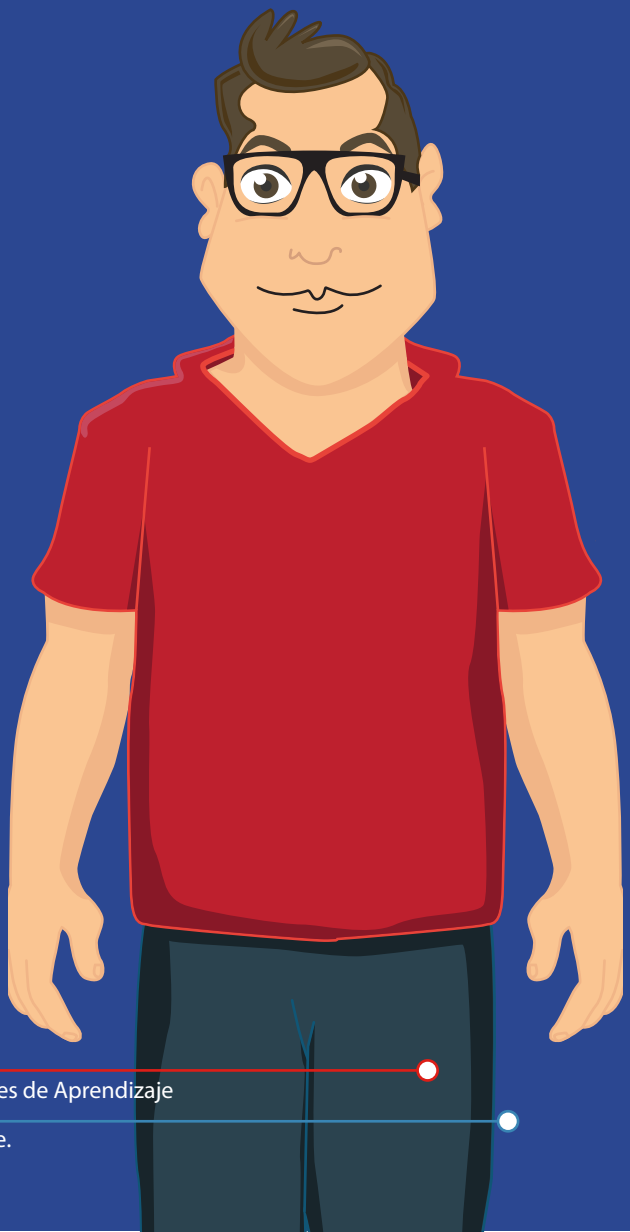
DreamWeaver® - Adobe® Inc. 2017. Todos los derechos Reservados

Balsamiq® Studios LLC. 2017. Todos los derechos reservados.

© Mockingbird By Some Character LLC. 2017. Todos los derechos reservados.

Registered trademark

Patrones de Diseño



FAVA - Formación en Ambientes Virtuales de Aprendizaje

SENA - Servicio Nacional de Aprendizaje.

ESTRUCTURA DE CONTENIDOS

Pág.

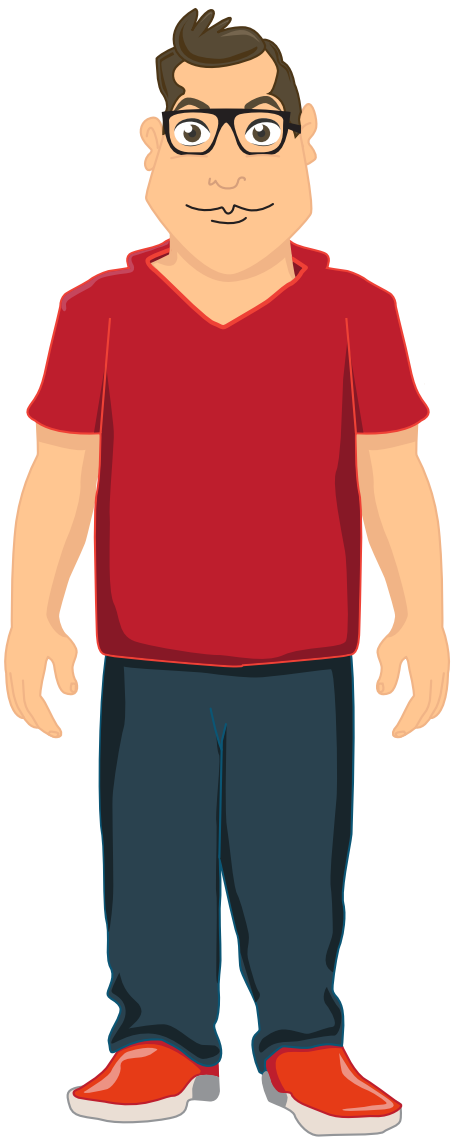
Introducción 3
Mapa de contenido 4
1. Generalidades. 5
2. Patrones GoF: Gang-of-Four. 6
2.1. Patrones creacionales. 6
2.2. Patrones estructurales 12
2.3. Patrones de comportamiento. 18
3. Patrón modelo vista controlador (MVC). 19
4. Patrones de arquitectura para aplicaciones empresariales. 22
5. Cómo elegir un patrón de diseño. 23
Glosario 24
Bibliografía 25
Control del documento 26

PATRONES DE DISEÑO

INTRODUCCIÓN

Durante la etapa de diseño de un sistema de información una de las principales tareas es la construcción de un prototipo con las clases que este sistema utilizará. En esta etapa existen problemas comunes al desarrollo de software que algunos autores han tipificados y generado soluciones que ha sido de gran aceptación. Las soluciones tipificadas, probadas y documentadas a estos problemas comunes se denominan patrones de diseño.

Los patrones de diseño son una herramienta que soporta la actividad de diseño de la arquitectura, proporcionando en algunas ocasiones el punto de inicio en la determinación de aspectos relacionados con la especificación de clases y su interacción en un sistema de información dado.

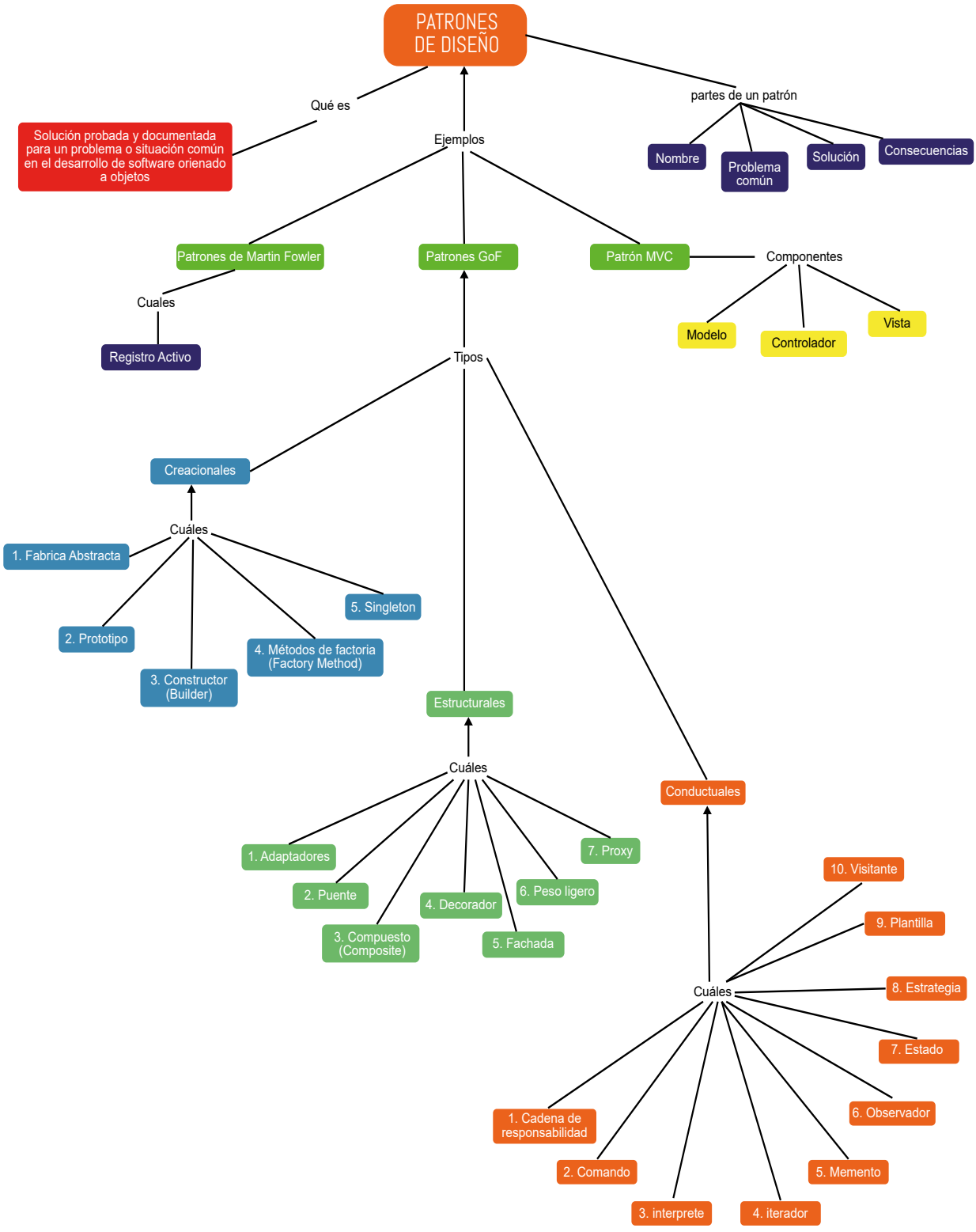


Es importante anotar que los patrones son influenciados por el paradigma de programación orientada a objetos. Desde este punto de vista los patrones o soluciones harán uso intensivo de las características propias de este paradigma como son: herencia, polimorfismo, encapsulamiento, entre otros.

Para un mejor entendimiento del recurso el aprendiz debe tener claro los siguientes conceptos de programación orientada a objetos vistos en el programa: clases, objetos, atributos, métodos, clase abstracta, interfaz, dependencia, generalización, especialización, instanciación,diagramas UML, entre otros.

La temática que se presenta en este objeto proporciona información sobre algunos de los patrones de diseño más importantes, con el fin de que sean considerados durante la fase de diseño del sistema de información que desarrollará el aprendiz.

MAPA DE CONTENIDO



DESARROLLO DE CONTENIDOS

1. Generalidades.

Un patrón de diseño es un modelo o plantilla de la solución a un problema común en el desarrollo de software. La historia de estos patrones se remonta a 1987 cuando Warde Cunningham y Kent Beck profesionales del desarrollo de software se dieron a la tarea de retomar algunas de las que consideraban “buenas ideas” en la solución de problemas orientados a objetos y desarrollaron cinco patrones de interacción publicados en un artículo académico (Fowler, 2002).

Sin embargo no fue hasta 1990 cuando el grupo denominado “Gang of Four” (La banda de los cuatro) que en adelante se identificarían como GoF retomaron un conjunto de 23 patrones de diseño y se empezó con mayor fuerza a difundir su uso.

Los patrones de diseño se enfocan en brindar una solución a problemas de software que han sido probados y tienen la documentación que permite analizar su posibilidad de uso. Esta documentación cuenta entre otros con (Gramma,1994):

- **Nombre:** describe el problema, solución y consecuencias a través de un nombre significativo.
- **Problema:** describe cuando debe utilizarse el patrón así como las condiciones para su aplicación.
- **Solución:** describe los elementos de diseño a incorporar, con la disposición general de las clases y objetos de forma abstracta.
- **Consecuencias:** los costos/beneficios al aplicar el diseño.

El principal objetivo que se consigue al utilizar esta técnica es el de la reutilización, lo que facilita conseguir apuntar a los principios de calidad en el diseño de software que son: la eficiencia, la mantenibilidad del sistema, que el sistema sea correcto y por ende la disminución en costos.

Una vez que se ha seleccionado el patrón de diseño a utilizar se recomienda leer inicialmente las partes relacionadas con la aplicabilidad y consecuencias para verificar que es el correcto, revisar concienzudamente la estructura para entender las clases y relaciones, así como verificar el código de ejemplo, que también puede ayudar a entender la implementación del mismo, escoger nombres significativos para los objetos y elementos, declarar las clases e identificar las clases que afectará, definir nombres para los métodos de acuerdo con la guía y por último redactar las operaciones.

2. Patrones GoF: Gang-of-Four.

Se pueden clasificar en 3 categorías (GRAMMA,1994):

- Patrones Creacionales (Creational patterns).
- Patrones Estructurales (Structural patterns).
- Patrones de Comportamiento (Bahavioral patterns).

2.1. Patrones creacionales.

Los patrones de creación muestran una guía de cómo crear objetos cuando su creación requiere tomar decisiones como qué clases instanciar o sobre que objetos se delegará responsabilidades.

Tienen relación con la instanciación de Clases. Pueden ser divididos en patrones de creación de clases (class-creation patterns) y patrones de creación de objetos (object-creational patterns).

- Los primeros usan la herencia dentro del proceso de instanciación.
- Los otros usan la delegación para realizar la tarea.

2.1.1. Patrón fabrica abstracta (abstract factory).

Problema:

¿Cómo crear familias de clases relacionadas que implementan una interfaz común? ¿Cómo hacer que el cambio de la familia de clases a utilizar sea transparente para la aplicación?

Solución:

Definir una interfaz que contenga métodos para la creación de los objetos de cada clase.

- Esta representa la Fábrica abstracta.
- Puede ser una interfaz o una clase abstracta.
- Definir las fábricas concretas para cada familia de clases.

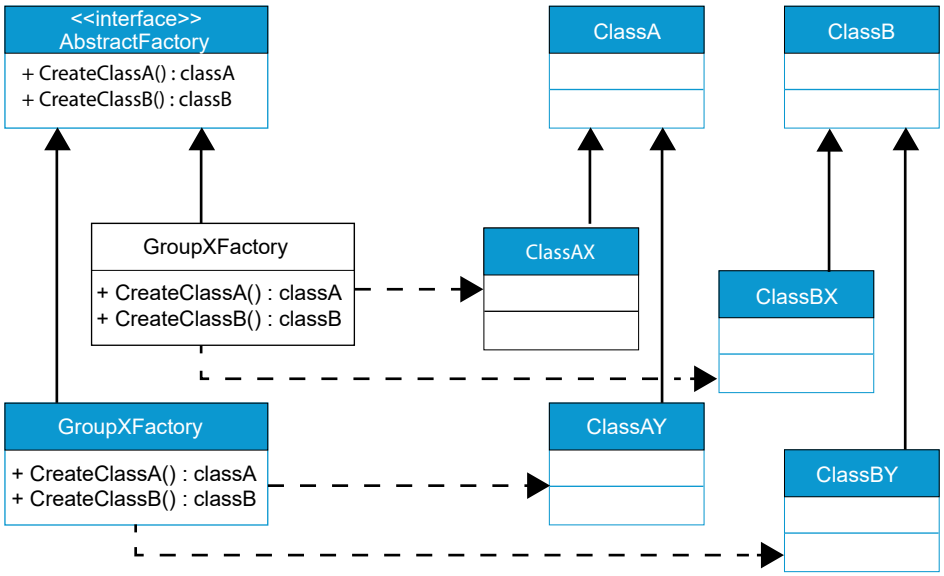


Figura 2.1. Patrón Fábrica Anstracta (Abstract Factory)

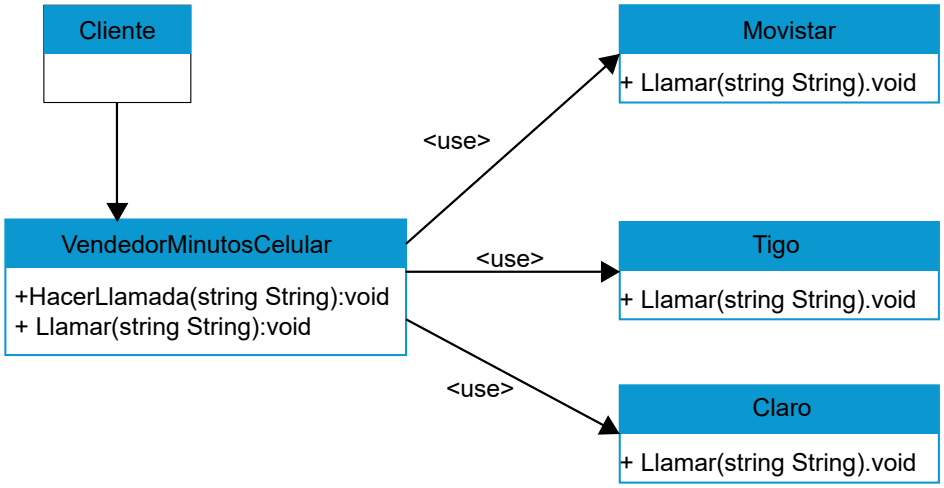


Figura 2.2. Ejemplo de Fábrica Abstracta (Abstract Factory)

2.1.2. Patrón Constructor (Builder).

Problema:

¿Cómo construir objetos complejos manteniendo la alta cohesión?
¿Cómo construir objetos que requieren varios pasos en su construcción?

Solución:

- Se crea una interfaz que defina cada uno de los pasos de la construcción.
- Se define una clase que dirija la construcción de los objetos.
- Para cada forma distinta de crear el objeto, se definine una clase concreta que la defina.

- Muy similar a Fábrica Abstracta.

Tener en cuenta que:

1. Fábrica Abstracta se aplica a familia de objetos.
2. Constructor se aplica a la creación objetos complejos

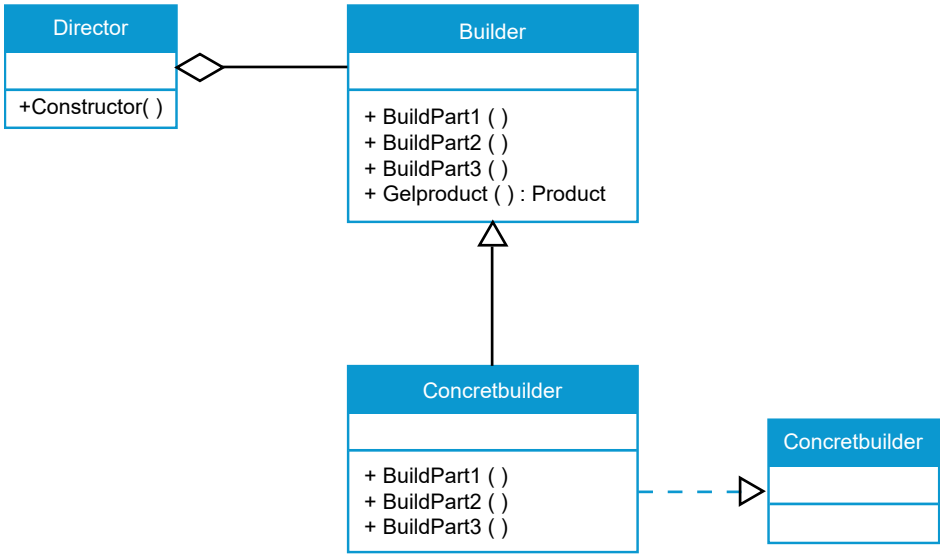


Figura 2.3. Patrón Constructor (Builder)

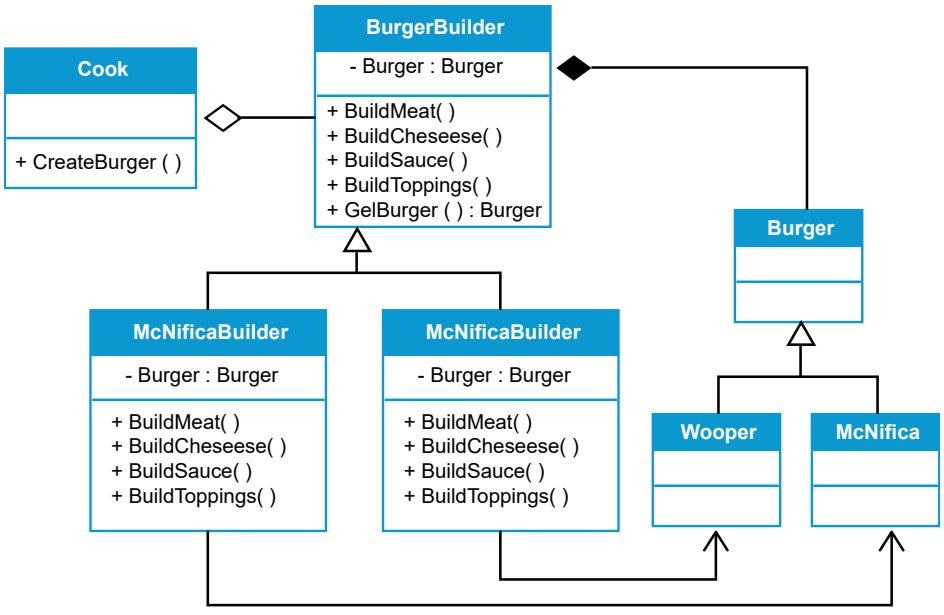


Figura 2.4. Ejemplo Patrón Constructor (Builder)

2.1.3. Patrón método factoría (factory method).

Problema:

¿Cómo crear instancias de elementos sin saber el tipo concreto del elemento?
¿Cómo crear instancias de una clase, donde la creación es un proceso complejo?

Solución:

- Se asigna la responsabilidad de la creación a un método particular.
- En caso de que sean múltiples tipos, es el método el que se encarga de la decisión de cuál elemento crear.
- En caso de creación compleja, el método encierra y se encarga de esta complejidad.

Tener en cuenta que:

1. Dado la responsabilidad de los constructores, estos no debieran dedicarse a lidiar con operaciones complejas.
2. Si existiesen operaciones muy complejas (lectura de archivos, acceso a bases de datos), en ese caso es mejor que un método se encargue de la responsabilidad de crear a los objetos de la clase.
3. Es una simplificación de Fábrica Abstracta, donde solo se construye instancias de una clase.

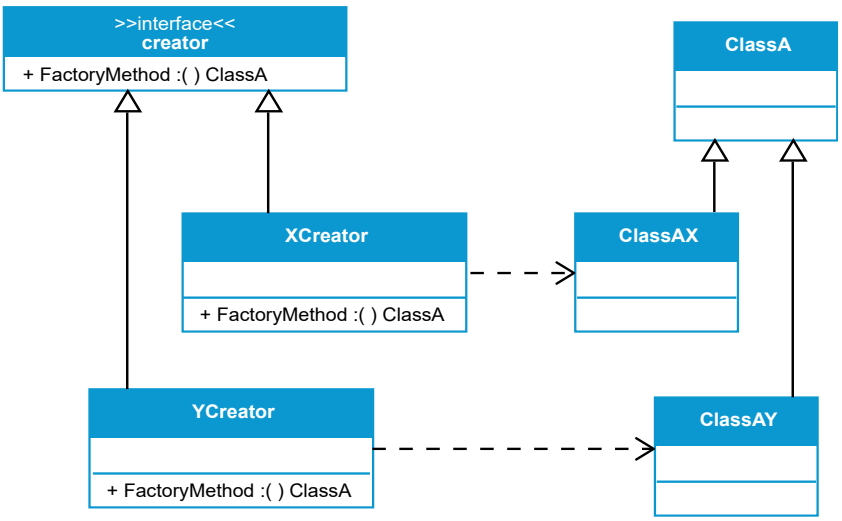


Figura 2.5. Patrón Método Factoría (Factory Method)

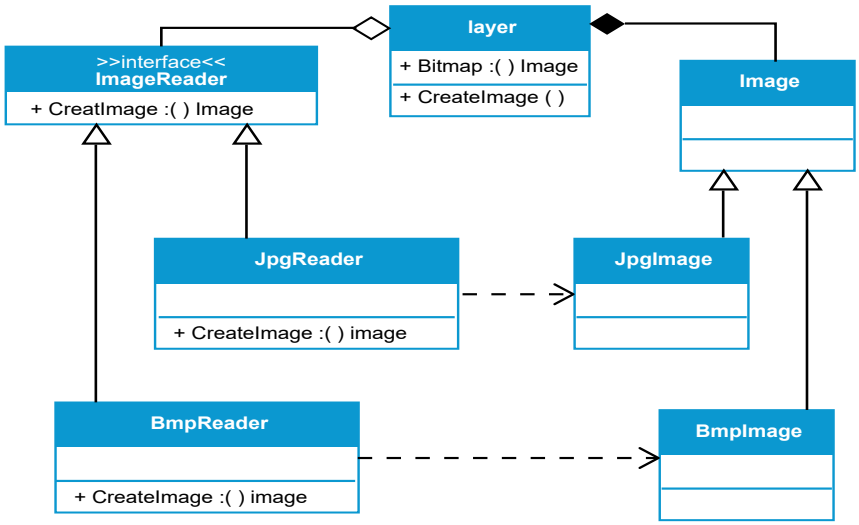


Figura 2.6. Ejemplo Patrón Método Factoría

2.1.4. Patrón prototipos (prototype).

Problema:

¿Cómo aumentar el rendimiento en la construcción de objetos que siempre son iguales?

Solución:

- Se crea una instancia del objeto y luego se clona.
- Para ello, se define una interfaz que debe implementar todos los objetos a ser clonados.
- En la mayoría de los lenguajes es posible hacer un Shadow copy de los objetos.
  - En C# existe el método MemberwiseClone().
  - En Java existe clone().
- Este se puede combinar con Método Factoría para la construcción de distintos objetos.

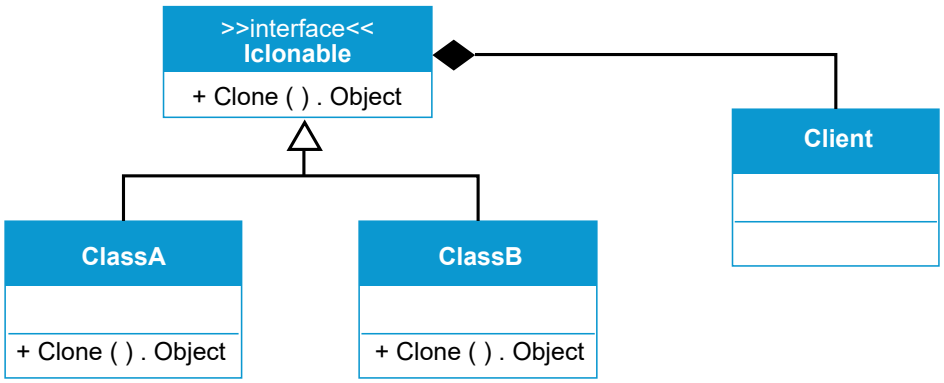


Figura 2.7. Patrón Prototipo

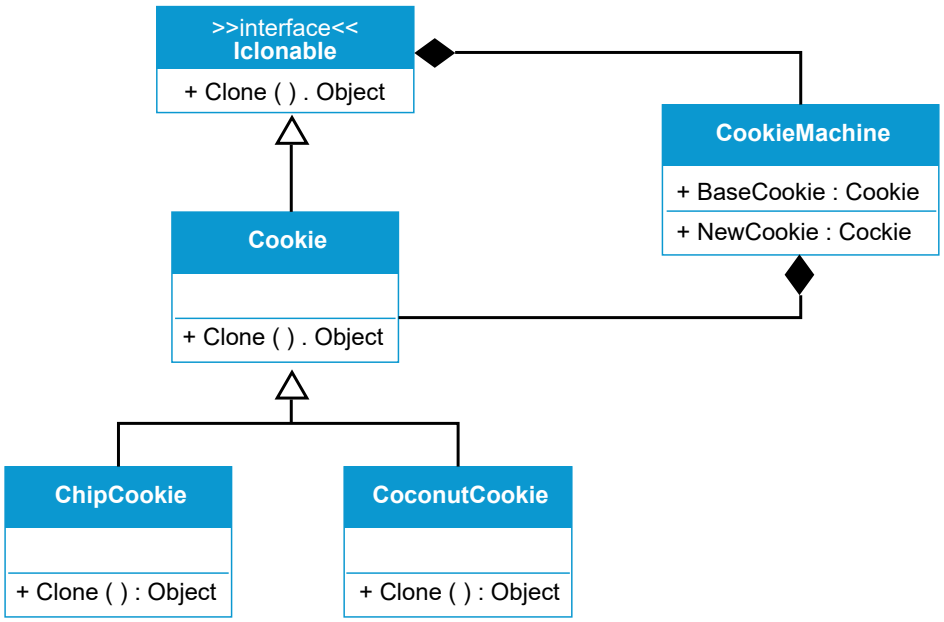


Figura 2.8. Ejemplo patrón prototipo

2.1.5. Patrón de diseño de instancia única (singleton).

Problema:
¿Qué se puede hacer para asegurar que de una clase determinada exista sólo una instancia durante toda la ejecución de la aplicación?

Solución:

- Dentro de la clase se elimina el acceso a los constructores, de modo que no sea posible construir elementos de esta clase, sólo a través de la misma clase. En la mayoría de los lenguajes esto se logra definiendo los métodos como private o protected.

- Se crea un método estático de acceso a esta instancia. Solamente la primera vez que se llame a este método se crea la instancia. Luego simplemente se retorna.

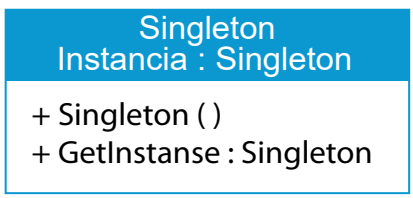


Figura 2.9. Patrón Singleton

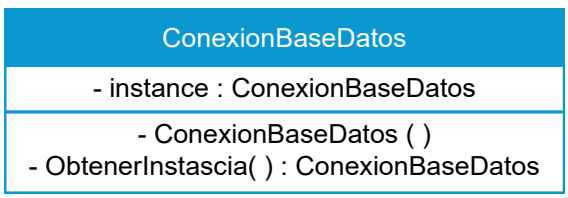


Figura 2.10. Ejemplo Patrón Singleton

2.2. Patrones estructurales.

Estos patrones describen las formas en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros, es decir la composición entre clases y objetos, se utiliza la herencia para componer interfaces y además definen formas de composición entre objetos para obtener determinada funcionalidad.

2.2.1. Patrón adaptador (adapter).

Problema:

¿Cómo hacer que una clase A utilice una clase B, si B no tiene la interfaz necesitada por A? Tener una Interfaz en su sentido más amplio. Tener métodos públicos.

¿Cómo hacer lo anterior sin cambiar B (porque no se tiene acceso, o porque está muy acoplada)?

Solución:

- Crear una clase que funcione de adaptador entre las dos clases que quieren comunicarse.
- El adaptador provee la interfaz que necesita la clase A.
- El adaptador traduce las llamadas a su interfaz en llamadas a la interfaz original (en este caso la clase B). En general se trata de transformaciones de parámetros. Pero puede llegar a cambiar las responsabilidades de la clase adaptada.
- Este patrón está pensado para situaciones en las que no se tiene acceso a la clase B, o se usa en otras partes del sistema y no se quiere modificar. Solamente la primera vez que se llame a este método se crea la instancia. Luego simplemente se retorna.

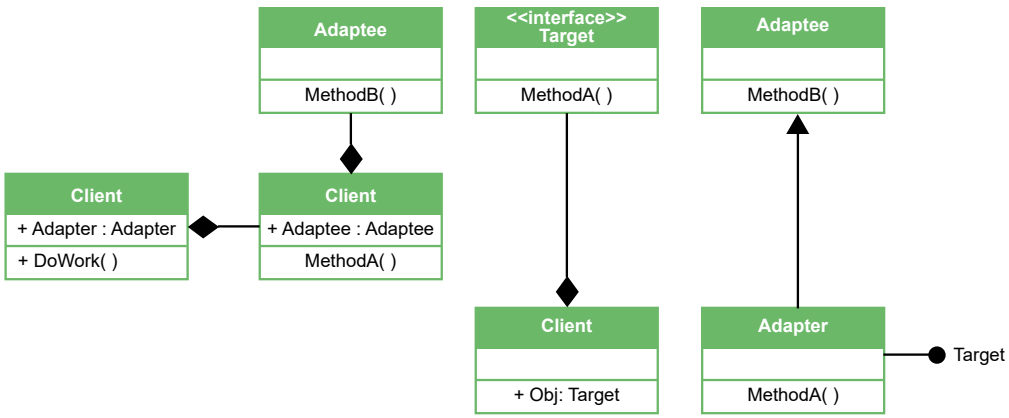


Figura 2.11. Patrón Adoptador

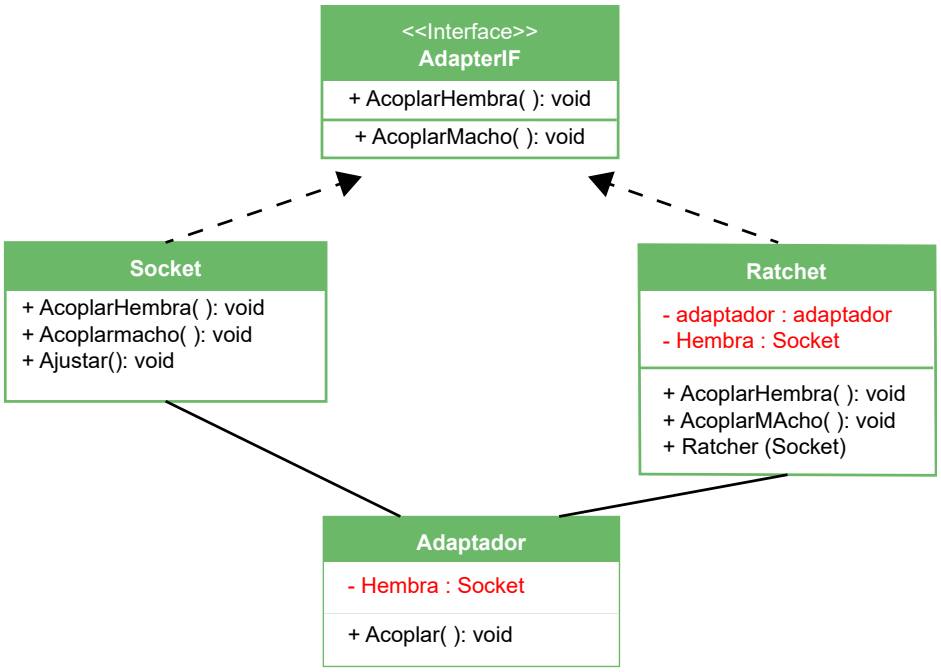


Figura 2.12. Ejemplo del patrón de diseño Adaptador

2.2.2 Patrón puente (bridge).

Problema:

¿Qué hacer para desacoplar una abstracción de su implementación de modo que ambas puedan variar independientemente? Abstracción se refiere a la interfaz.

¿Qué hacer cuando una implementación varía en tiempo de ejecución y se quiere mantener la abstracción estable?

¿Qué hacer cuando una abstracción varía en tiempo de ejecución y se quiere mantener la implementación estable?

Solución

- Se crea una clase abstracta que represente la abstracción y otra clase abstracta o interfaz que represente la implementación.
- La abstracción contiene un objeto de la implementación el cual utiliza para cumplir su propósito

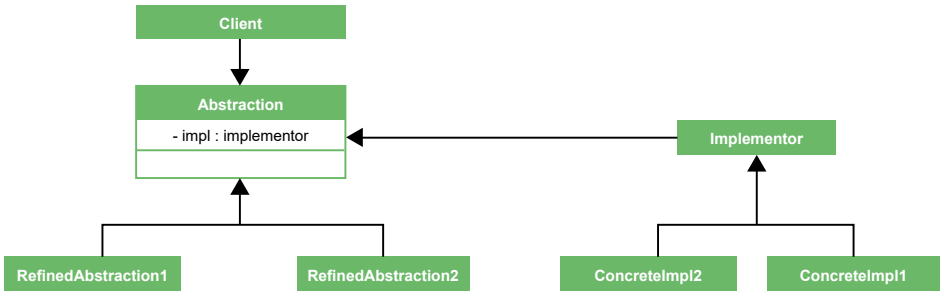


Figura 2.13. Patrón Puente (Bridge)

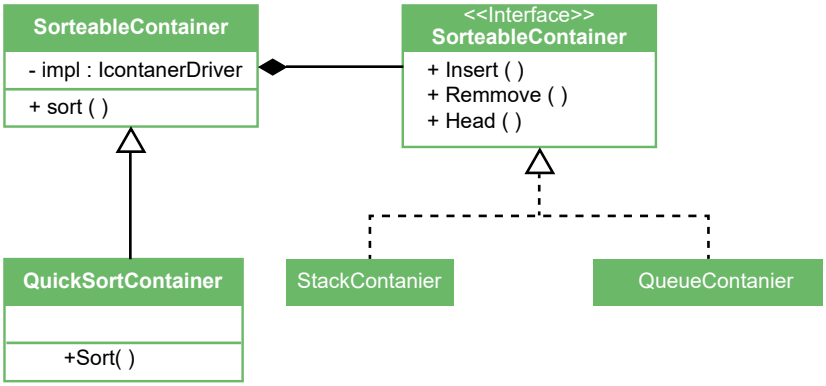


Figura 2.14. Ejemplo Patrón Puente (Bridge)

2.2.3 Patrón compuesto (composite).

Problema:

¿ Cómo tratar a un grupo de elementos de la misma forma como se trata uno de sus elementos? El grupo es una composición. El elemento único es un objeto atómico.
¿Cómo tratar a este mismo grupo si además permitimos que contenga otros grupos de los mismos elementos?

Solución:

- Definir clases para composiciones y objetos atómicos que implementen la misma interfaz.
- Los objetos se representan como una estructura de árbol.
  - Los objetos compuestos son ramas.
  - Los objetos atómicos son hojas.
- Permite la realización recursiva de las operaciones.

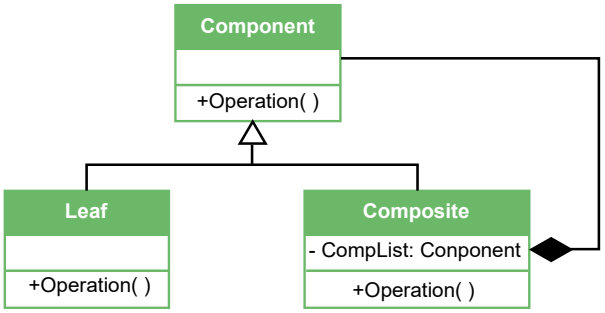


Figura 2.15 . Patrón Compuesto (Composite)

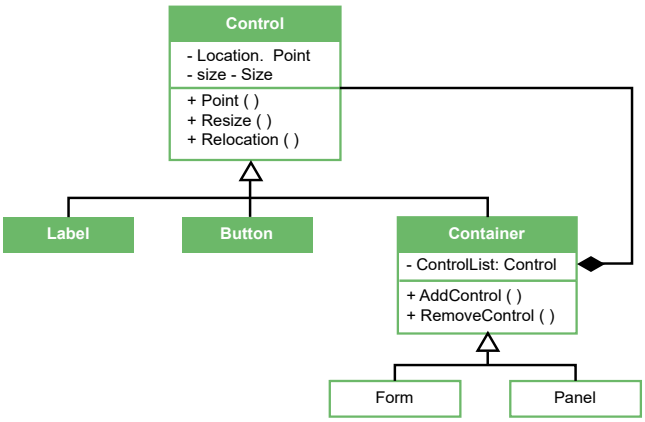


Figura 2.16 . Ejemplo Patrón Compuesto (Composite)

2.2.4. Patrón decorador (decorator).

Problema:

¿Cómo se pueden añadir nuevas funcionalidades a una clase de forma dinámica? ¿Cómo se puede decidir en tiempo de ejecución las funcionalidades que se desea que realice cierta clase? Herencia implica tener una subclase para cada combinación de funcionalidades. Para 4 funcionalidades serían 16 subclases distintas.

Solución:

- Se crea una clase que decore la clase a la cual se quiere agregar funcionalidad.
- La clase decoradora envuelve a la otra clase, pero comparte con esta la misma interfaz.

Se debe tener en cuenta que :

1. La clase decoradora tiene como atributo a la otra clase.
2. Como las dos clases comparten la interfaz, se puede tener una clase decoradora que envuelve a otra clase decoradora. De esta forma se puede elegir los adornos en tiempo de ejecución simplemente creando decoradores que envuelvan otros decoradores.

Funcionamiento recursivo similar a Compuesto.

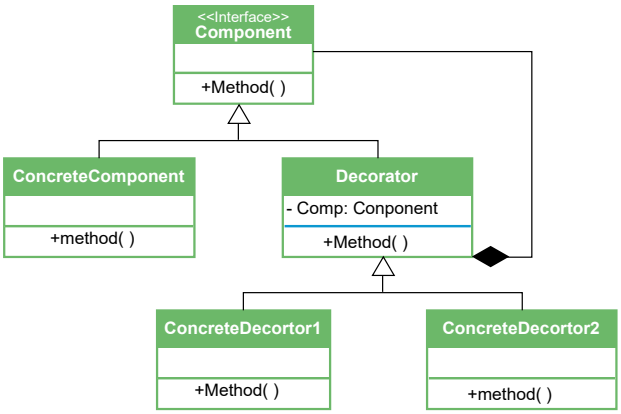


Figura 2.17. Patrón Decorador (Decorator)

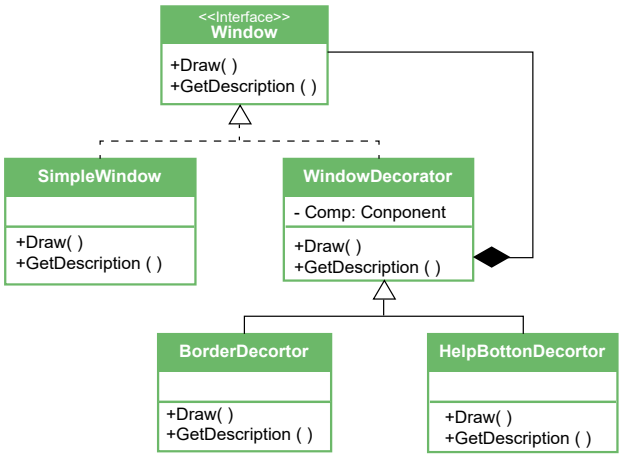


Figura 2.18. Ejemplo Patrón Decorador (Decorator)

2.2.5 Patrón fachada (facade).

Problema:

¿Cómo reducir el acoplamiento con un subsistema de clases complejas,que están sujetas a futuros cambios, pero manteniendo la funcionalidad del subsistema?

¿Cómo simplificar la realización de tareas comunes con una serie de clases de un subsistema?

Solución:

- Se crea una interfaz que exponga las funcionalidades del subsistema, es decir, un punto de contacto único con el subsistema, disminuyendo el posible acoplamiento con este. A esta interfaz se le llama la fachada del subsistema.
- La clase fachada se preocupa de trabajar con las clases del subsistema para conseguir la funcionalidad solicitada.
- Una fachada se usa para simplificar el trabajo con el subsistema.
- Es un patrón similar al patrón Adaptador. El Adaptador se utiliza cuando es necesario respetar una interfaz preexistente. Fachada se utiliza para definir una interfaz simplificada de acceso a un subsistema o conjunto de clases.

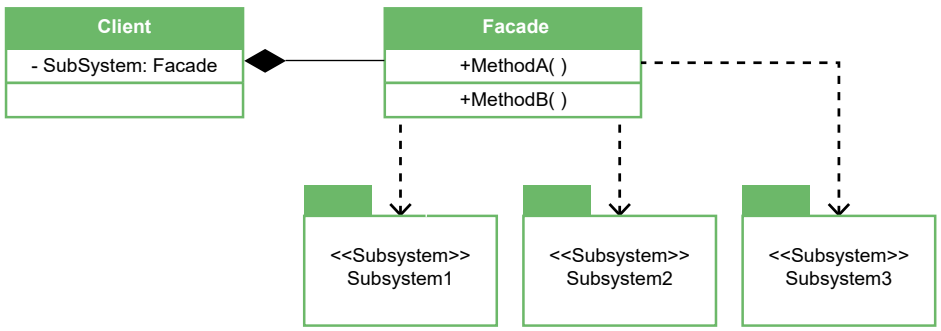


Figura 2.19. Patrón Fachada (Facade)

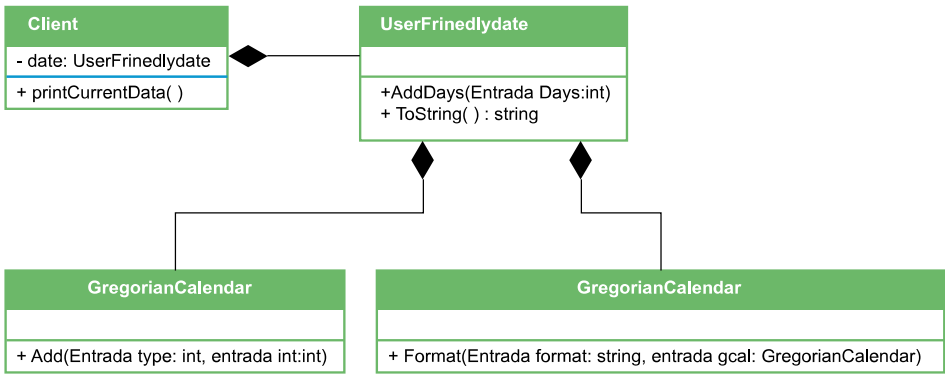


Figura 2.20. Ejemplo del patrón Fachada

2.2.6. Patrón peso ligero (FlyWeight).

Problema:

¿Cómo se puede aumentar el rendimiento o uso de memoria de una aplicación que posee elementos redundantes?

¿Cómo se pueden agrupar ciertas características de algunos elementos, de modo de asegurarnos que siempre sean iguales en todos los objetos?

Solución:

- Se define una clase Flyweight que contenga las características comunes.
- Luego todos los elementos redundantes hacen referencia a un mismo objeto de la clase Flyweight. Con lo anterior se reduce el uso de memoria y se elimina la redundancia de estas características. Las características referencian al mismo objeto y cambiar una implica cambiarla solo en un objeto.

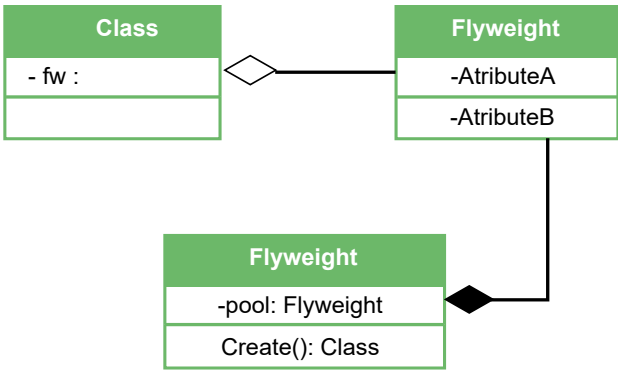


Figura 2.21. Patrón Peso Ligero (Flyweight)

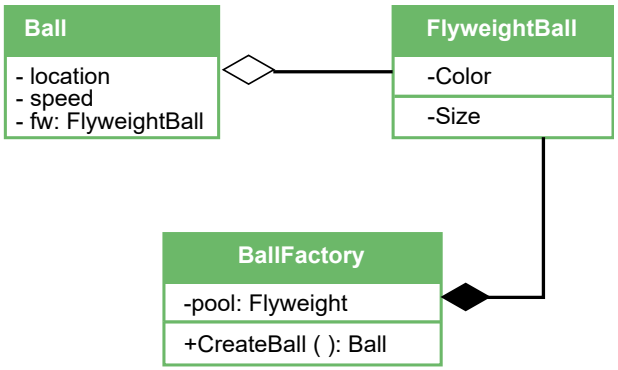


Figura 2.22. Ejemplo del patrón Peso Ligero

2.2.7. Patrón proxy.

Problema:

¿Qué hacer para controlar el uso de recursos de un objeto determinado dada su complejidad o uso de memoria?
¿Qué hacer para que los recursos se consuman solo si es necesario?

Solución:

- Se crea un intermediario con la misma interfaz que el objeto final. El intermediario contiene un objeto de la clase que se va a utilizar. Este objeto se crea solo cuando sea necesario.
- Este intermediario es el llamado proxy. Al compartir la interfaz con la clase final su uso es transparente para la clase cliente.

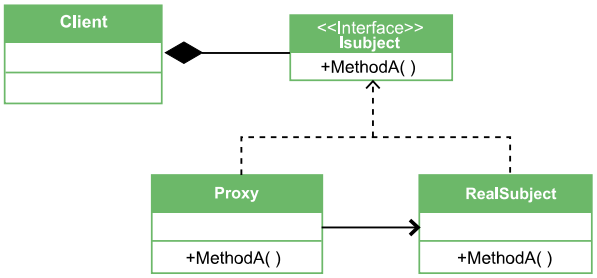


Figura 2.23. Patrón Proxy

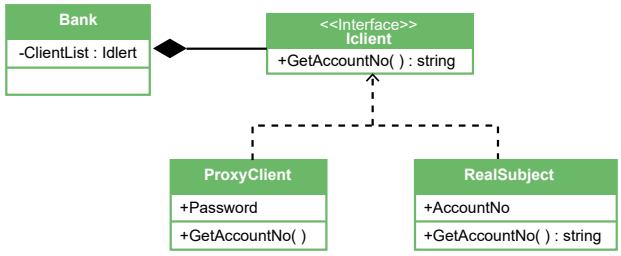


Figura 2.24. Ejemplo de patrón Proxy

2.3. Patrones de comportamiento.

Los patrones de este tipo son utilizados para organizar, manejar y combinar comportamientos.

Relación de patrones de comportamiento.

2.3.1. Cadena de Responsabilidades (Chain of Responsability): se usa para permitir que ante una petición se realicen acciones distintas dependiendo del estado del sistema.

2.3.2. Comando (Command): se usa para manejar acciones de forma independiente y generalizable, de modo que se pueden hacer invocaciones con estas acciones sin necesidad de saber quien la realiza realmente.

2.3.3. Intérprete (Interpreter): se usa para representar la gramática de un cierto lenguaje para poder interpretarlo.

2.3.4. Iterador (Iterator): se usa para recorrer objetos sin necesidad de conocer su estructura.

2.3.5. Mediador (Mediator): se usa para desacoplar la interacción entre distintos elementos, permitiendo que todos se comuniquen a través de un objeto común reduciendo así la complejidad de la interacción.

2.3.6. Recuerdo (Memento): se usa para manejar estados de la aplicación permitiendo volver a estados anteriores.

2.3.7. Observador (Observer): se usa para definir una dependencia de uno a muchos entre objetos de forma que muchos objetos puedan enterarse de los cambios en uno de ellos.

2.3.8. Estado (State): se usa para permitir que la ejecución de las acciones de una clase sea dependiente del estado en que se encuentra. El estado de un objeto puede variar en tiempo de ejecución.

2.3.9. Estrategia (Strategy): se usa cuando se quiere tener diversos algoritmos para realizar una tarea y poder cambiarlos en tiempo de ejecución.

2.3.1.0. Método Plantilla (Template Method): se usa para reutilizar el esqueleto de un algoritmo cambiando sólo el contenido de los pasos de éste último.

2.3.11. Visitante (Visitor): se usa para definir nuevas operaciones sobre una jerarquía de clases sin modificar las clases sobre las cuales opera.

3. Patrón modelo vista controlador (MVC).

MVC son las siglas de Modelo Vista Controlador, que es un patrón de arquitectura de software cuya función es subdividir una aplicación en tres módulos o capas que corresponden a la vista del usuario (la interfaz a la que accede el usuario), una lógica de control para capturar los eventos que el usuario ha generado a través de la interfaz, y un modelo que gestiona los datos según le indique la lógica de control (GRAMMA,1994).

Este patrón tiene su origen en la forma como se hacían las interfaces en el lenguaje de programación Smalltalk-80 (GRAMMA,1994).

El objetivo del patrón MVC es desacoplar la presentación de la información (vista) de su representación (modelo) para así reducir la complejidad en el diseño arquitectónico de la interfaz de usuario e incrementar la flexibilidad y mantenibilidad del código (GRAMMA,1994).

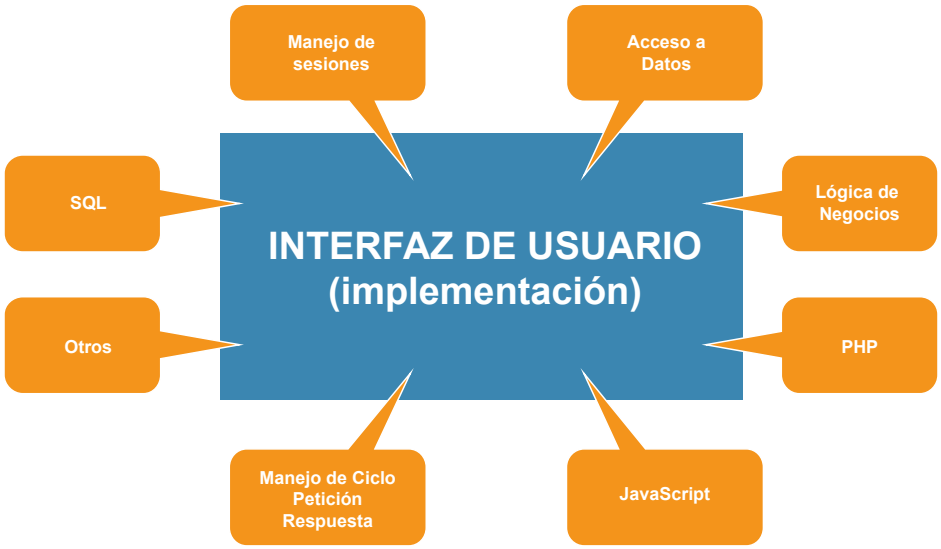


Figura 3.1. Sistema sin modelo MVC

Vista.

Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario. Por lo tanto, la vista es la encargada de presentar los datos al usuario y la interfaz necesaria para modificarlos.

Controlador.

Este responde a eventos que son generalmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

Desde este perspectiva el controlador sería la unidad central que comunica la vista con el modelo y viceversa asociando los eventos del usuario con los cambios que se producirán en el modelo.

También devuelve los datos resultantes que genere el modelo a la vista que corresponda.

Modelo.

Esta es la representación específica de la información con la cual el sistema opera y se compone por el Sistema de Gestión de Base de Datos (SGBD) y la lógica de negocio. La lógica de negocio asegura la integridad de estos y permite derivar nuevos datos.

El Sistema de Gestión de Base de Datos (SGBD) será el encargado de almacenar los cambios generados por la lógica de negocio. Ejemplos de SGBD son MySQL, Oracle, SQLServer, entre otros.

Es recomendable usar una capa de abstracción extra como las suministradas por las tecnologías Data Access Object (DAO), ODBC, JDBC, entre otras para el acceso al motor de bases de datos ya que permiten la separación completa entre el controlador y el motor de bases de datos.

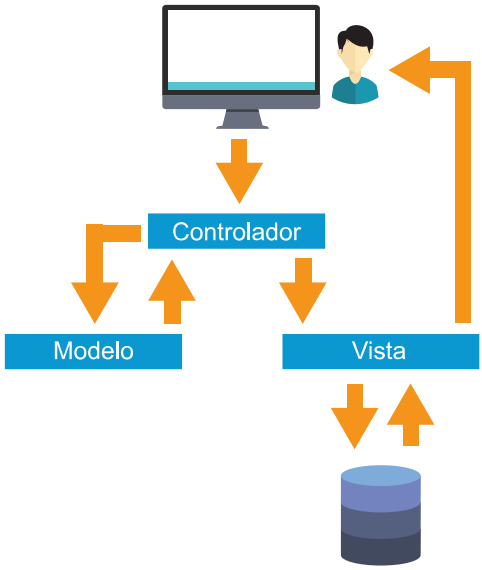


Figura 3.2. Sistema con el modelo MVC

Aunque se pueden encontrar diferentes implementaciones de MVC, el flujo de control generalmente es el siguiente:

a. El usuario interactúa con la interfaz de alguna manera (ej. presionando un botón, un enlace).
b. El controlador recibe por parte de los objetos de la interfaz vista la notificación de la acción solicitada por el usuario.
c. El controlador accede al modelo para consultar o actualizar los datos enviados por el usuario.
d. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario.
e. La vista usa el modelo para generar la interfaz apropiada para el usuario donde se refleja los cambios en el modelo.
f. En algunas implementaciones la vista no tiene acceso directo al modelo dejando que el controlador envíe los datos del modelo a la vista.
g. La interfaz espera por nuevas interacciones de usuario para iniciar nuevamente el ciclo.

4. Patrones de arquitectura para aplicaciones empresariales.

Además de los patrones introducidos por la banda de los cuatro (GoF) y el patrón MVC introducido en el lenguaje Smalltalk-80 existen otros aportes como los expuestos por Martin Fowler en su libro: Patrones de arquitectura para aplicaciones empresariales (Patterns of Enterprise Application Architecture) .

En su libro Fowler introduce una serie de patrones de común uso en el ámbito de aplicaciones empresariales especialmente las que acceden bases de datos y trabajan con usuarios de manera simultánea o concurrente.

En este recurso se introduce el patrón “registro activo” que puede ser usado de manera intensiva en la etapa de desarrollo del sistema de información. El aprendiz puede profundizar los demás patrones de Martin Fowler a través de la bibiografía que se especifica en este recurso.

4.1 patrón registro activo (active record).

Es un objeto que representa un registro de una tabla o vista de una base de datos. Además encapsula el acceso a la base de datos y agrega la lógica del dominio sobre esos datos (FOWLER,2002).

Este patrón se usa con mucha frecuencia en las aplicaciones que utilizan formularios o formas para capturar un registro de la base de datos que luego será modificado o para capturar un registro que será agregado a esa misma base de datos.

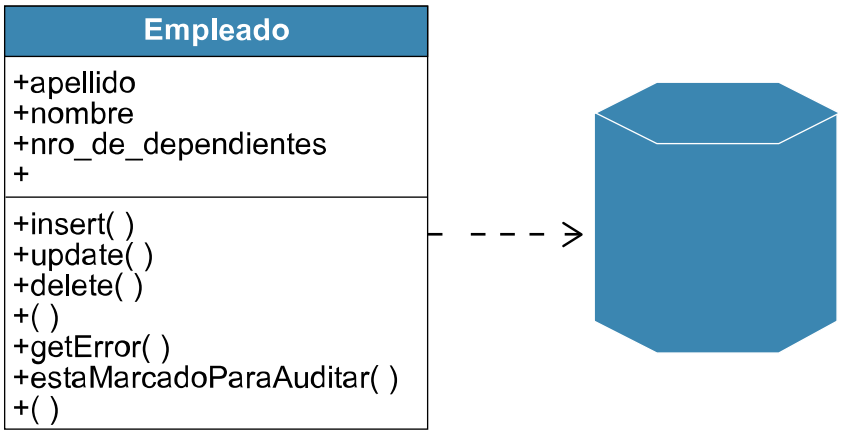


Figura 4.1. Patrón Registro Activo

La estructura de datos de un registro activo debe coincidir exactamente con la de la base de datos, es decir, un campo de la clase representar cada campo del registro en la base de datos.(FOWLER,2002)

El registro activo generalmente tiene métodos para hacer lo siguiente (FOWLER,2002):

a. Construir una instancia de la clase Registro Activo a partir de los resultados de una sentencia SQL.
b. Construir una instancia para luego insertar en la base de datos.
c. Métodos de búsqueda para representar sentencias SQL comúnmente usadas y retornar objetos del tipo Registro activo.
d. Actualizar la base de datos e insertar en esta los datos del Registro Activo.
e. Métodos para traer y definir los campos (setters and getters).
f. Métodos para implementar algunas reglas de negocio.

5. Cómo elegir un patrón de diseño.

Para seleccionar el patrón adecuado se pueden seguir los siguientes pasos:

1. Entienda el contexto donde quiere aplicar el patrón
2. Revise los patrones ya estudiados
3. Identifique y estudie los patrones que pueden aplicar
4. Aplique el patrón de la manera adecuada.

Al usar los patrones de diseño se debe evitar lo siguiente:

1. Usar un patrón equivocado.
2. Usar demasiados patrones que puedan complicar la implementación.

GLOSARIO

DAO: Data Access Object. Software intermedio que permite el acceso a una base de datos.

Encapsulamiento: propiedad de la programación orientada a objetos en la cual sólo se hacen visibles algunos atributos y métodos de estos últimos.

Herencia: cualidad de la programación orientada a objetos en la cual estos últimos tienen los atributos y métodos de una clase superior.

JDBC: Java Database Connectivity. Software intermedio que permite el acceso a bases de datos.

MVC: Model-View-Controller. Tipo de arquitectura que divide la interacción entre el usuario y el sistema en tres capas denominadas Modelo-Vista-Controlador.
ODBC: Open DataBase Connectivity : Software intermedio que permite el acceso a bases de datos.

Patrón de diseño: solución probada y documentada a un problema de diseño de software.

Polimorfismo: propiedad de la programación orientada a objetos que permite que un mismo objeto se comporte de manera diferente de acuerdo a los parámetros de entrada.

SGBD: sistema de gestión de bases de datos.

BIBLIOGRAFÍA

Gramma, E., Helm, R., Johnson, R., Vlissides, J., (1994). *Design Patterns: Elements of Reusable Object-Oriented Software.* USA: Addison-Wesley.

Fowler, M. (2002). *Patterns of Enterprise Application Architecture.* USA: Addison-Wesley.

CONTROL DEL DOCUMENTO

CONSTRUCCIÓN
OBJETO DE
APRENDIZAJE

SENA

PATRONES DE DISEÑO	
Centro Industrial de Mantenimiento Integral - CIMI Regional Santander	
Líder línea de producción:	Santiago Lozada Garcés.
Asesores pedagógicos:	Rosa Elvia Quintero Guasca. Claudia Milena Hernández Naranjo.
Líder expertos temáticos: **Experto temático:** **Experto temático:**	Rita Rubiela Rincón Badillo. César Marino Cuellar. (V1) Nelson Mauricio Silva Maldonado.(V2)
Diseño multimedia:	Tirso Fernán Tabares Carreño.
Programador:	Francisco José Lizcano Reyes.
Producción de audio:	Víctor Hugo Tabares Carreño.

creative commons

Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de la licencia que el trabajo original.

BY NC SA



CONVERSIÓN MER A MODELO RELACIONAL

SENA

FAVA - Formación en Ambientes Virtuales de Aprendizaje

SENA - Servicio Nacional de Aprendizaje.

Estructura de contenidos

Pág.

Conversión MER a Modelo Relacional 3
Introducción 3
Mapa de contenido 4
1. Modelo ER a modelo relacional 5
1.1 Transformación mer a modelo relacional 5
1.2 Reglas generales de transformación 6
1.2.1 Transformación para las entidades y atributos 6
1.2.2 Transformación para las relaciones 8
2. Transformacion MER-extendido a relacional 12
3. Construccion transformacion MER-extendido a relacional 15
3.1 Relación de especialización y generalización 15
3.2 Relación recursiva 17
4. Normalizacion 20
4.1 Grado de normalización 22
4.1.1 Primera forma normal 1n 22
4.1.2 Segunda forma normal 2n 23
4.1.3 Tercera forma normal 3n 24
4.1.4 Forma normal boyce-cood (FNBC o BCFN) 25
4.1.5 Cuarta forma normal (4fn) dependencias multievaluadas 26
4.1.6 Quinta forma normal (5fn) dependencia de join o de reunión 27
Glosario 31
Bibliografía 32
Control del documento 33

Conversión MER a Modelo Relacional

Introducción



Almacenar información es uno de los usos más frecuentes que una organización o empresa realiza día a día en un computador. La construcción e implementación de bases de datos permite que la organización pueda almacenar la información relevante del negocio por medio del modelado de datos. En este proceso están identificados tres modelos:

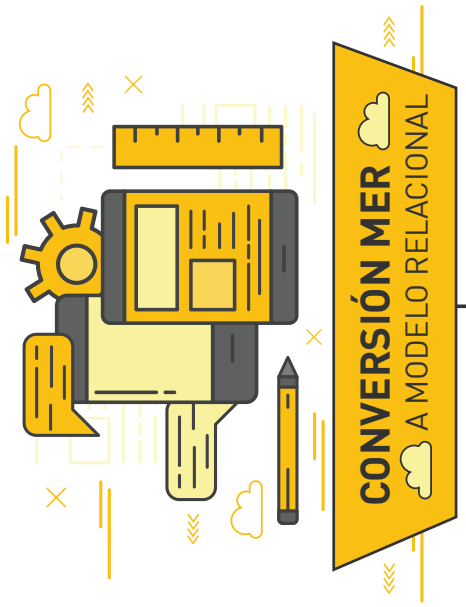
- Modelo Entidad Relación (MER).
- Modelo Relacional.
- Modelo Físico.

En este material de consulta se presentará el proceso de construcción de un modelo relacional a partir de un MER, lo cual se denomina:

“Transformación del MER a Modelo Relacional”.

La intención es que el aprendiz logre desarrollar y despertar su interés en el modelado de datos, utilizando dicho modelo para la creación e implementación de una base de datos relacional.

Mapa de contenido



CONVERSIÓN MER A MODELO RELACIONAL

PARA TRANSFORMAR SE UTILIZA

- REGLAS GENERALES DE TRANSFORMACIÓN
  - UTILIZANDO
    - TRANSFORMACIÓN MER EXTENDIDO RELACIONAL
      - A TRAVES DE
        - RELACIÓN DE ESPECIALIZACIÓN Y GENERALIZACIÓN
        - RELACIÓN RECURSIVA
    - TRANSFORMACIÓN MER EXTENDIDO A RELACIONAL
    - NORMALIZACIÓN
      - A TRAVES DE
        - PRIMERA FORMA NORMAL (1FN)
        - SEGUNDA FORMA NORMAL (2FN)
        - TERCERA FORMA NORMAL (3FN)
        - FORMA NORMAL BOYCE - CODD
        - CUARTA FORMA NORMAL (4FN)
        - QUINTA FORMA NORMAL (5FN)
    - TRANSFORMACIÓN PARA LAS ENTIDADES Y ATRIBUTOS

Desarrollo de contenidos

1. Modelo ER a modelo relacional

1.1 Transformación MER a modelo relacional

Uno de los pasos iniciales para la construcción de una base de datos es el modelado de ellos por medio de un Modelo Entidad Relación (MER), modelo que permite por medio de entidades, atributos y relaciones acercarse a una representación preliminar del diseño de una base de datos.

Luego de terminar un MER debe entrarse en el diseño de un modelo lógico, el cual es llamado modelo relacional. La forma como se logra desarrollar una base de datos relacional es a través del modelado y la metodología de diseño de una base de datos puede estar enfocada en los siguientes 3 pasos:

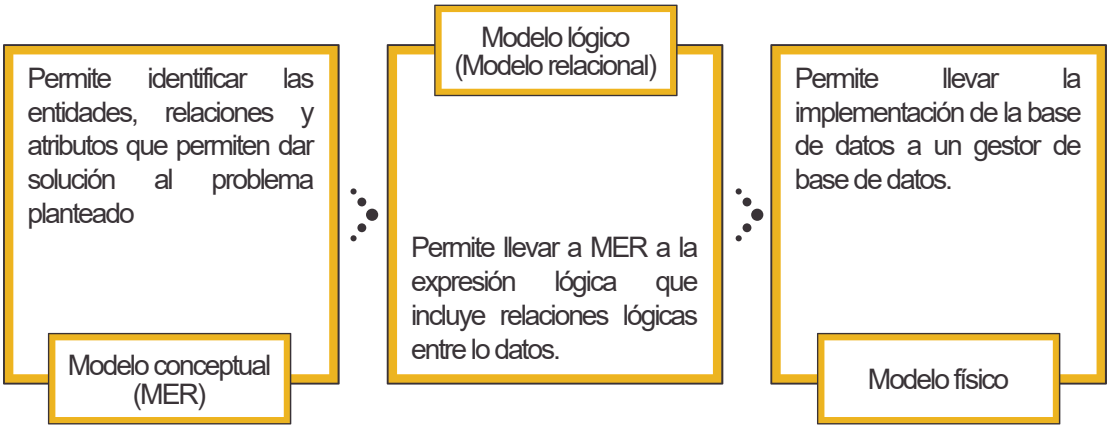


Figura 1. Pasos para la conversión del modelo ER a Relacional

En estos momentos se abordará el proceso de la construcción de un modelo relacional a partir de un MER.

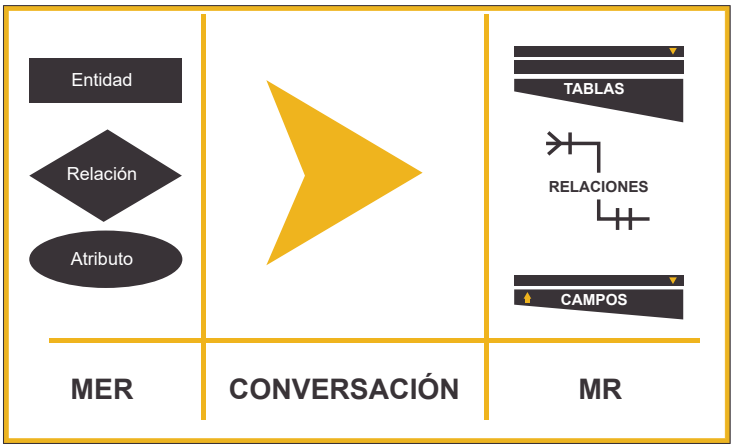


Figura 2. Modelo de conversión MER a MR

1.2 Reglas generales de transformación

La transformación de un Modelo Entidad Relación a Relacional, se puede expresar mediante un esquema relacional o un modelo relacional.

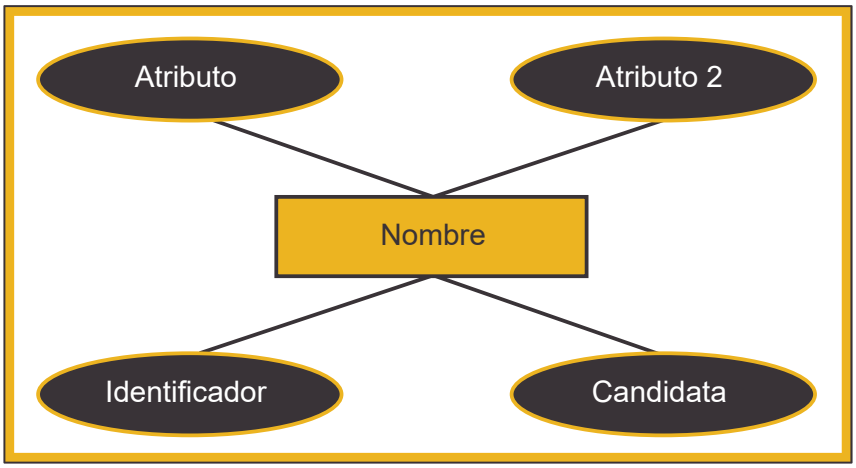


Figura 3. Esquema relacional

El esquema relacional consta de palabras y relaciones mediante flechas que indican la asociación de las entidades; el modelo relacional es un dibujo que representa las tablas, atributos, llaves primarias y llaves foráneas según la transformación realizada.

A continuación se explica detalladamente la transformación de un MER según las reglas para las entidades y se expresa en esquema relacional y modelo relacional.

NombreTabla(Identificador, Atributo1, Atributo2, Candidata)

1.2.1 Transformación para las entidades y atributos

- Entidades. Las entidades pasan a ser tablas.
- Atributos. Los atributos pasan a ser columnas o atributos de la tabla.
- Identificadores principales. Pasan a ser claves primarias.
- Identificadores candidatos. Pasan a ser claves candidatas.

La siguiente imagen ilustra los anteriores elementos:

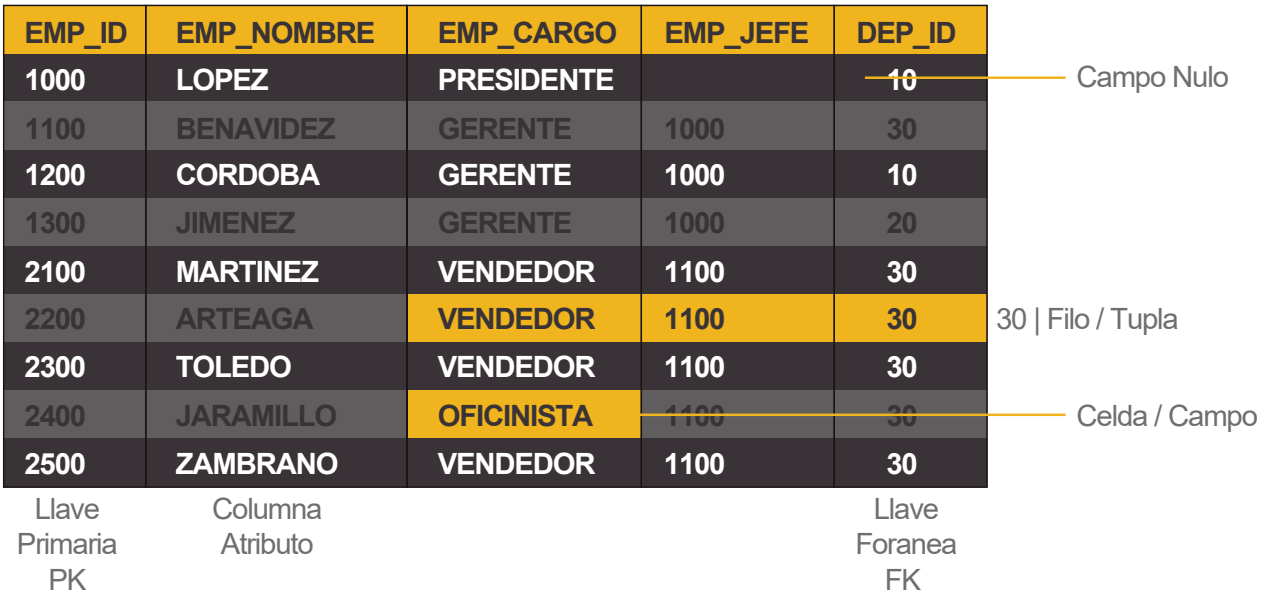


Figura 4. Transformación para las entidades y atributos

Por ejemplo:

Si se tiene la siguiente entidad Aprendiz se convierte entonces:

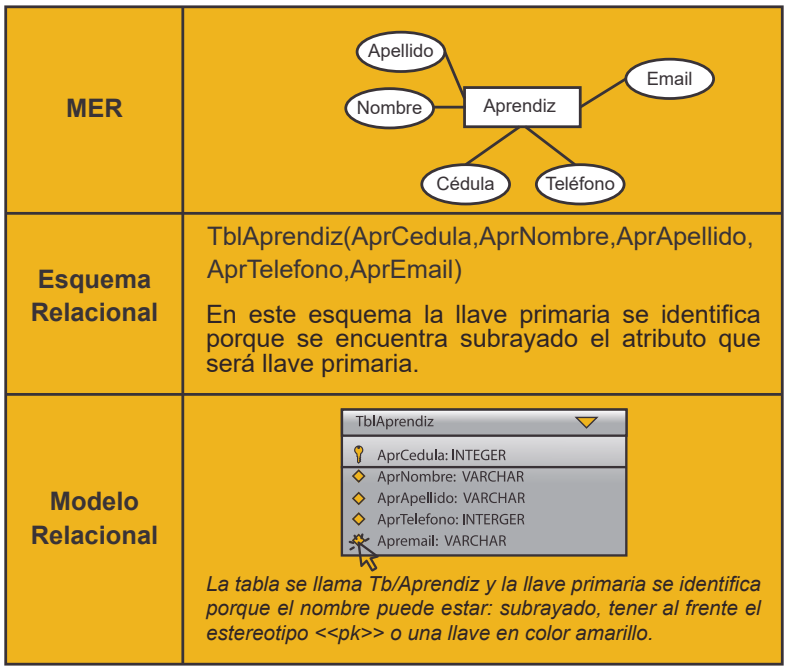


Figura 5. Transformación para las entidades y atributos

> ***NOTA:*** *para garantizar unicidad de estilos se debe recordar que debe tener una estandarización para nombrar las tablas y los atributos. Por ejemplo para este caso la entidad aprendiz se convierte en la tabla aprendiz y es llamada como TblAprendiz, el estilo de nombrado es iniciar con las tres primeras letras Tbl para indicar que es tabla, seguido del nombre de la entidad e iniciando con la primera letra en mayúscula.*

Para el atributo Apellido, en el modelo relacional se convierte en AprApellido indicando Apr que viene de la tabla Aprendiz y seguido del nombre del atributo cada uno iniciando con Mayúscula la primera letra.

1.2.2 Transformación para las relaciones

Relación muchos a muchos: esta relación se distingue en un MER porque su cardinalidad entre las dos entidades esta designada por (M..M) o (N..M).

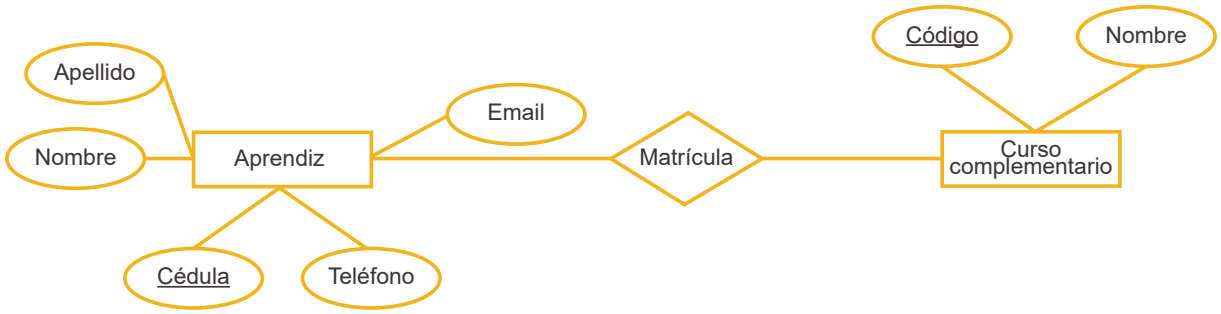


Figura 6. Relación muchos a muchos

- **Transformación relación muchos a muchos**: cuando se tiene una relación de este tipo se debe crear una nueva relación (tabla) la relación se transforma en una tabla cuyos atributos son: los atributos de la relación y las claves de las entidades relacionadas (que pasarán a ser claves externas). La clave de la tabla la forman todas las claves externas.

- **Transformación a esquema relacional**: cuando se hace la transformación a esquema relacional se debe tener un esquema de nombrado para las tablas que se generaran, por ejemplo para este caso la entidad aprendiz la tabla generada se denomina TblAprendiz, anteponiéndole tres letras Tbl que indican que es una tabla; de la misma manera para los atributos de esta entidad se le antepone las tres letras Apr para indicar que son atributos de la tabla Aprendiz. Así se continuara el estilo de nombrado para todos los casos de estudio y ejemplos de este material.

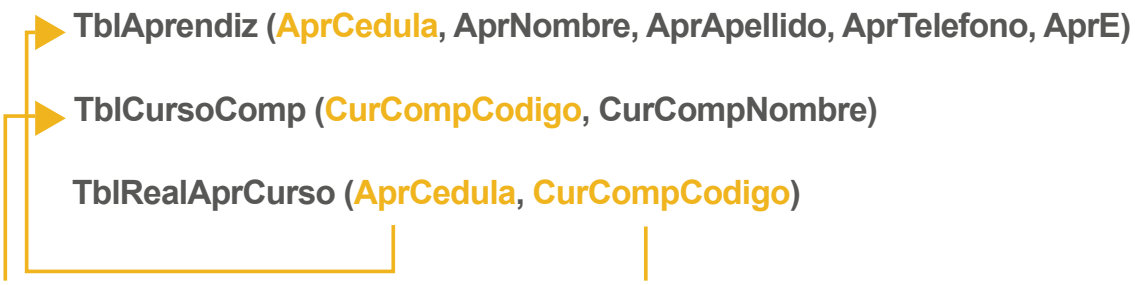


Figura 7. Esquema relacional

- **Transformación a modelo relacional**: para realizar las transformaciones a modelo relacional se puede utilizar herramientas case, para estos ejemplos se ha utilizado el software Día, el uso de ésta última se detallará en los ejercicios que acompañan el laboratorio correspondiente.

  Se visualiza la nueva tabla donde las llaves primarias son a la vez llaves foráneas, nótese que la cardinalidad de la relación está representada por los rombos en blanco (1) o sombreados (M).

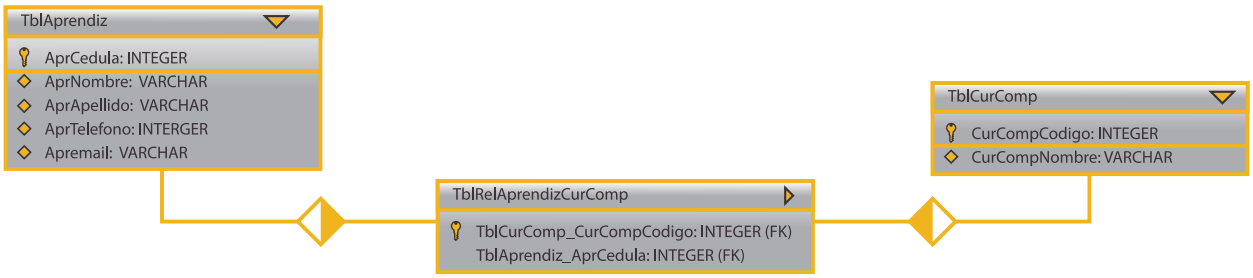
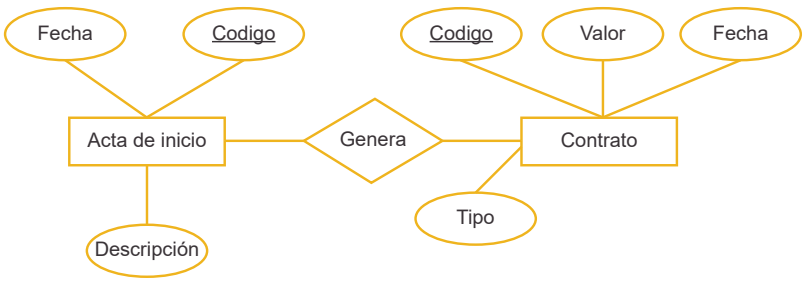


Figura 8. Transformación a modelo relacional

La nueva tabla creada es llamada TblRelAprendizCurComp, que tiene incluida las dos llaves primarias de la tabla TblAprendiz y TblCurComp. Estas llaves en la nueva tabla son llamadas llaves foráneas y son representadas por un <fk> en el modelo.

El jueves se le pregunta al Experto si decidió sacar la siguiente información o se le pasó tenerla en cuenta:

Nota: *las llaves primarias pasada a la nueva tabla se llaman llaves foráneas. 3.2.2 Relación 1 a 1 (1..1): esta relación se distingue en un MER porque su cardinalidad entre las dos entidades esta designada por (1..1).* Ejemplo:



Transformación Relación 1 a 1 (1..1): cuando se tiene una relación de este tipo se envía la llave primaria de una tabla a la otra tabla que participa de la relación, se envía la llave primaria que se considere sea más relevante para enviar a la otra tabla.

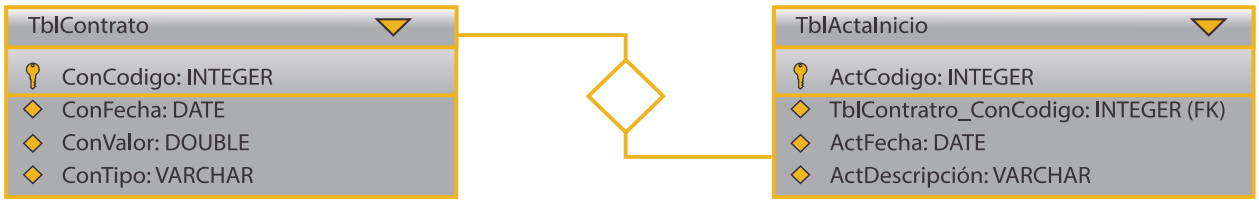
Esquema relacional

TblContrato (ConCodigo, ConValor, ConFecha, ConTipo)

TblActaInicio (ActCodigo, ActFecha, ActDescripción, TblContratoConCodigo)

Para este ejemplo específico se ha pasado la llave primaria del contrato (ConCodigo) a la tabla ActaInicio.

Transformación a modelo relacional



En la tabla TblActaInicio se incluye la clave primaria de la tabla TblContrato (ConCodigo) quedando como llave foránea.

Relación uno a muchos: esta relación se distingue en un MER porque su cardinalidad entre las dos entidades esta designada por (1..M).

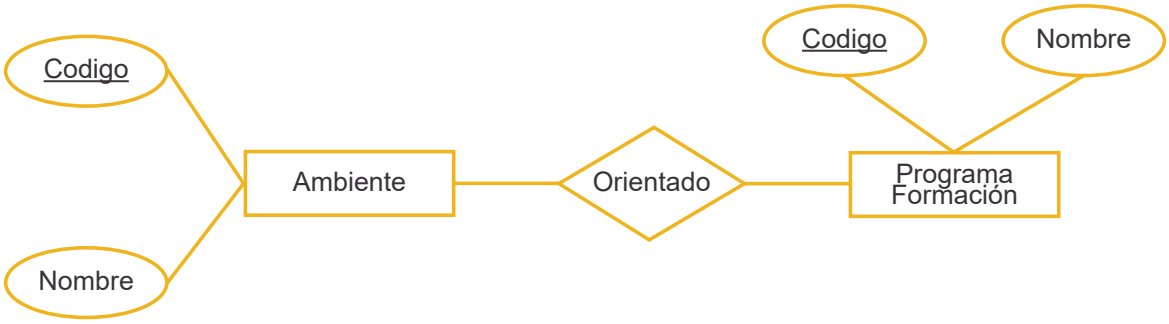


Figura 9. Relación uno a muchos

- **Transformación relación uno a muchos**: cuando se tiene una relación de este tipo se debe colocar o enviar la llave primaria de la tabla que tiene en cardinalidad 1 a la tabla que tiene cardinalidad M.

Transformación a esquema relacional:

TblAmbiente (AmbCodigo, AmbNombre)

TblProgramaFormación (PrgForCodigo, PrgForNombre, AmbCodigo

Figura 10. Esquema relacional uno a muchos

Visualizar que La tabla TblAmbiente tiene como cardinalidad 1, por la tanto la clave primaria de esta tabla es pasada a la tabla TblProgramaFormacion.

- Transformación a modelo relacional: Notar que la clave de la tabla TblAmbiente es pasada a la Tabla TblProgramaFormacion la cual tiene su estereotipo <fk>.

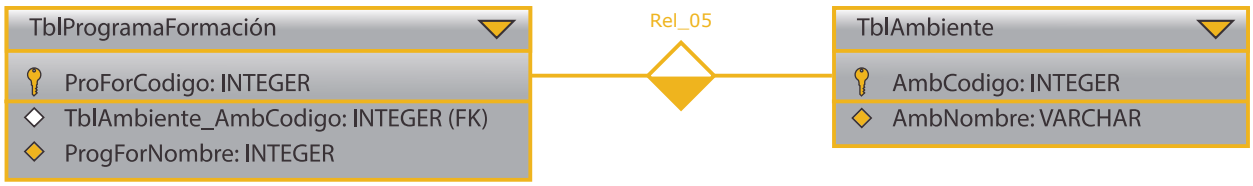


Figura 11. Esquema relacional uno a muchos

2. Transformacion MER-extendido a relacional

A continuación se presenta un ejemplo de transformación de modelo entidad relación a Relacional, que aplica las reglas descritas. Para este ejemplo se toma una situación problemática y su modelo conceptual, para lo cual se presenta la descripción del problema, el MER y el modelo relacional resultante.

Descripción del problema.

En el SENA Regional Cauca del Centro de Teleinformática y Producción industrial el coordinador requiere un sistema de información que permita gestionar los horarios de los instructores pertenecientes a un centro de formación que orientan formación profesional integral, tanto de planta como de contrato, tenga en cuenta que un instructor de contrato puede tener su relación de contrato por horas o asignación mensual.

Para lo cual es necesario conocer: un instructor orienta formación profesional (clase) en un programa de formación (el programa de formación se refiere a las tecnologías o tecnólogos que ofrece el SENA), o en un curso complementario (los cuales son cursos de duración de 2 a 3 meses y solicitados por empresas del sector para capacitar a sus empleados).

Ambos: programa de formación y complementarios son identificados por un código.

Cuando un aprendiz se matricula a un programa de formación o curso complementario tiene que durante toda su etapa lectiva (clase), cursar competencias laborales.

Una competencia podría asemejarse como una asignatura que un aprendiz debe cursar, pero para este caso siempre manejaremos el término competencia.

Asignación de competencias: para asignar una competencia a un instructor es necesario que esté legalmente contratado por el centro o que tenga resolución de nombramiento y según su especialidad se le asignan las competencias que va a orientar.

Ambientes de formación: es el sitio específico (salón) donde se orientaran las competencias asignadas. Cada ambiente tiene una codificación asignada.

Reglas del negocio: un instructor de contrato puede iniciar labores una vez tenga el acta de inicio.

Tener en cuenta que un instructor puede laborar de lunes a viernes de 7:00 a 10:00 p.m. de la noche, sábados y domingos de 8:00 a 6:00 p.m.

Un instructor puede orientar diferentes competencias en diferentes programas de formación y cursos complementarios.

Un instructor no puede orientar una competencia en diferentes ambientes.

Dos o más instructores pueden orientar una o más competencias en un mismo ambiente.

Usuarios: los usuarios que interactúan con la base de datos.

Coordinador académico.

Instructores del SENA centro de Teleinformática y PI Regional Cauca.

Informes por Usuario: los informes necesarios para mostrar a los usuarios identificados son:

- Visualizar el horario de todos los instructores de planta.
- Visualizar el horario de todos los instructores de contrato.
- Visualizar el horario de un solo instructor.
- Visualizar el horario de todos los ambientes.
- Visualizar el horario de un solo ambiente.
- Visualizar el horario de todos los programas de formación.

Modelo entidad relación

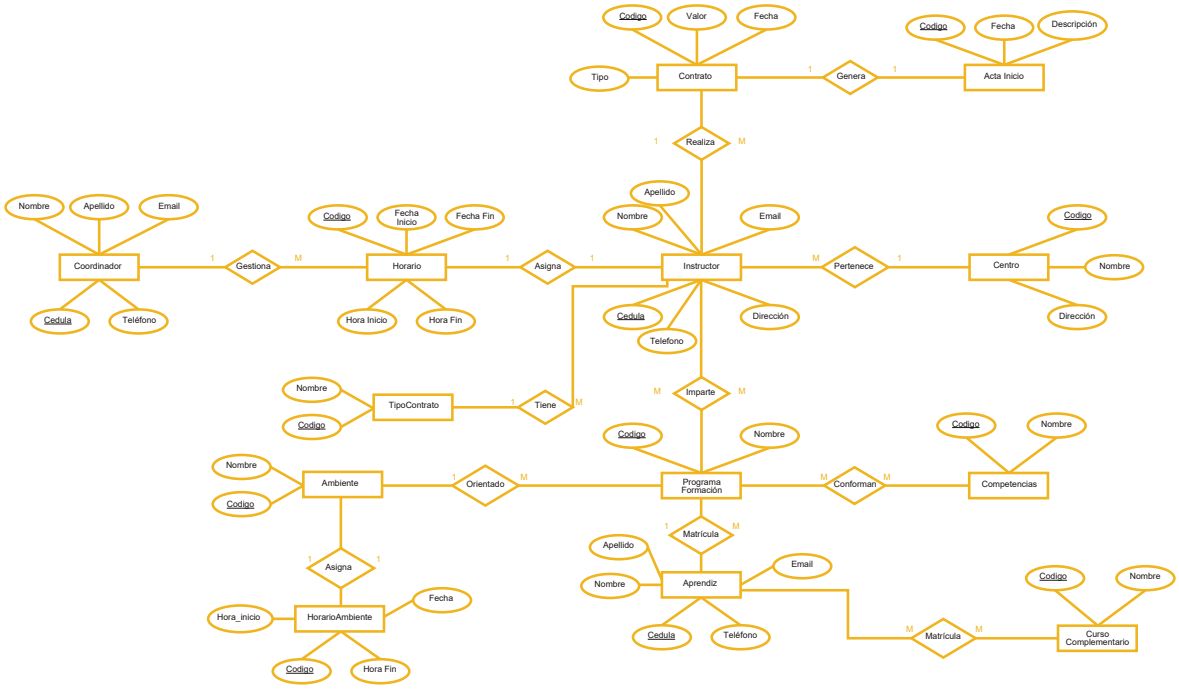


Figura 12. Modelo entidad relación

Modelo relacional

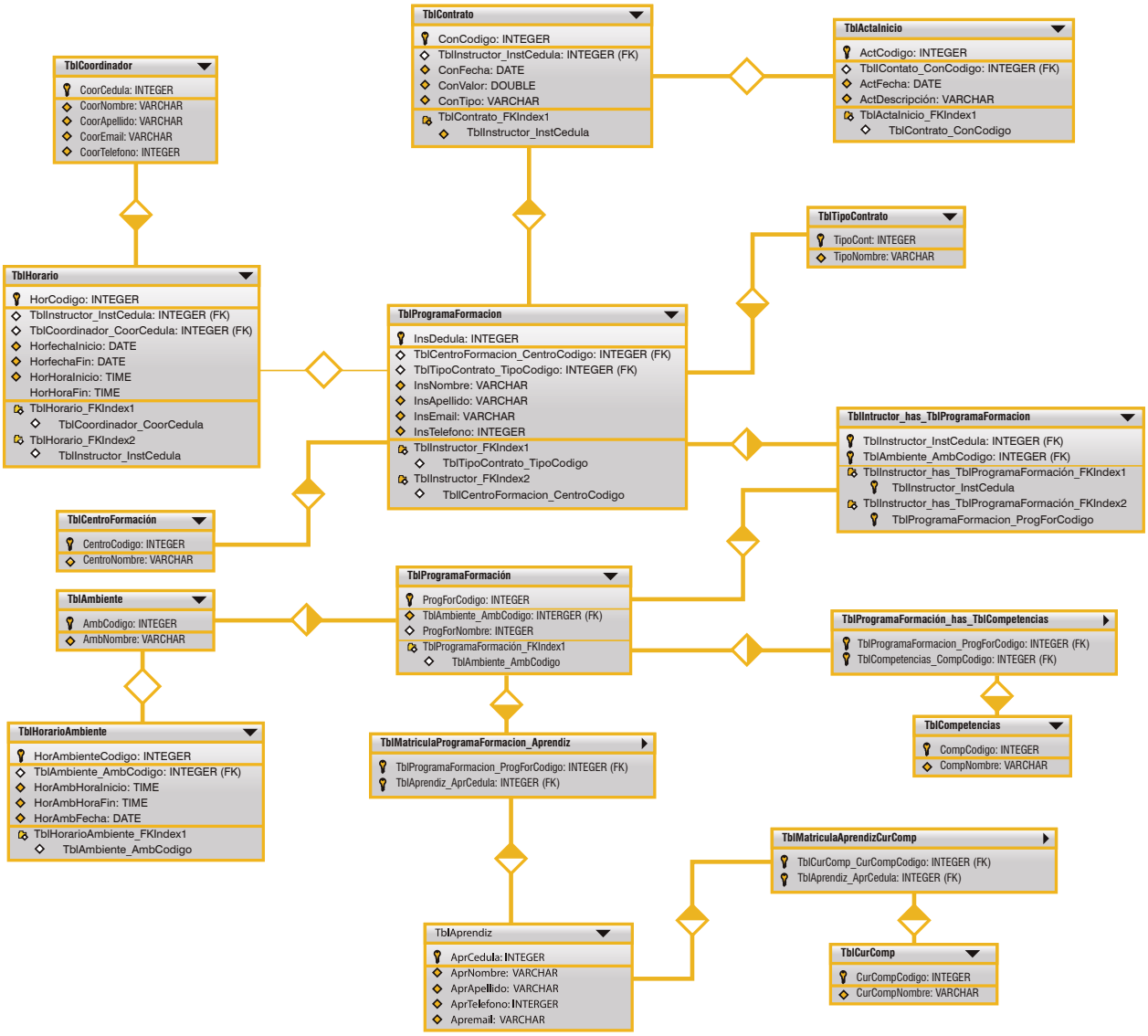


Figura 13. Modelo relacional

3. Construccion transformacion MER-extendido a relacional

El MER-Extendido ha sido creado para modelar otros tipos de relaciones que no contempla el MER tradicional, los cuales son la generalización y especialización, recursivas, inclusividad y exculsividad. A continuación se muestra como se transforma cada una de estas relaciones a un modelo relacional.

3.1 Relación de especialización y generalización

Recordar que en un MER-Extendido la generalización puede identificarse como solapamiento, exclusividad, parcialidad o totalidad, en estas relaciones se encuentran entidad superclase o supertipos y entidades subclase o subtipos.

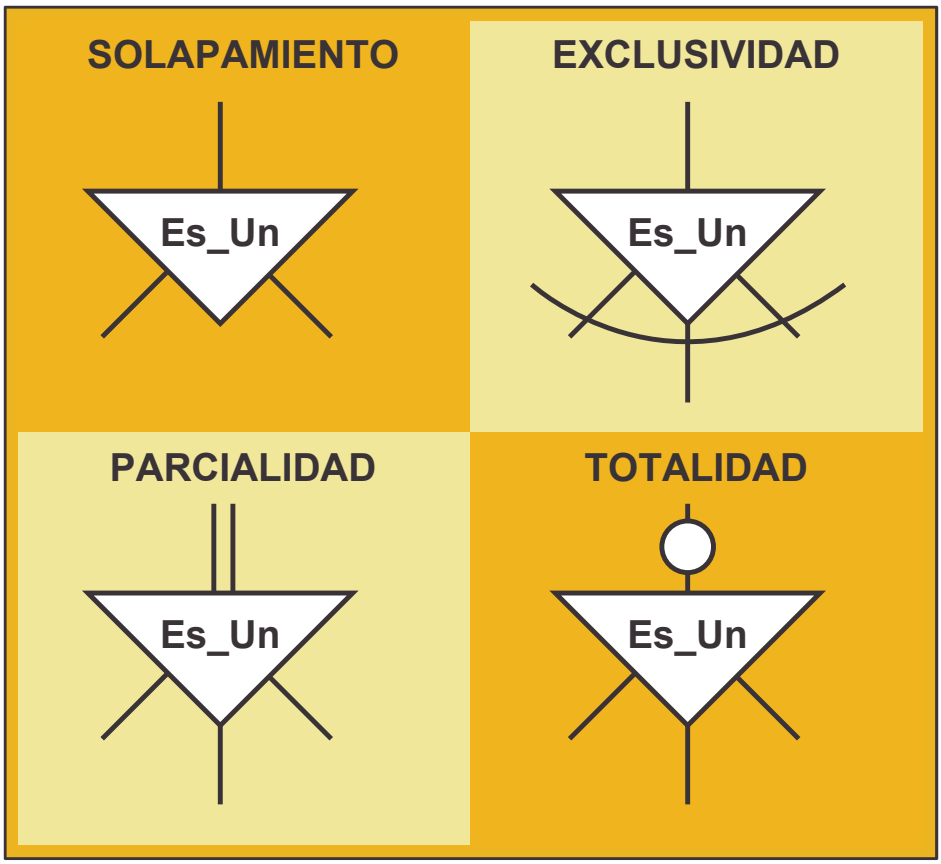


Figura 14. Relación de generalización de un MER extendido

Existen cuatro opciones para la transformación de estos tipos de relaciones.

1. Si es solapada y parcial: crear una relación (tabla) para la superclase, con sus atributos correspondientes y una relación (tabla) para cada subclase con sus atributos incluyendo la clave primaria de la superclase.

Generalización: esta representación es solapada y parcial, es decir una persona puede ser profesor y estudiante a la vez y no es obligatorio que asuma un subtipo.

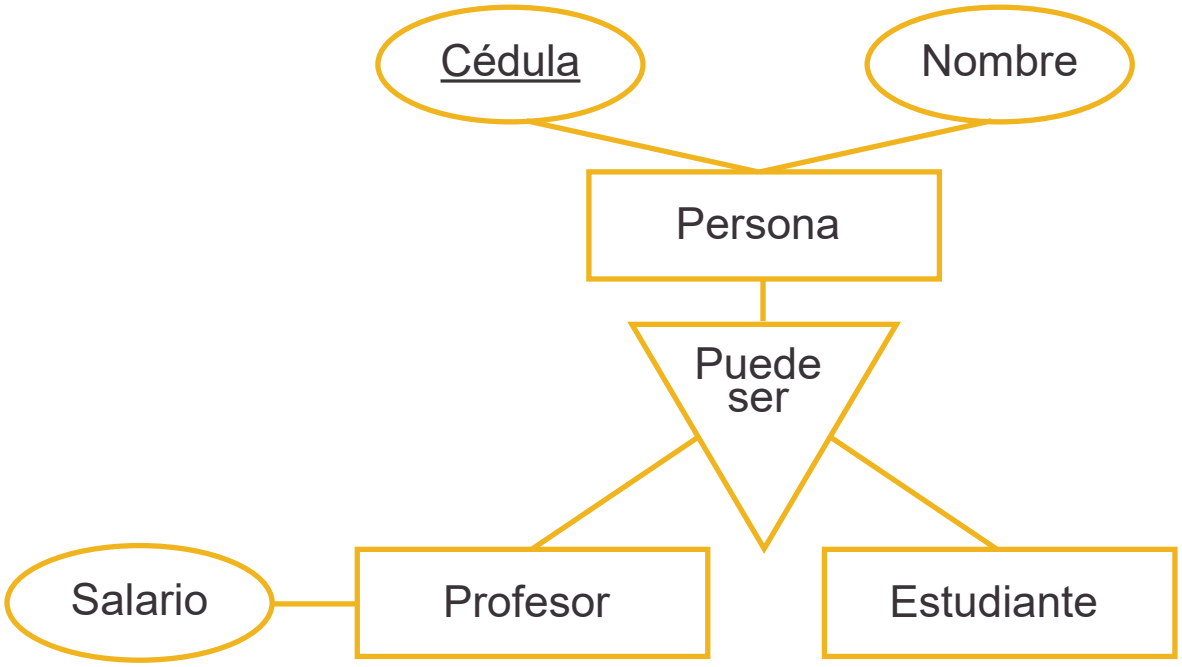


Figura 15. Generalización

Transformación a esquema relacional.

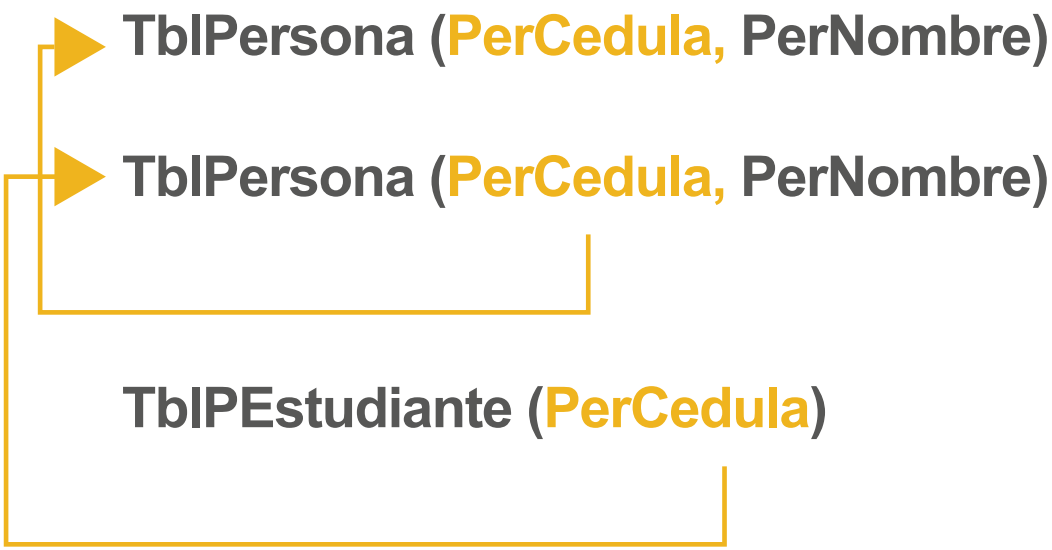


Figura 16. Transformación a esquema relacional

Transformación modelo relacional

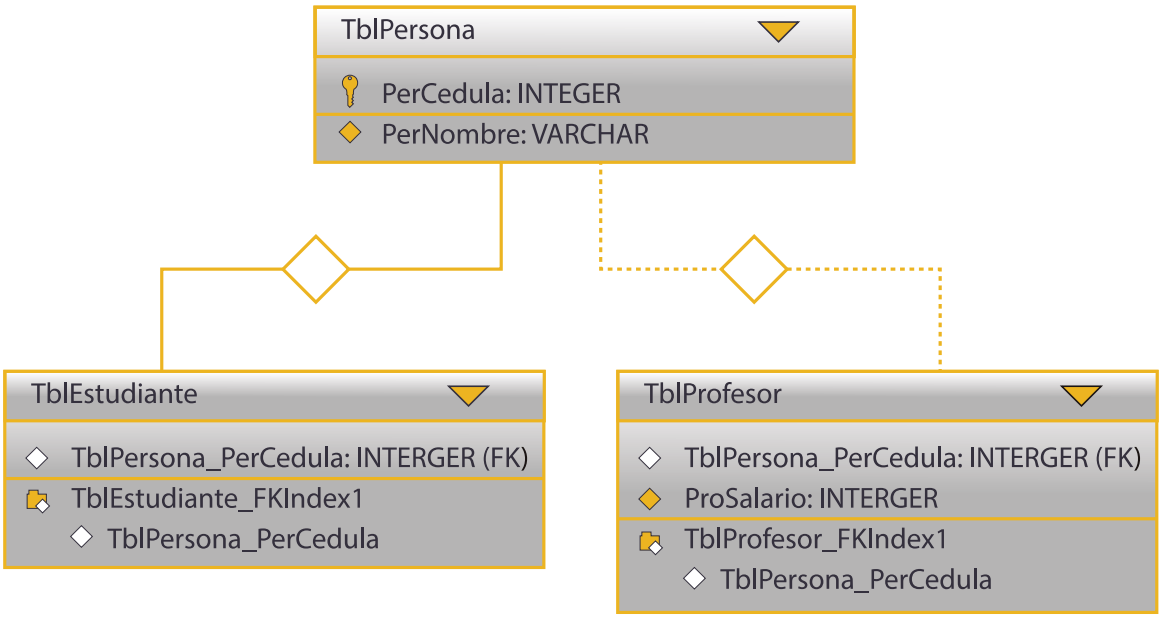


Figura 17. Transformación a modelo relacional

3.2 Relación recursiva

Las relaciones recursivas se tratan de la misma forma que las otras, sólo que hay que imaginar que la tabla se divide en dos, una por cada rol. Teniendo en cuenta eso, la solución es idéntica a lo ya resuelto en los casos anteriores.

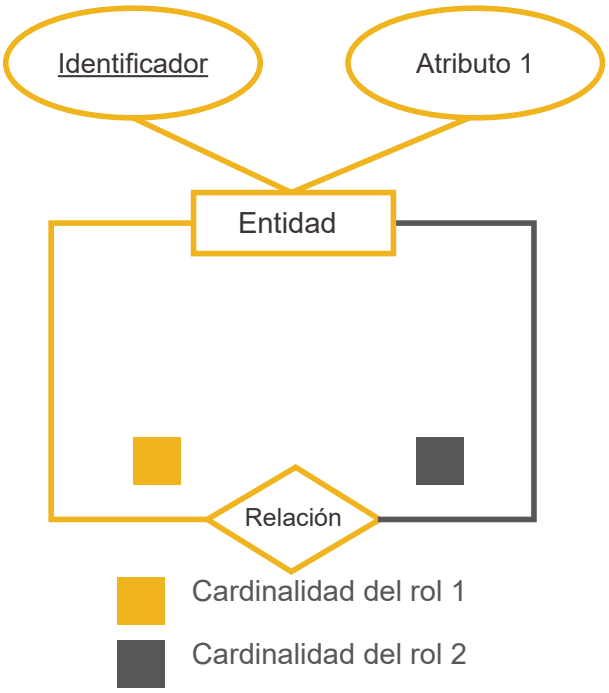
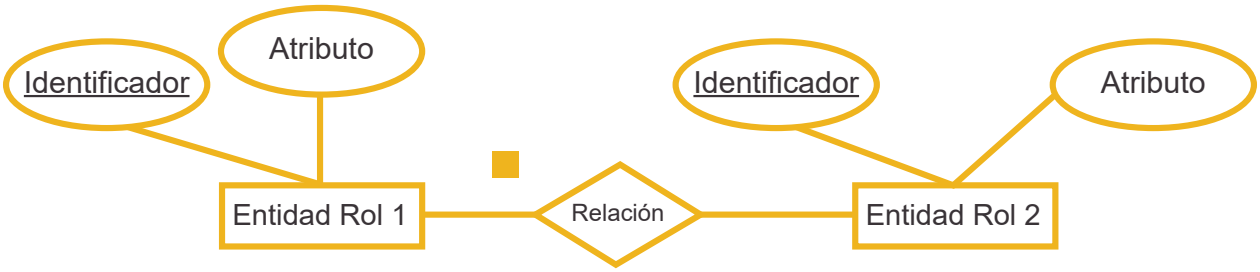


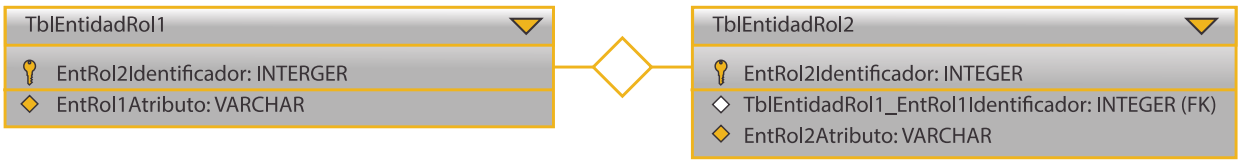
Figura 18. Relación recursiva

La forma de imaginarse las relaciones recursivas para resolverlas es:



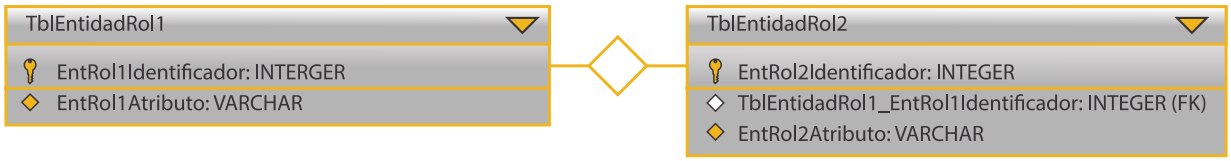
Donde se “duplica la entidad” y se conservan las cardinalidades, luego dependiendo de ésta, se aplica la regla de transformación correspondiente teniendo las siguientes posibilidades:

1 caso: si la cardinalidad de la relación es de 1 a 1.

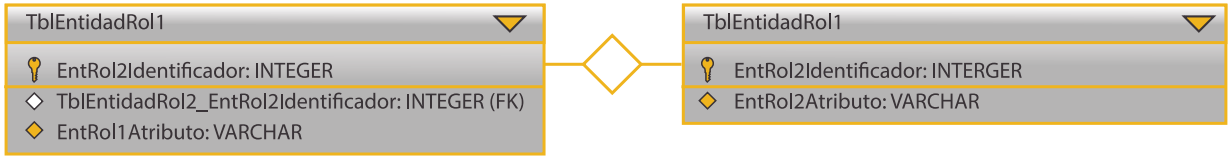


La entidad con Rol débil hereda el atributo de identificación de la otra entidad y pasa a ser su llave foránea

2 caso: si la cardinalidad de la relación es de 1 a M:

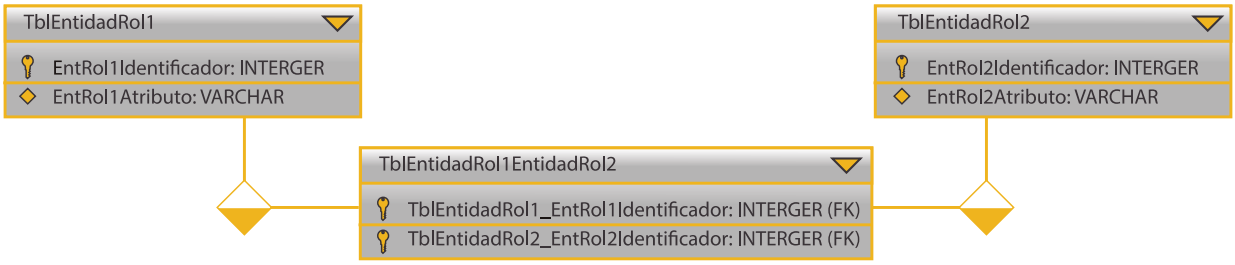


O de M a 1



En cualquiera de los casos, la entidad con relación muchos hereda el atributo de identificación de la otra y esta se convierte en la llave foránea

3 caso: si la cardinalidad de la relación es de M a M:



Se genera en este caso una tercera tabla que relaciona las dos entidades, está compuesta por los atributos de identificación que hereda de cada entidad, más los atributos propios de la relación (si existieran).

4. Normalizacion

Una vez finalizado el proceso de conversión, se pueden presentar ciertos inconvenientes tales como:

- Incapacidad de almacenar ciertos hechos.
- Redundancias y por tanto posibilidad de inconsistencias.
- Ambigüedades.
- Perdida de información.
- Perdida de ciertas restricciones de integridad que dan lugar a interdependencia de los datos.
- Existencia de valores nulos.
- Aparición de estados inválidos en los proceso de manipulación.

La normalización es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más pequeñas, que además de ser más simples y más estables, son más fáciles de mantener. También se puede entender la normalización como una serie de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar un esquema que minimice los problemas de lógica. Cada regla está basada en la que le antecede.

La normalización también hace las cosas fáciles de entender. Los seres humanos tenemos la tendencia de simplificar las cosas al máximo. Lo hacemos con casi todo, desde los animales hasta con los automóviles. Vemos una imagen de gran tamaño y la hacemos más simple agrupando cosas similares juntas. Las guías que la normalización provee crean el marco de referencia para simplificar una estructura de datos compleja.

Otra ventaja de la normalización de base de datos es el consumo de espacio. Una base de datos normalizada ocupa menos espacio en disco que una no normalizada. Hay menos repetición de datos, lo que tiene como consecuencia un mucho menor uso de espacio en disco.



Una vez obtenido el esquema relacional resultante del esquema entidad/ relación que representa la base de datos, normalmente se tendrá una buena base de datos. Pero otras veces, debido a fallos en el diseño o a problemas indetectables, se tendrá un esquema que puede producir una base de datos que incorpore estos problemas:

Redundancia: se llama así a los datos que se repiten continua e innecesariamente por las tablas de las bases de datos. Cuando es excesiva es evidente que el diseño hay que revisarlo, es el primer síntoma de problemas y se detecta fácilmente.

Ambigüedades: datos que no clarifican suficientemente el registro al que representan. Los datos de cada registro podrían referirse a más de un registro o incluso puede ser imposible saber a qué ejemplar exactamente se están refiriendo. Es un problema muy grave y difícil de detectar.

Perdida restricciones de integridad: normalmente debido a dependencias funcionales. Más adelante se explica este problema. Se arreglan fácilmente siguiendo una serie de pasos concretos.

Anomalías en operaciones de modificación de datos: el hecho de que al insertar un solo elemento haya que repetir tuplas en una tabla para variar unos pocos datos. O que eliminar un elemento suponga eliminar varias tuplas necesariamente (por ejemplo que eliminar un cliente suponga borrar seis o siete filas de la tabla de clientes, sería un error muy grave y por lo tanto un diseño terrible).

El principio fundamental reside en que las tablas deben referirse a objetos o situaciones muy concretas, relacionados exactamente con elementos reconocibles por el sistema de información de forma inequívoca. Cada fila de una tabla representa inequívocamente un elemento reconocible en el sistema. Lo que ocurre es que conceptualmente es difícil agrupar esos elementos correctamente.

En cualquier caso la mayor parte de problemas se agravan si no se sigue un modelo conceptual y se decide crear directamente el esquema relacional. En ese caso, el diseño tiene una garantía casi asegurada de funcionar mal.

Cuando aparecen los problemas enumerados, entonces se les puede resolver usando reglas de normalización. Estas reglas suelen forzar la división de una tabla en dos o más tablas para arreglar ese problema.

4.1 Grado de normalización

Las formas normales corresponden a una teoría de normalización iniciada por el propio Coddy continuada por otros autores (entre los que destacan Boycey Fagin). Codd definió en 1970 la primera forma normal, desde ese momento aparecieron la segunda, tercera, la Boyce-Codd, la cuarta y la quinta forma normal.

Una tabla puede encontrarse en primera forma normal y no en segunda forma normal, pero no al contrario. Es decir los números altos de formas normales son más restrictivos (la quinta forma normal cumple todas las anteriores).

4.1.1 Primera forma normal 1N

Es una forma normal inherente al esquema relacional. Es decir toda tabla realmente relacional la cumple.

Se dice que una tabla se encuentra en primera forma normal si impide que un atributo de una tupla pueda tomar más de un valor. La tabla:

EMPLEADO		
identificación	nombre	departamento
12345	Pedro Murcia	Sistemas
24680	Maria Dolores Rosas	Contabilidad
12232	Juan José Muñoz	Gerencia Recursos Humanos

Lo que se ve, es una tabla, pero no una tabla relacional (lo que en terminología de bases de datos relacionales se llama relación). No cumple la primera forma normal.

Se podría mejorar para que cumpla la primera forma normal así:

EMPLEADO		
identificación	nombre	departamento
12345	Pedro Murcia	Sistemas
24680	María Dolores Rosas	Contabilidad
12232	Juan José Muñoz	Gerencia
12232	Juan José Muñoz	Recursos Humanos

Asignación de competencias:

4.1.2 Segunda forma normal 2N

Si una tabla está en primera forma normal y además cada atributo que no sea clave, depende de forma funcional completa respecto de cualquiera de las claves. Toda la clave principal debe hacer dependientes al resto de atributos, si hay atributos que depende sólo de parte de la clave, entonces esa parte de la clave y esos atributos formarán otra tabla.

Para asignar una competencia a un instructor es necesario que esté legalmente contratado por el centro o que tenga resolución de nombramiento y según su especialidad se le asignan las competencias que va a orientar.

Alumno				
identificacion	codCurso	nombre	apellido	nota
11	34	Sofia	Loren	5
11	25	Sofía	Loren	4
12	34	Marcos	Pinto	3.5
13	25	Mireya	Ruano	4.5
13	34	Mireya	Ruano	5

Suponiendo que la identificación y el código del curso formen una clave principal para esta tabla, sólo la nota tiene dependencia funcional completa. El nombre y el apellido dependen de forma completa de la identificación. La tabla no es 2FN, para convertirla se debe convertir esa tabla en dos tablas así:

Alumno		
identificacion	nombre	apellido
11	Sofia	Loren
11	Sofía	Loren
12	Marcos	Pinto
13	Mireya	Ruano
13	Mireya	Ruano
NotasCurso		
identificacion	codCurso	nota
11	34	4
11	25	4
12	34	3.5
13	25	4.5
13	34	5

4.1.3 Tercera forma normal 3N

Ocurre cuando una tabla está en 2FN y además ningún atributo que no sea clave depende transitivamente de las claves de la tabla. Es decir no ocurre cuando algún atributo depende funcionalmente de atributos que no son clave.

Ejemplo:

ALUMNOS				
identificación	nombre	apellido	codCiudad	ciudad
31	Pedro	Picapiedra	100	Cali
32	Mónica	Galindo	100	Cali
33	Amparo	Grisales	101	Popayán
34	Roberto	Gómez	102	Neiva
35	Blanca	Moreno	102	Neiva

En la tabla anterior la ciudad depende funcionalmente del código de la ciudad, lo que hace que no esté en 3FN. La manera de solucionar el problema sería dividiendo la tabla en dos así:

ALUMNOS			
identificación	nombre	apellido	codCiudad
31	Pedro	Picapiedra	100
32	Mónica	Galindo	100
33	Amparo	Grisales	101

En la primera tabla creamos la tabla Alumnos con los atributos propios de él y solo se agrega el atributo de codCiudad para saber a qué ciudad pertenece.

ALUMNOS

CIUDAD	
codCiudad	Ciudad
100	Cali
100	Cali
101	Popayán
102	Neiva
102	Neiva

La segunda tabla corresponde a la ciudad con sus atributos.

4.1.4 Forma normal Boyce-Cood(FNBC o BCFN)

Ocurre si una tabla está en tercera forma normal y además todo determinante es una clave candidata. Ejemplo:

EMPRESA		
Empleado	Departamento	Responsable
Pedro	Dirección	Juan
Mari	Ventas	Luis
Jose	Mercadeo	Erika
Rosa	Ventas	Roció
Blanca	Gerencia	Andrés
Julian	Ventas	Mireya

La cuestión es que un empleado puede trabajar en varios departamentos. En cada departamento hay varios responsables, pero cada empleado sólo tiene asignado uno. El detalle importante que no se ha tenido en cuenta, es que el o la responsable sólo puede ser responsable en un departamento. Este detalle último produce una dependencia funcional ya que:

Por lo tanto se encuentra un determinante que no es clave candidata. No está por tanto en FNBC. En este caso la redundancia ocurre por mala selección de clave. La redundancia del departamento es completamente evitable. La solución sería:

PERSONAL	
Empleado	Responsable
Pedro	Juan
Mari	Luis
Jose	Erika
Rosa	Roció
Blanca	Andrés
Julian	Mireya

RESPONSABLES	
Responsable	Departamento
Juan	Dirección
Luis	Ventas
Erika	Mercadeo
Roció	Ventas
Andrés	Gerencia
Mireya	Ventas

En las formas de Boyce-Codd hay que tener cuidado al descomponer ya que se podría perder información por una mala descomposición.

4.1.5 Cuarta forma normal (4FN) dependencias multievaluadas.

Para el resto de formas normales (las diseñadas por Fagin, mucho más complejas), es importante definir este tipo de dependencia, que es distinta de las funcionales. Si las funcionales eran la base de la segunda y tercera forma normal (y de la de Boyce-Codd), éstas son la base de la cuarta forma normal.

Una dependencia multivaluada de X sobre Y (es decir X->>Y), siendo X e Y atributos de la misma tabla, ocurre cuando Y tiene un conjunto de valores bien definidos sobre cualquier valor de X. Es decir, dado X sabremos los posibles valores que puede tomar Y.

Se refiere a posibles valores (en plural) y se trata de que los valores de ese atributo siempre son los mismos según el valor de un atributo y no del otro.

Ejemplo:

Código Curso	Instructor	Material
11	Marta	1
11	Marta	2
11	Jaime	1
11	Jaime	2
12	Marta	1
12	Marta	2
12	Marta	3

La tabla cursos, instructores y materiales del curso. La tabla está en FNBC ya que no hay dependencias transitivas y todos los atributos son clave sin dependencia funcional hacia ellos. Sin embargo hay redundancia. Los materiales se van a repetir para cualquier instructor dando cualquier curso, ya que los instructores van a utilizar todos los materiales del curso (de no ser así no habría ninguna redundancia).

Los materiales del curso dependen de forma multivaluada del curso y no del instructor en una dependencia multivaluada (no hay dependencia funcional ya que los posibles valores son varios). Para el Código curso e Instructor podemos saber los materiales; pero se sabe por el curso y no por el instructor.

Cuarta forma normal

Ocurre esta forma normal cuando una tabla está en forma normal de BoyceCodd y toda dependencia multivaluada no trivial es una dependencia funcional. Son triviales aquellas dependencias multivaluadas en las que el conjunto formado por el determinante y el implicado no forman la clave primaria de la tabla y además el implicado no forma parte del determinante: es decir si X->>Y y además Y ¢ X y X,Y no es la clave de la tabla, tenemos una dependencia multivaluada no trivial (como ocurre en el ejemplo anterior).

Codigo Curso	Material
11	1
11	2
12	1
12	2
12	3

Codigo Curso	Instructor
11	Marta
11	Jaime
12	Marta

4.1.6 Quinta forma normal (5FN) dependencia de join o de reunión

Una proyección de una tabla es la tabla resultante de tomar un subconjunto de los atributos de una. Es decir una tabla formada por unas cuantas columnas de la tabla original.

La operación JOIN procedente también del álgebra relacional, consiste en formar una tabla con la unión de dos tablas. La tabla resultante estará formada por la combinación de todas las columnas y filas de ambas, excepto las columnas y filas repetidas.

Se dice que se tiene una tabla con dependencia de unión (o de tipo JOIN) si se puede obtener esa tabla como resultado de combinar mediante la operación JOIN varias proyecciones de la misma.

Quinta forma normal o forma normal de proyección-unión: ocurre cuando una tabla está en 4FN y cada dependencia de unión (JOIN) en ella es implicada por las claves candidatas.

Esta forma normal es la más compleja y polémica de todas. Es polémica porque no es claro que al pasar a 5FN mejore la base de datos, premisa definida por **Fagin**. Es raro encontrarse este tipo de problemas cuando la normalización llega a 4FN. Se deben a restricciones semánticas especiales aplicadas sobre la tabla.

Ejemplo:

Proveedor	Material	Proyecto
1	1	2
1	2	1
2	1	1
1	1	1

Indican códigos de material suministrado por un proveedor y utilizado en un determinado proyecto. Así vista la tabla, no permite ninguna proyección en la que no perdamos datos. Pero si ocurre una restricción especial como por ejemplo: cuando un proveedor nos ha suministrado alguna vez un determinado material, si ese material aparece en otro proyecto, haremos que el proveedor anterior nos suministre también ese material para el proyecto.

Eso ocurre en los datos como el proveedor número 1 nos suministró el material número 1 para el proyecto 2 y en el proyecto 1 utilizamos el material 1, aparecerá la tupla proveedor 1, material 1 y proyecto 1. Si un nuevo proyecto necesitará el material 1, entonces habrá que pedirlo a los proveedores 1 y 2 (ya que en otros proyectos los hemos utilizado).

La dependencia de reunión que produce esta restricción es muy difícil de ver ya que es lejana. Para esa restricción esta proyección de tablas sería válida:

Proveedor	Material
1	1
1	2
2	1

Proveedor	Proyecto
1	2
1	1
2	1

Material	Proyecto
1	2
2	1
1	1

Esa descomposición no pierde valores en este caso, sabiendo que si el proveedor nos suministra un material podremos relacionarle con todos los proyectos que utilizan ese material. Una tabla no está en quinta forma normal si hay una descomposición de esa tabla que muestre la misma información que la original y esa descomposición no tenga como clave la clave original de la tabla.

Forma Normal de Dominio Clave (FNDC):

Se la conoce más con sus siglas en inglés DKNF. Se trata de una forma normal enunciada también por Faginen 1981 al darse cuenta de los problemas de redundancia que ocurrían con algunos dominios. En este caso no se basó en dependencias entre los datos, sino que se basó en restricciones de dominio y restricciones de clave.

Restricciones de dominio: se trata de la restricción que hace que un determinado atributo obtenga sólo ciertos valores, los que estén de acuerdo a la definición de un determinado dominio.

Restricción de clave: es la restricción que permite que un atributo o un conjunto de atributos forme una clave candidata.

Fagin dice que una tabla está en FNDC si toda restricción sobre la tabla es consecuencia lógica de aplicar las restricciones de dominio y clave sobre la misma. Fagin demostró que si esto ocurría la tabla incluso estaba en. Ejemplo:

Alumno	Nivel Logrado	Nota
Rosa	Muy alto y eficiente	9
María	Muy alto aunque sin trabajo constante	9
Juan	Alto con trabajo habitual	7
José	Medio aunque con trabajo	5

Observando los datos de la tabla se observa que cuando la nota es igual o superior a 9, en el nivel aparece la palabra alto, cuando es un 7 o un 8 medio, y un 5 o 6 sería medio. Es decir tenemos restricciones que no son ni de dominio ni de clave en esa tabla. Lo lógico sería:

Alumno	Nivel logrado	Nota
Rosa	Eficiente	9
María	Trabajo constante	9
Juan	Trabajo habitual	7
José	Trabajo medio	5

Nivel académico	Nota mínima	Nota máxima
Rosa	9	10
María	7	8,99
Juan	5	6,99
José	0	4,99

No se pierde información al diseñar las tablas de esta forma y de hecho es más eficiente para la base de datos.

Glosario

Modelo Entidad Relación: modelo que consta de entidades, atributos y relaciones y permite modelar conceptualmente una base de datos relacional.

Modelo Entidad Relación Extendido: modelo que permite modelar nuevos tipos de relaciones, que no incluye el MER normal, tales como agregación, generalización, exclusión e inclusión.

Modelo Relacional: es un modelo que se obtiene de la tabla: es la representación de una entidad en un modelo relacional, la cual incluye atributos.

Campo: es la representación de un atributo en un modelo relacional el cual contiene su nombre, tipo de dato y descripción de si es clave primaria o llave foránea.

Clave Primaria: atributo único que no se repite, es identificado en un modelo relacional porque esta subrayado o tiene el estereotipo <pk> y en un esquema relacional se encuentra subrayado.

Clave Foránea: atributo que es pasado a una tabla, según la regla de transformación, se distingue en un modelo relacional porque tiene el estereotipo <fk> y en un esquema relacional porque despliega una flecha hacia la tabla de la cual proviene.

Relación: es una asociación que existe entre dos entidades, en un MER se distinguen tres tipos de relaciones: uno a uno (1..1), uno a muchos (1..M) y muchos a muchos (M..M).

Esquema Relacional: expresar el modelo relacional por medio un esquema de palabras y asociaciones entre ellas (flechas) que representan las relaciones.

1FN: abreviatura de Primera Forma Normal. 2FN: abreviatura de Segunda Forma Normal.

3FN: abreviatura de Tercera Forma Normal.

4FN: abreviatura de Cuarta Forma Normal. 5FN: abreviatura de Quinta Forma Normal.

FNBC: abreviatura de Froma Normal de BoyceCood.

Bibliografía

Korth, H. (2006). *Fundamentos de Bases de Datos*. McGraw-Hill.

Sánchez, J. (2004). *Principios sobre bases de datos*. Recuperado de http://www.jorgesanchez.net/bd/bdrelacional.pdf

Silberschatz, A., Korth, H. (2004). *Fundamentos de Bases de Datos*. Madrid: McGraw-Hil.

Control del documento

CONSTRUCCIÓN OBJETO DE APRENDIZAJE

SENA

CONVERSIÓN MER A MODELO RELACIONAL	
Centro Industrial de Mantenimiento Integral - CIMI Regional Santander	
Líder línea de producción:	Santiago Lozada Garcés
Asesores pedagógicos:	Rosa Elvia Quintero Guasca Claudia Milena Hernández Naranjo
Líder expertos temáticos:	Rita Rubiela Rincón Badillo
Experto temático:	Nelson Mauricio Silva Maldonado -V1
Experto temático:	Edward Jose Beltran Lozano - V2
Diseño multimedia:	Jesús Antonio Vecino Valero
Programador:	Francisco José Lizcano Reyes
Producción de audio:	Victor Hugo Tabares Carreño

creative commons
BY NC SA

Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de la licencia que el trabajo original.

Términos de Referencia



ESTRUCTURA DE CONTENIDOS

Pág.

Términos de referencia.....3
1. Ejemplo de un documento de términos de referencia aplicados a un desarrollo de un sistema de información.....4
1.1. Objetivos.....4
1.2. Fuente de recursos.....4
1.3. Proponentes habilitados o restricciones.....4
1.4. Idioma.....4
1.5. Alcance de las actividades a realizar.....5
1.6. Entregables, hitos o fechas de entrega.....7
1.7. Equipo de trabajo.....7
1.8. Duración del contrato.....7
1.9. Sede de las labores.....7
1.10. Documentos a presentar.....7
1.11. Proceso de selección del proveedor.....8
1.12. Forma de pago.....8
1.13. Confidencialidad de la información 8
1.14. Soporte post-implementación.....8
Glosario.....9
Bibliografía.....10
Control del documento.....11

TÉRMINOS DE REFERENCIA

Se define al documento que confecciona una compañía para contratar servicios externos que permitan mejorar uno o varios procesos internos, como, por ejemplo, sistemas de producción, de administración, de sistemas informáticos, de finanzas, entre otros.

Una vez la compañía en sus procesos de mejoramiento continuo, detecta que tiene debilidades u oportunidades de mejora en alguna de sus áreas, o en su defecto desea mejorar o aprovechar oportunidades que no pueden realizarse con los recursos internos entonces se acude a buscar asistencia externa.

El documento que origina esa necesidad de asistencia externa se denomina **"Términos de referencia"**.

Componentes de un documento de términos de referencia:

- Objetivos.
- Fuente de recursos.
- Proponentes habilitados o restricciones.
- Idioma.
- Actividades a realizar.
- Entregables, hitos o fechas de entrega y detalle de los entregables.
- Equipo de trabajo.
- Duración del contrato.
- Sede o localización de las labores a realizar.
- Documentos a presentar.
- Proceso para seleccionar al proveedor.
- Forma de pago.
- Confidencialidad de la información.
- Soporte Post-Implementación.



1. Ejemplo de un documento de términos de referencia aplicados a un desarrollo de un sistema de información

1.1 Objetivo

Contratar los servicios de una empresa especializada en desarrollo de software a la medida para realizar el análisis, diseño, desarrollo, pruebas e implementación del módulo de Logística y Administración de Bodegas para la compañía XYZ.

Dentro de los entregables están:



- Código fuente.
- Licencia de uso.
- Manuales técnicos y de usuario final.
- Capacitación técnica y de usuario final.
- Herramientas de desarrollo (manejador de bases de datos, herramientas de generación de reportes, herramienta generadora de código fuente y las que se necesiten para el mantenimiento del sistema).
- Soporte post-implementación.

1.2 Fuente de recursos

Los valores monetarios destinados para esta labor serán los contemplados en el presupuesto del área de logística destinados para el año en curso.

1.3 Proponentes habilitados o restricciones

Los proponentes habilitados serán aquellas personas jurídicas, con experiencia en desarrollo de sistemas informáticos que tengan más de 5 años en el mercado demostrables. Se debe de anexar casos de éxito y referencias empresariales.

1.4 Idioma

El sistema de información deberá estar implementado en idioma español.

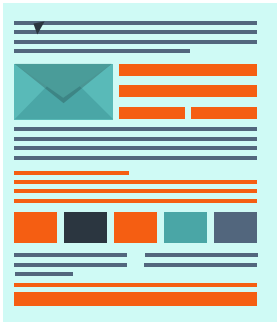
1.5 Alcance de las actividades a realizar

Comprende el diseño, desarrollo, pruebas, implementación y puesta en marcha de un sistema de información de acuerdo a las siguientes especificaciones técnicas:

- **Seguridad:** los usuarios deberán autenticarse y ser identificados consultando la base de datos usuarios registrados. Las claves de acceso de cada usuario deberán guardarse en texto encriptado utilizando algoritmo RSA o MD5. Los usuarios deben de clasificarse por roles, perfiles y permisos.

  El sistema de información debe de incluir certificados digitales (SSL) bajo protocolo https para darle mayor seguridad a nivel transaccional y debe de proveer opción de auditoría para operaciones claves del sistema.

- **Bases de datos:** el sistema de información deberá ser compatible con un gestor de base de datos relaciones tipo PostgreSQL, Microsoft SQL Server u Oracle. La base de datos deberá estar normalizada, se debe de disponer de su modelo relacional y se debe de contemplar copias de seguridad diferenciales semanales y completas mensuales.



- **Control transaccional:** el sistema de información debe de generar alertas automáticas para eventos críticos del sistema, por ejemplo, campos obligatorios, valores de stock mínimos o cualquier otro tipo de control que pueda interferir en la integridad de la información a procesar.



- **Disponibilidad:** debe de ser continua con un nivel de servicio para los usuarios 24/7 y ante fallas de disponibilidad de la solución en cualquiera de sus componentes, se debe generar alarmas.

- **Interoperatibilidad:** el sistema de información debe de tener capacidad de interactuar con otros sistemas a través de archivos en formato XML o Web Services (Servicios Web) para el intercambio de información con los otros sistemas transaccionales existentes.



- **Accesibilidad:** para tener acceso al sistema de información se podrá realizar vía web (desde cualquier desde cualquier parte del mundo), utilizando navegadores web como Google Chrome y mediante cualquier dispositivo electrónico que cuente con acceso a Internet. La interface de usuario será en entorno Web la cual deberá ser intuitiva y amigable.

- **Parametrizable:** el sistema de información deberá de contener módulo que permita parametrizar o administrar las tablas básicas del sistema (tablas maestras). Por ejemplo, Maestros de Clientes, Usuarios, Bodegas, Tasas de cambio, Proveedores, entre otros. Las opciones a proveer son Adición, Modificación, Cambio de estado Habilitado / No habilitado (no eliminación) e Impresión. No se debe de permitir la eliminación de registros debido a que no se debe de perder la trazabilidad de las operaciones.



- **Colaboración:** el sistema de información podrá generar y enviar alertas a los usuarios respectivos sobre estados críticos que son parametrizables. Estas alertas son en correos electrónicos, mensajes de texto (sms) o vía interface web.

- **Multimoneda:** el sistema de información debe de permitir el uso de las siguientes monedas USD dólar y COP pesos colombianos y el manejo de las conversiones entre estas según la tasa de cambio legal vigente.



- **Informes:** el sistema de información deberá proporcionar al usuario autorizado reportes que contengan los filtros predefinidos o que se permita generar consultas para extraer información en formato PDF y con exportación a Excel.

- **Arquitectura y escalabilidad:** el sistema de información deberá estar diseñado con conceptos de programación orientada a objetos, desarrollado en un lenguaje de última generación y en capas de software (capa de presentación, capa de lógica de negocios, capa de datos) que permitan una mayor escalabilidad del sistema como un mejor mantenimiento del mismo. Se debe de contemplar en el dimensionamiento de la base de datos un crecimiento del 10% de los usuarios actuales.

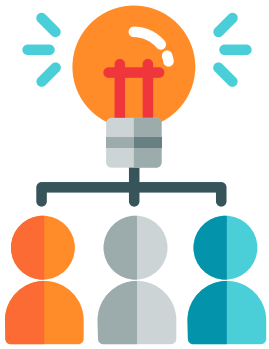


- **Desempeño:** el sistema de información deberá ofrecer un tiempo de respuesta óptimo en operaciones transaccionales y presentar un excelente desempeño ante una alta demanda de usuarios concurrentes.

1.6 Entregables, hitos o fechas de entrega

El proveedor debe de relacionar un cronograma de trabajo, donde se especifiquen los entregables según las especificaciones del sistema de información y las fechas de entrega de los mismos.

1.7 Equipo de trabajo



El proveedor debe de relacionar las personas involucradas en el desarrollo, especificando su experiencia en sus hojas de vida y debe de asignar un Coordinador o Gerente de Proyecto quien es la persona responsable de su ejecución e interlocutora entre el proveedor y la compañía.

1.8 Duración del contrato

Se debe de relacionar el tiempo máximo de duración del contrato. Por ejemplo, la duración total que incluye las fases de análisis, diseño, desarrollo, pruebas e implementación será de diez meses calendario a partir de la firma del contrato y debe ser finalizado en la fecha prevista. La finalización del contrato debe estar sujeta a la aprobación de los entregables con base en los términos de referencia estipulados.

1.9 Sede de las labores

Se debe de especificar el lugar de trabajo donde se va a implementar el sistema de información.

1.10 Documentos a presentar



Se debe de especificar el tiempo máximo de entrega de los documentos. Por ejemplo, el proveedor debe de entregar propuesta en un máximo de 15 días, donde se especifique la oferta económica, respuesta a cada una de las especificaciones del sistema, cronograma de trabajo y recursos humanos involucrados. La moneda en la que se debe de presentar la

propuesta es en COP pesos colombianos y debe de incluir los honorarios, viáticos y todos los gastos para el desarrollo de la propuesta. El contrato final se realizará con precio fijo.

1.11 Proceso de selección del proveedor



Se debe de especificar el número mínimo de propuestas a recibir (por ejemplo, mínimo 3 propuestas). Se procede a realizar una calificación de cada una según lo requerido en los términos de condiciones. Para esta calificación se provee de una plantilla en Excel con los criterios de evaluación.

Se debe de especificar la duración de la evaluación de las propuestas y la forma de notificación al proveedor seleccionado. Por ejemplo, la compañía se reserva un máximo de un mes para la evaluación y calificación de las propuestas recibidas y se notificará de manera formal mediante comunicación escrita al proveedor seleccionado.

1.12 Forma de pago

Los pagos al proveedor seleccionado se realizarán según los entregables realizados, en común acuerdo y especificados en el contrato.

1.13 Confidencialidad de la información

Los proveedores deberán de abstenerse de distribuir información de los términos de referencia a terceros. Igualmente, la información referente a la evaluación de las propuestas no se dará a conocer hasta que se realice la evaluación y se haya notificado la adjudicación al proveedor seleccionado.

1.14 Soporte post-implementación

En caso de que existan errores en el producto entregado, luego de la entrega, el proveedor solucionará el o los errores sin que la compañía incurra en gastos adicionales.



GLOSARIO

Mejoramiento continuo: basado en procesos de calidad, el mejoramiento continuo analiza y busca oportunidades de mejora en los procesos organizacionales, con el objeto de proveer mejores productos o servicios y así obtener mayor satisfacción de los clientes.

SSL: "Secure Sockets Layer": es un protocolo informático diseñado para darle mayor seguridad al envío y recepción de datos utilizando claves de cifrado.

BIBLIOGRAFÍA

Guevara, G. (2000). *Habilidades Básicas [Paráfrasis], Manuscrito no Publicado.* México: Facultad de Filosofía, U. V.

Lledó, P. (2016). *Director de Proyectos: Cómo aprobar el examen PMP.*

Pressman, R. (2010). *Ingeniería del software, un enfoque práctico (Séptima edición).* Bogotá: McGraw-Hill.

Project Management Institute (2013). *Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK) Quinta Edición.* Project Management Institute, Inc.

CONTROL DEL DOCUMENTO

CONSTRUCCIÓN OBJETO DE APRENDIZAJE

TÉRMINOS DE REFERENCIA	
Centro Industrial de Mantenimiento Integral - CIMI Regional Santander	
Líder línea de producción:	Santiago Lozada Garcés
Asesores pedagógicos:	Rosa Elvia Quintero Guasca Claudia Milena Hernández Naranjo
Líder expertos temáticos:	Rita Rubiela Rincón Badillo
Experto temático:	Edgar Eduardo Vega Arango
Diseño multimedia:	Catalina Martínez Ávila

creative commons

BY NC SA

Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de la licencia que el trabajo original.