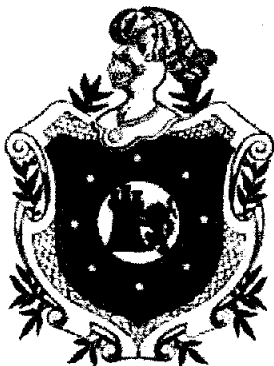


RECINTO UNIVERSITARIO "RUBÉN DARÍO"
UNAN MANAGUA



TEMA DE INVESTIGACION:
**Sistema de gestión de proyectos de construcción para la
cooperativa de obras civiles COOCI, en el año 2015.**

(SIMPLAEP)

AUTORES:

BR. IVAN RAFAEL LUNA GONZALEZ
BR. RAQUEL DEL CARMEN LOPEZ LEIVA
BR. GABRIEL ANTONIO ALLEX OBANDO

*José P
2015
Lun
2015*

TUTOR:

Msc. Juan de Dios Bonilla Anduray

LUGAR Y FECHA:

MANAGUA, 02 de Julio de 2015

Escuela Central "Salomón de la Selva"
UNAN-Managua
Impreso: 2015/07/02
Vencido: 2015/07/02
ECSHE



Índice

I. Resumen.....	5
II. Introducción.....	6
III. Antecedentes.....	7
IV. Planteamiento del problema.....	8
V. Justificación.....	10
VI. Objetivos.....	11
6.1. Objetivo general.....	11
6.2. Objetivos específicos.....	11
VII. Marco de Referencia.....	12
7.1. Marco Teórico.....	12
7.1.1. Sistema de la información (SI).....	12
7.1.1.1. Concepto de un sistema de información.....	12
7.1.1.2. Elementos de un sistema de información.....	12
7.1.1.3. Actividades básicas del sistema de información.....	13
7.1.1.4. Ciclo de vida de los sistemas de la información.....	14
7.1.1.5. Clasificación, según el entorno de aplicación.....	15
7.1.2. Aplicación Web.....	16
7.1.2.1. Conceptos.....	16
7.1.2.2. Interfaz.....	17
7.1.2.3. Estructura de las aplicaciones web.....	17
7.1.2.4. Ventajas.....	17
7.1.3. Base de datos.....	18
7.1.3.1. Conceptos.....	18
7.1.3.2. Modelos de bases de datos.....	19
7.1.3.3. Modelos de Datos.....	23
7.1.3.4. Clasificación de los modelos de datos.....	23
7.1.4. Normalización de bases de datos.....	24
7.1.4.1. Gestor de Bases de Datos Relacionales.....	25
7.1.4.2. Dependencia.....	25
7.1.4.3. Propiedades deducidas.....	27

7.1.4.4.	Formas Normales	28
7.1.4.5.	Reglas de Codd	31
7.1.5.	Motor de Base de Datos.....	34
7.1.5.1.	Conceptos.....	34
7.1.5.2.	Tareas del motor de base de datos.....	34
7.1.5.3.	Algunos motores de base de datos	35
7.1.6.	Lenguaje de Programación.....	35
7.1.6.1.	VB.net.....	35
7.1.6.2.	PHP.....	37
7.1.6.3.	Java	38
7.1.6.4.	SQL.....	40
7.1.7.	Herramientas de desarrollo.....	42
7.1.7.1.	Visual Studio 2010.....	42
7.1.7.2.	NetBeans	42
7.1.7.3.	WampServer.....	43
7.1.7.4.	Eclipse	44
7.1.8.	Patrones de desarrollos.....	45
7.1.8.1.	Modelo Vista Controlador (MVC)	45
7.1.9.	Codeigniter	49
7.1.10.	WORDPRESS.....	50
7.1.10.1.	Introducción	50
7.1.10.2.	Acceder en modo Editor	50
7.1.10.3.	Explorando el escritorio de wordpress.com	51
7.1.10.4.	Administración de Entradas	52
7.1.10.5.	Administración de Categorías	58
7.1.10.6.	Administración de Etiquetas de las entradas	61
7.1.10.7.	Administración de contenido multimedia	64
7.1.10.8.	Administración de Enlaces.....	67
7.1.10.9.	Administración de Páginas.....	70
7.1.10.10.	Administración de Comentarios	75
7.1.10.11.	Administración del Perfil	76
7.2.	La Empresa	78

7.2.1.	Misión	78
7.2.2.	Visión.....	78
7.2.3.	Sector a que se dirige.....	79
7.3.	Marco Conceptual	80
7.3.1.	Sistema informático:.....	80
7.3.2.	Aplicación web:.....	80
7.3.3.	Aplicación informática:.....	80
7.3.4.	Sistema Operativo:.....	81
7.3.5.	Lenguajes de Programación:.....	81
7.3.6.	Utilidad (informática):.....	81
7.3.7.	Middleware:.....	81
7.3.8.	Framework:.....	81
7.3.9.	Estándares:.....	82
7.3.10.	Optimizar:.....	82
VIII.	Marco metodológico.....	83
1.1.	Tipo de investigación	83
1.2.	Área de estudio.....	83
1.3.	Universo y Muestra	83
1.3.1.	Universo	83
8.3.2.	Ubicación geográfica.....	83
8.3.3.	Muestra	84
8.4.	Método de recolección de datos	84
8.5.	Hipótesis	84
8.6.	Operacionalización de las variables	85
8.7.	Cronograma de trabajo.....	86
8.8.	Estudio de Factibilidad.....	86
8.8.2.	Estudio de Factibilidad.....	86
IX.	Metodología a implementar	88
9.2	¿Qué es un modelo para el desarrollo de software?.....	88
9.3	¿Qué es un modelo espiral?.....	89
9.4	Características del modelo en espiral para el desarrollo de software.....	90
9.5	Ventajas del modelo espiral.....	91

9.6	Desventajas del modelo espiral	91
X.	Marco de Desarrollo.....	92
10.1.	Ficha requerimientos.....	92
10.1.2.	Ficha de requerimientos Funcionales	92
10.1.2.	Ficha requerimientos no funcionales:.....	94
10.2.	Casos de usos	95
10.2.1.	Diagramas de Casos de uso	95
10.3.	Diagrama de Clase.....	101
10.5.	Base de Datos.....	109
10.6.	Interfaz	110
XI.	Conclusiones.....	121
XII.	Recomendaciones.....	122
XIII.	Anexos	123
13.1.	Formato de planificación de proyectos	123
13.2.	Ejemplo de formato de planificación de proyectos.....	124
XIV.	Bibliografía.....	130

I. Resumen

Los lenguajes de programación como Visual Basic .net y su entorno de trabajo permiten crear aplicaciones de escritorio y sitios web, así como servicios para usuarios en el web. El primer apartado del informe actual corresponde a la introducción de la investigación donde se brinda una idea general del tema a profundizar en el estudio.

Los antecedentes de investigación mostraran al lector un detalle de los estudios previos a nuestro informe.

La MIPYME COOCI, Cooperativa de Obras Civiles encargada del área de construcción, debe realizar la planificación de proyectos a los clientes que soliciten su servicio, actualmente lo lleva de forma manual, esta gran cantidad de datos provoca problemas a la cooperativa situación explicada en detalle en el apartado planteamiento del problema de este documento.

La solución planteada para la institución por nuestra parte es el desarrollo de un Sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015, que resuelva los problemas actuales al crear un diseño eficaz y sencillo, como se puede observar en nuestros objetivos.

En cuanto a teoría relacionada al tema de estudio presentamos un apartado que reúne los conceptos más relevantes y pilares básicos para la solución final del problema. La metodología que seguimos en este proceso investigativo se describe detalladamente en el apartado correspondiente a material y método.

El análisis de resultados, muestra los diferentes diagramas producto de la aplicación de las etapas de la metodología de desarrollo de sistemas seleccionada para el presente estudio.

Finalmente, presentamos las conclusiones de nuestra investigación que demuestran la síntesis del proceso y análisis de los resultados obtenidos. Cabe destacar que presentamos recomendaciones para la aplicación a futuro, así como la bibliografía consultada para soporte de nuestra investigación.

II. Introducción

La Empresa COOCI que se encuentra en la ciudad de Managua, departamento de Nicaragua, es una empresa dedicada al servicio de creación de obras civiles, las cuales son muy necesarias para ciertas compañías que requieran este servicio.

Esta empresa realiza un proceso de planificación de proyectos para mantener control sobre las solicitudes de sus clientes y así poder evaluar el desempeño del personal a cargo de la construcción, también le permita tomar algunas decisiones en el momento preciso y no sufrir pérdidas, razón por la cual se ve la necesidad de implantar un sistema de control, planificación y administración de proyecto para mantener esta información sin que se pierda o sufra daño la misma.

Esta aplicación de escritorio (Cliente-Servidor) podrá brindar una facilidad en la administración y planificación de proyectos para las diferentes obras de construcción. Además contará con una interfaz eficaz para el manejo y manipulación de los datos requeridos por el sistema.

Poseerá una pantalla de seguridad para resguardar el acceso a los datos solamente a usuarios registrados y también a los administradores del sistema que podrán gozar de privilegios en las opciones definidas para su rol, una vez dentro del sistema, se podrá realizar las operaciones cotidianas usando la interfaz amigable para el usuario que este brinda.

Mediante esta aplicación se podrá organizar, calcular el presupuesto de un proyecto y podrá dar seguimiento a cada fase de este, Además de mantener un control diario del presupuesto.

III. Antecedentes

En Nicaragua, existen varias MIPYMES que se encuentran en el sector de la construcción, una de ellas es COOCI, la cual se encarga de controlar, administrar, planificar proyectos de obras civiles.

En el mercado informático existen dos herramientas para planificar proyectos, dentro del software privativo el exponente en este caso es Microsoft Project, por su parte en código abierto se encuentra Office Project.

Sin embargo, estas herramientas no cubren las expectativas de COOCI para la gestión integral de los proyectos que desarrollan, por tal razón es necesario el desarrollo de una aplicación personalizada que permita poder administrar y dar un seguimiento a todos los proyectos con la visión de optimizar los recursos de este.

IV. Planteamiento del problema

La Cooperativa de Obras Civiles (COOCI) es una empresa de servicios de construcción en el ámbito público y privado ubicada en: Managua Residencial Los Robles, del hotel Colón, 1 cuadra al Este, 3 cuadras al Sur, casa # 95. La empresa lleva más de 5 años de ofrecer sus servicios de construcción a la comunidad.

A pesar del tiempo que la empresa ha brindado sus servicios de construcción, posee poco personal ya que es una constructora mediana. La cantidad de empleados que la empresa maneja son obreros con contratos momentáneos de acuerdo a la necesidad que posee la empresa en la realización o participación de algún proyecto.

COOCI ha utilizado herramientas informática tales como Microsoft Project para la planificación de proyectos sin embargo esta herramienta no permite realizar actividades tales como:

- Control de recurso y personal en las actividades del proyecto.
- Realizar un análisis de riesgo para el proyecto que permita a la Empresa aprobar o rechazar dicha obra.
- Brindar un control exacto de los materiales que entran y salen del área de bodega.

Por la razones antes expuestas es que el área de dirección de proyectos ha decido implantar un sistema de planificación y ejecución de proyectos de obras civiles que si cumpla las necesidades de la empresa en el área de construcción de obras civiles que han sido solicitadas por el cliente. (Estas actividades que Project no cumple serán realizadas en el sistema de gestión de proyectos de construcción para la cooperativa de obras civiles (COOCI), en el año 2015)

La formulación del problema

Basado en la caracterización antes expuesta, el problema que enfrenta el área de dirección de Proyecto es que no existe un sistema para la planificación y ejecución de proyectos de construcción.

¿Cómo se podría automatizar la planificación y ejecución de proyectos de construcción en el área de planificación de proyecto, durante el año 2015?

La sistematización del problema

El tema antes delimitado se refiere a que necesita un sistema para la planificación y ejecución de proyectos de construcción durante el año 2015 a partir de la pregunta antes expuesta se considera relevante las siguientes preguntas de sistematización.

¿Cuál sería la mejor forma de automatizar el sistema para la planificación y ejecución de proyectos de construcción en el área de planificación de proyecto, durante el año 2015?

¿De qué manera se podría optimizar la gestión de la planificación y ejecución de proyectos de construcción en el área de planificación de proyecto, durante el año 2015?

V. Justificación

El sector construcción es un rubro importante en la economía nicaragüense, ya que brinda una gran cantidad de empleos temporales, la característica principal de este sector, es su versatilidad en el manejo de proyectos y una competencia agresiva.

Debido a lo anterior, la dirección de la MIPYME COOCI, entra en conciencia de la necesidad del control ágil de la información de los proyectos a su cargo, solicitando la creación de una solución automatizada que le brinde los informes adecuados y el registro de los datos relevantes.

Un sistema de gestión de proyectos automatizado le permitirá a la gerencia realizar el proceso de planificación, control y seguimiento en ejecución de los diferentes proyectos.

Este sistema además permitirá la realización de:

- Actualizar una actividad o asignación de recurso registrando los motivos del retraso.
- Evaluación del avance de un proyecto con respecto lo planeado.
- Brindar consultas básicas al usuario para el manejo del sistema.
- Dar de alta/baja un material.
- Controlar los pagos según el tipo de cargo que posea en la obra.
- Se establecerá consultas a la información de proyectos desde una aplicación web utilizando diferentes tecnologías como por ejemplo laptops.

VI. Objetivos

6.1. Objetivo general

Implantar un sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.

6.2. Objetivos específicos

Determinar los procesos principales del sistema multiplataforma de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.

Utilizar los marcos de trabajo gráfico como Visual Studio 2010, APTANA Studio y Neatbean para el diseño de las interfaces de usuario para el sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.

Aplicar el Modelo Vista Controlador (MVC) como patrón de desarrollo de las rutinas del sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.

VII. Marco de Referencia

7.1. Marco Teórico

7.1.1. Sistema de la información (SI)

7.1.1.1. Concepto de un sistema de información

Un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. En un sentido amplio, un sistema de información no necesariamente incluye equipo electrónico (hardware). Sin embargo, en la práctica se utiliza como sinónimo de sistema de información computarizado.

7.1.1.2. Elementos de un sistema de información

El equipo computacional: es decir, el hardware necesario para que el sistema de la información pueda operar lo constituyen las computadoras y los equipos periféricos que puedan conectarse a ella.

El recurso humano que interactuará con el sistema de información, el cual está formado por las personas que utilizan el sistema, alimentándolo con datos o utilizando los recursos que genere.

Los datos o información fuente que son introducidos en el sistema, son todas las entradas que éste necesita para generar como resultado la información que se desea.

Módulos del software del sistema de la información que hará que los datos de entrada introducidos sean procesados y que generen los resultados que se esperan.

La telecomunicación que es básicamente "hardware" y "software" facilitan la transmisión de texto, datos, imágenes y voz en forma electrónica.

Políticas y reglas de operación del sistema tanto en la parte funcional del proceso de negocio y los mecanismos para hacer trabajar una aplicación en la computadora.

7.1.1.3. Actividades básicas del sistema de información

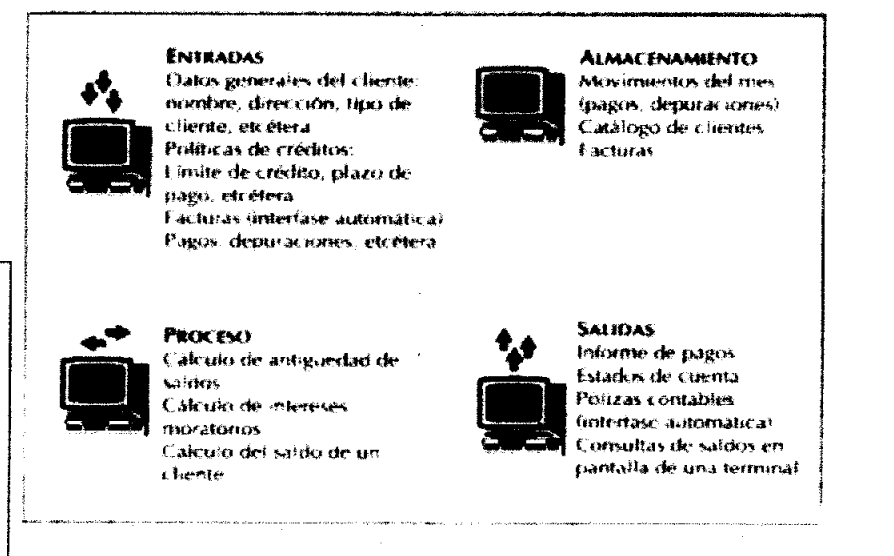
Entrada de la información es el proceso por el cual el SI toma los datos que requiere para procesar la información.

Almacenamiento de datos El almacenamiento es una de las actividades o capacidades más importante de la computadora, ya que a través de esta propiedad el sistema puede mostrar la información guardada en la sesión o proceso anterior. Existe muchas formas de almacenar los datos pero el más común es la implementación de bases de datos para el almacenamiento de los datos ingresado por el usuario.

Procesamiento de la información es la capacidad del SI de realizar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien que están almacenados.

Figura #1

Actividades básicas del sistema de información



7.1.1.4. Ciclo de vida de los sistemas de la información

Existen pautas básicas para el desarrollo de un SI para una organización:

- **Conocimiento de la Organización.** Analizar y conocer todos los sistemas que forman parte de la organización, así como los futuros usuarios del SI. En las empresas (fin de lucro presente), se analiza el proceso de negocio y los procesos transaccionales a los que dará soporte el SI.
- **Identificación de problemas y oportunidades.** El segundo paso es descubrir las situaciones que tiene la organización y de las cuales se puede sacar una ventaja competitiva (Por ejemplo: una empresa con un personal capacitado en manejo informático reduce el costo de capacitación de los usuarios), así como las situaciones desventajosas o limitaciones que hay que sortear o que tomar en cuenta (Por ejemplo: el edificio de una empresa que cuenta con un espacio muy reducido no permitirá instalar más de dos computadoras).
- **Determinar las necesidades.** Este proceso también se denomina elicitación o traspaso de información de los diferentes programas utilizados. En el mismo, se procede identificar a través de algún método de recolección de información (el que más se ajuste a cada caso) la información relevante para el SI que se propondrá.
- **Diagnóstico.** En este paso se elabora un informe resaltando los aspectos positivos y negativos de la organización. Este informe formará parte de la propuesta del SI y, también, será tomado en cuenta a la hora del diseño.
- **Propuesta.** Contando ya con toda la información necesaria acerca de la organización, es posible elaborar una propuesta formal dirigida hacia la organización donde se detalle: el presupuesto, la relación costo-beneficio y la presentación del proyecto de desarrollo del SI.

- **Diseño del sistema.** Una vez aprobado el proyecto, se comienza con la elaboración del diseño lógico del SI; la misma incluye: el diseño del flujo de la información dentro del sistema, los procesos que se realizarán dentro del sistema, el diccionario de datos, los reportes de salida, etc. En este paso es importante seleccionar la plataforma donde se apoyará el SI y el lenguaje de programación a utilizar.
- **Codificación.** Con el algoritmo ya diseñado, se procede a su reescritura en un lenguaje de programación establecido (programación) en la etapa anterior, es decir, en códigos que la máquina pueda interpretar y ejecutar.
- **Implementación.** Este paso consta de todas las actividades requeridas para la instalación de los equipos informáticos, redes y la instalación de la aplicación (programa) generada en la etapa de Codificación.
- **Mantenimiento.** Proceso de retroalimentación, a través del cual se puede solicitar la corrección, el mejoramiento o la adaptación del SI ya creado a otro entorno de trabajo o plataforma. Este paso incluye el soporte técnico acordado anteriormente.

7.1.1.5. Clasificación, según el entorno de aplicación

- **Sistema de procesamiento de transacciones:** Un sistema de procesamiento de transacciones (TPS) es un tipo de sistema de información que recolecta, almacena, modifica y recupera toda la información generada por las transacciones producidas en una organización.
- **Entorno transaccional:** Una transacción es un suceso o evento que crea/modifica los datos. El procesamiento de transacciones consiste en captar, manipular y almacenar los datos, y también, en la preparación de documentos; en el entorno transaccional, por tanto, lo importante es qué datos se modifican y cómo, una vez que ha terminado la transacción de los datos. Los TPS son los SI típicos que se pueden encontrar en este entorno.

- **Sistemas de soporte a decisiones (DSS):** Un DSS puede adoptar muchas formas diferentes. En general, podemos decir que un DSS es un sistema informático utilizado para servir de apoyo, más que automatizar, el proceso de toma de decisiones. Estos sistemas de apoyo son del tipo OLAP o de minería de datos, que proporcionan información y soporte para tomar una decisión.
- **Entorno decisional:** Este es el entorno en el que tiene lugar la toma de decisiones; en una empresa, las decisiones se toman a todos los niveles y en todas las áreas (otra cosa es si esas decisiones son estructuradas o no), por lo que todos los SI de la organización deben estar preparados para asistir en esta tarea, aunque típicamente, son los DSS los que se encargan de esta función. Si el único SI de una compañía preparado para ayudar a la toma de decisiones es el DSS, éste debe estar adaptado a todos los niveles jerárquicos de la empresa.

7.1.2. Aplicación Web

7.1.2.1. Conceptos

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

7.1.2.2. Interfaz

Las interfaces web tienen ciertas limitaciones en las funcionalidades que se ofrecen al usuario. Hay funcionalidades comunes en las aplicaciones de escritorio como dibujar en la pantalla o arrastrar-y-soltar que no están soportadas por las tecnologías web estándar. Los desarrolladores web generalmente utilizan lenguajes interpretados (scripts) en el lado del cliente para añadir más funcionalidades, especialmente para ofrecer una experiencia interactiva que no requiera recargar la página cada vez (lo que suele resultar molesto a los usuarios). Recientemente se han desarrollado tecnologías para coordinar estos lenguajes con las tecnologías en el lado del servidor, como ejemplo, AJAX es una técnica de desarrollo web que usa una combinación de varias tecnologías.

7.1.2.3. Estructura de las aplicaciones web

Aunque existen muchas variaciones posibles, una aplicación web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web ofrece la primera capa, y un motor capaz de usar alguna tecnología web dinámica, por ejemplo: PHP, Java Servlets o ASP, ASP.NET, CGI, ColdFunction embPerl, Python Ruby on Rails que constituye la capa intermedia. Por último, una base de datos constituye la tercera y última capa.

El navegador web manda peticiones a la capa intermedia que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario.

7.1.2.4. Ventajas

Ahorra tiempo: Se pueden realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa.

No hay problemas de compatibilidad: Basta tener un navegador actualizado para poder utilizarlas.

No ocupan espacio en nuestro disco duro.

Actualizaciones inmediatas: Como el software lo gestiona el propio desarrollador, cuando nos conectamos estamos usando siempre la última versión que haya lanzado.

Consumo de recursos bajo: Dado que toda (o gran parte) de la aplicación no se encuentra en nuestro ordenador, muchas de las tareas que realiza el software no consumen recursos nuestros porque se realizan desde otro ordenador.

Multiplataforma: Se pueden usar desde cualquier sistema operativo porque sólo es necesario tener un navegador.

Portables: Es independiente del ordenador donde se utilice (un PC de sobremesa, un portátil...) porque se accede a través de una página web (sólo es necesario disponer de acceso a Internet). La reciente tendencia al acceso a las aplicaciones web a través de teléfonos móviles requiere sin embargo un diseño específico de los ficheros CSS para no dificultar el acceso de estos usuarios.

La disponibilidad suele ser alta porque el servicio se ofrece desde múltiples localizaciones para asegurar la continuidad del mismo.

Los virus no dañan los datos porque éstos están guardados en el servidor de la aplicación.

Colaboración: Gracias a que el acceso al servicio se realiza desde una única ubicación es sencillo el acceso y compartición de datos por parte de varios usuarios. Tiene mucho sentido, por ejemplo, en aplicaciones online de calendarios u oficina.

Los navegadores ofrecen cada vez más y mejores funcionalidades para crear aplicaciones web ricas (RIAs).

7.1.3. Base de datos

7.1.3.1. Conceptos

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital (electrónico), y por ende se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos. (Wikipedia, wikipedia.org, 2012)

7.1.3.2. Modelos de bases de datos

Un modelo de datos es básicamente es una "descripción" de algo conocido como contenedor de datos (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general se refieren a algoritmos, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos:

Bases de datos jerárquicas

En este modelo los datos se organizan en forma de árbol invertido (algunos dicen raíz), en donde un nodo padre de información puede tener varios hijos. El nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se los conoce como hojas.

Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

Base de datos de red

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de nodo: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales.

Bases de datos transaccionales

Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.

Un ejemplo habitual de transacción es el traspaso de una cantidad de dinero entre cuentas bancarias. Normalmente se realiza mediante dos operaciones distintas, una en la que se decrementa el saldo de la cuenta origen y otra en la que se incrementa el saldo de la cuenta destino. Para garantizar la atomicidad del sistema (es decir, para que no aparezca o desaparezca dinero), las dos operaciones deben ser atómicas, es decir, el sistema debe garantizar que, bajo cualquier circunstancia (incluso una caída del sistema), el resultado final es que, o bien se han realizado las dos operaciones, o bien no se ha realizado ninguna.

Bases de datos relacionales

Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por registros (las filas de una tabla), que representarían las tuplas, y campos (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es, Structured Query Language (SQL) o Lenguaje Estructurado de

Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos.

Durante los años 80 la aparición de dBASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que dBase no utilizaba SQL como lenguaje base para su gestión.

Bases de datos multidimensionales

Son bases de datos ideadas para desarrollar aplicaciones muy concretas, como la creación de Cubos OLAP o bien lo que se conoce como Procesamiento analítico en línea que brinda una solución utilizada en el campo de la llamada Inteligencia empresarial (o Business Intelligence) cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos relacional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.

Bases de datos orientadas a objetos

Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los objetos completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

SQL: 2003, es el estándar de SQL92 ampliado, soporta los conceptos orientados a objetos y mantiene la compatibilidad con SQL92.

Bases de datos documentales

Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. Taurus es un sistema de índices optimizado para este tipo de bases de datos.

Bases de datos deductivas

Un sistema de base de datos deductiva, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos. Las bases de datos deductivas son también llamadas bases de datos lógicas, a raíz de que se basa en lógica matemática. Este tipo de base de datos surge debido a las limitaciones de la Base de Datos Relacional de responder a consultas recursivas y de deducir relaciones indirectas de los datos almacenados en la base de datos.

7.1.3.3. Modelos de Datos

Los modelos se utilizan en todo tipo de ciencias. Su finalidad es la de simbolizar una parte del mundo real de forma que sea más fácilmente manipulable. En definitiva es un esquema mental (conceptual) en el que se intentan reproducir las características de una realidad específica. (Jorge Sánchez, 2004)

En el caso de los modelos de datos, lo que intentan reproducir es una información real que deseamos almacenar en un sistema informático. Se denomina esquema a una descripción específica en términos de un modelo de datos. El conjunto de datos representados por el esquema forma la base de datos. (Jorge Sánchez, 2004)

7.1.3.4. Clasificación de los modelos de datos

Mundo real. Contiene la información tal cual la percibimos como seres humanos. Es el punto de partida (Jorge Sánchez, 2004)

Esquema conceptual. Representa el modelo de datos de forma independiente del DBMS que se utilizará. (Jorge Sánchez, 2004)

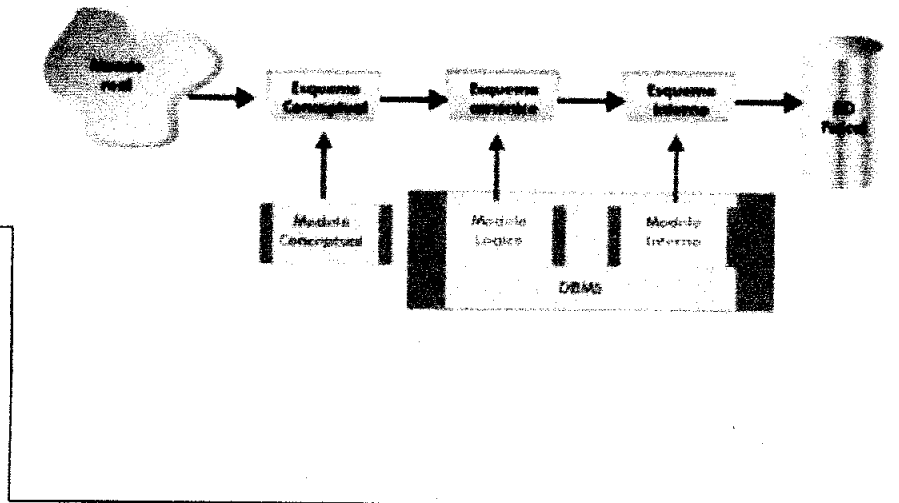
Esquema canónico (o de base de datos). Representa los datos en un formato más cercano al del ordenador. (Jorge Sánchez, 2004)

Esquema interno. Representa los datos según el modelo concreto de un sistema gestor de bases de datos (por ejemplo Oracle). (Jorge Sánchez, 2004)

Base de datos física. Los datos tal cual son almacenados en disco. (Jorge Sánchez, 2004)

Figura #2

Clasificación de los modelos de datos



7.1.4. Normalización de bases de datos

El proceso de normalización de bases de datos consiste en aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional.

Las bases de datos relacionales se normalizan para:

- Evitar la redundancia de los datos.
- Disminuir problemas de actualización de los datos en las tablas.
- Proteger la integridad de los datos.

En el modelo relacional es frecuente llamar tabla a una relación, aunque para que una tabla sea considerada como una relación tiene que cumplir con algunas restricciones:

- Cada tabla debe tener su nombre único.
- No puede haber dos filas iguales. No se permiten los duplicados.
- Todos los datos en una columna deben ser del mismo tipo.
- Terminología relacional equivalente

Relación = tabla o archivo.

Registro = registro, fila, renglón o tupla.

Atributo = columna o campo.

Clave = llave o código de identificación.

Clave Candidata = superclave mínima.

Clave Primaria = clave candidata elegida.

Clave Ajena (o foránea) = clave externa o clave foránea.

Clave Alternativa = clave secundaria.

Dependencia Multivaluada = dependencia multivalor.

RDBMS = Del inglés Relational Data Base Manager System que significa, Sistema

7.1.4.1. Gestor de Bases de Datos Relacionales.

1FN = Significa, Primera Forma Normal o 1NF del inglés First Normal Form.

Los términos Relación, Tupla y Atributo derivan del álgebra y cálculo relacional, que constituyen la fuente teórica del modelo de base de datos relacional.

Todo atributo en una tabla tiene un dominio, el cual representa el conjunto de valores que el mismo puede tomar. Una instancia de una tabla puede verse entonces como un subconjunto del producto cartesiano entre los dominios de los atributos. Sin embargo, suele haber algunas diferencias con la analogía matemática, ya que algunos RDBMS permiten filas duplicadas, entre otras cosas. Finalmente, una tupla puede razonarse matemáticamente como un elemento del producto cartesiano entre los dominios.

7.1.4.2. Dependencia

Dependencia funcional

Una dependencia funcional es una conexión entre uno o más atributos. Por ejemplo si se conoce el valor de DNI tiene una conexión con Apellido o Nombre.

Las dependencias funcionales del sistema se escriben utilizando una flecha, de la siguiente manera:

dataDeNacimiento → Edad

De la normalización (lógica) a la implementación (física o real) puede ser sugerible tener éstas dependencias funcionales para lograr la eficiencia en las tablas.

Propiedades de la Dependencia funcional

Dependencia funcional Reflexiva

Si "y" está incluido en "x" entonces $x \rightarrow y$

A partir de cualquier atributo o conjunto de atributos siempre puede deducirse él mismo. Si la dirección o el nombre de una persona están incluidos en el DNI, entonces con el DNI podemos determinar la dirección o su nombre.

Dependencia funcional Aumentativa

$x \rightarrow y$ entonces $xz \rightarrow yz$

DNI \rightarrow nombre

DNI, dirección \rightarrow nombre, dirección

Si con el DNI se determina el nombre de una persona, entonces con el DNI más la dirección también se determina el nombre y su dirección.

Dependencia funcional transitiva

Sean X, Y, Z tres atributos (o grupos de atributos) de la misma entidad. Si Y depende funcionalmente de X y Z de Y, pero X no depende funcionalmente de Y, se dice entonces que Z depende transitivamente de X. Simbólicamente sería:

$X \rightarrow Y \rightarrow Z$ entonces $X \rightarrow Z$

FechaDeNacimiento \rightarrow Edad

Edad \rightarrow Conducir

FechaDeNacimiento \rightarrow Edad \rightarrow Conducir

Entonces tenemos que FechaDeNacimiento determina a Edad y la Edad determina a Conducir, indirectamente podemos saber a través de FechaDeNacimiento a Conducir (En muchos países, una persona necesita ser mayor de cierta edad para poder conducir un automóvil, por eso se utiliza este ejemplo).

"C será un dato simple (dato no primario), B, será un otro dato simple (dato no primario), A, es la llave primaria (PK). Decimos que C dependerá de B y B dependerá funcionalmente de A."

7.1.4.3. Propiedades deducidas

Unión

$x \rightarrow y$ y $x \rightarrow z$ entonces $x \rightarrow yz$

Pseudo – transitiva

$x \rightarrow y$ y $wy \rightarrow z$ entonces $wx \rightarrow z$

Descomposición

$x \rightarrow y$ y z está incluido en y entonces $x \rightarrow z$

Claves

Una clave primaria es aquella columna (o conjunto de columnas) que identifica únicamente a una fila. La clave primaria es un identificador que va a ser siempre único para cada fila. Se acostumbra a poner la clave primaria como la primera columna de la tabla pero es más una conveniencia que una obligación. Muchas veces la clave primaria es numérica auto-incrementada, es decir, generada mediante una secuencia numérica incrementada automáticamente cada vez que se inserta una fila.

En una tabla puede que tengamos más de una columna que puede ser clave primaria por sí misma. En ese caso se puede escoger una para ser la clave primaria y las demás claves serán claves candidatas.

Una clave ajena (foreign key o clave foránea) es aquella columna que existiendo como dependiente en una tabla, es a su vez clave primaria en otra tabla.

Una clave alternativa es aquella clave candidata que no ha sido seleccionada como clave primaria, pero que también puede identificar de forma única a una fila dentro de una tabla. Ejemplo: Si en una tabla clientes definimos el número de documento (id_cliente) como clave primaria, el número de seguro social de ese cliente podría ser una clave alternativa. En este caso no se usó como clave primaria porque es posible que no se conozca ese dato en todos los clientes.

Una clave compuesta es una clave que está compuesta por más de una columna.

La visualización de todas las posibles claves candidatas en una tabla ayuda a su optimización. Por ejemplo, en una tabla PERSONA podemos identificar como claves su DNI, o el conjunto de su nombre, apellidos, fecha de nacimiento y dirección. Podemos usar cualquiera de las dos opciones o incluso todas a la vez como clave primaria, pero es mejor en la mayoría de sistemas la elección del menor número de columnas como clave primaria.

7.1.4.4. Formas Normales

Las formas normales son aplicadas a las tablas de una base de datos. Decir que una base de datos está en la forma normal N es decir que todas sus tablas están en la forma normal N.

En general, las primeras tres formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos. El creador de estas 3 primeras formas normales (o reglas) fue Edgar F. Codd.

7.1.4.4.1 Primera Forma Normal (1FN)

Una tabla está en Primera Forma Normal si:

- Todos los atributos son atómicos. Un atributo es atómico si los elementos del dominio son indivisibles, mínimos.
- La tabla contiene una clave primaria única.
- La clave primaria no contiene atributos nulos.
- No debe existir variación en el número de columnas.
- Los Campos no clave deben identificarse por la clave (Dependencia Funcional)
- Debe Existir una independencia del orden tanto de las filas como de las columnas, es decir, si los datos cambian de orden no deben cambiar sus significados
- Una tabla no puede tener múltiples valores en cada columna.
- Los datos son atómicos (a cada valor de X le pertenece un valor de Y y viceversa).
- Esta forma normal elimina los valores repetidos dentro de una BD

7.1.4.4.2 Segunda Forma Normal (2FN)

Dependencia Funcional. Una relación está en 2FN si está en 1FN y si los atributos que no forman parte de ninguna clave dependen de forma completa de la clave principal. Es decir que no existen dependencias parciales. (Todos los atributos que no son clave principal deben depender únicamente de la clave principal).

En otras palabras podríamos decir que la segunda forma normal está basada en el concepto de dependencia completamente funcional. Una dependencia funcional $X \rightarrow Y$ es completamente funcional si al eliminar los atributos A de X significa que la dependencia no es mantenida, esto es que $A \in X, X - \{A\} \not\rightarrow Y$. Una dependencia funcional $x \rightarrow y$ es una dependencia parcial si hay algunos atributos $A \in X$ que pueden ser eliminados de X y la dependencia todavía se mantiene, esto es $A \in X, X - \{A\} \rightarrow Y$.

Por ejemplo $\{\text{DNI}, \text{ID_PROYECTO}\} \rightarrow \text{HORAS_TRABAJO}$ (con el DNI de un empleado y el ID de un proyecto sabemos cuántas horas de trabajo por semana trabaja un empleado en dicho proyecto) es completamente funcional dado que ni $\text{DNI} \rightarrow \text{HORAS_TRABAJO}$ ni $\text{ID_PROYECTO} \rightarrow \text{HORAS_TRABAJO}$ mantienen la dependencia.

Sin embargo $\{\text{DNI}, \text{ID_PROYECTO}\} \rightarrow \text{NOMBRE_EMPLEADO}$ es parcialmente dependiente dado que $\text{DNI} \rightarrow \text{NOMBRE_EMPLEADO}$ mantiene la dependencia.

7.1.4.4.3 Tercera Forma Normal (3FN)

La tabla se encuentra en 3FN si es 2FN y si no existe ninguna dependencia funcional transitiva entre los atributos que no son clave.

Un ejemplo de este concepto sería que, una dependencia funcional $X \rightarrow Y$ en un esquema de relación R es una dependencia transitiva si hay un conjunto de atributos Z que no es un subconjunto de alguna clave de R, donde se mantiene $X \rightarrow Z$ y $Z \rightarrow Y$.

Por ejemplo, la dependencia $\text{SSN} \rightarrow \text{DMGRSSN}$ es una dependencia transitiva en EMP_DEPT de la siguiente figura. Decimos que la dependencia de DMGRSSN el atributo clave SSN es transitiva vía DNUMBER porque las dependencias $\text{SSN} \rightarrow \text{DNUMBER}$ y $\text{DNUMBER} \rightarrow \text{DMGRSSN}$ son mantenidas, y DNUMBER no es un subconjunto de la clave de EMP_DEPT. Intuitivamente, podemos ver que la dependencia de DMGRSSN sobre DNUMBER es indeseable en EMP_DEPT dado que DNUMBER no es una clave de EMP_DEPT.

Formalmente, un esquema de relación R está en 3 Forma Normal Elmasri-Navathe, si para toda dependencia funcional $X \rightarrow A$, se cumple al menos una de las siguientes condiciones:

X es superllave o clave.

A es atributo primo de R ; esto es, si es miembro de alguna clave en R .

Además el esquema debe cumplir necesariamente, con las condiciones de segunda forma normal.

7.1.4.4.4 Forma normal de Boyce-Codd (FNBC)

La tabla se encuentra en FNBC si cada determinante, atributo que determina completamente a otro, es clave candidata. Deberá registrarse de forma anillada ante la presencia de un intervalo seguido de una formalización perpetua, es decir las variantes creadas, en una tabla no se llegaran a mostrar, si las ya planificadas, dejan de existir.

Formalmente, un esquema de relación R está en FNBC, si y sólo si, para toda dependencia funcional $X \rightarrow A$ válida en R , se cumple que

X es superllave o clave.

De esta forma, todo esquema R que cumple FNBC, está además en 3FN; sin embargo, no todo esquema R que cumple con 3FN, está en FNBC.

7.1.4.4.5 Cuarta Forma Normal (4FN)

Una tabla se encuentra en 4FN si, y sólo si, para cada una de sus dependencias múltiples no funcionales $X \twoheadrightarrow Y$, siendo X una super-clave que, X es o una clave candidata o un conjunto de claves primarias.

7.1.4.4.6 Quinta Forma Normal (5FN)

Una tabla se encuentra en 5FN si:

La tabla está en 4FN

No existen relaciones de dependencias no triviales que no siguen los criterios de las claves. Una tabla que se encuentra en la 4FN se dice que está en la 5FN si, y sólo si, cada relación de dependencia se encuentra definida por claves candidatas.

7.1.4.5. Reglas de Codd

Codd se percató de que existían bases de datos en el mercado las cuales decían ser relacionales, pero lo único que hacían era guardar la información en las tablas, sin estar estas tablas literalmente normalizadas; entonces éste publicó 12 reglas que un verdadero sistema relacional debería tener, en la práctica algunas de ellas son difíciles de realizar. Un sistema podrá considerarse "más relacional" cuanto más siga estas reglas.

Regla No. 1 - La Regla de la información

Toda la información en un RDBMS está explícitamente representada de una sola manera por valores en una tabla.

Cualquier cosa que no exista en una tabla no existe del todo. Toda la información, incluyendo nombres de tablas, nombres de vistas, nombres de columnas, y los datos de las columnas deben estar almacenados en tablas dentro de las bases de datos. Las tablas que contienen tal información constituyen el Diccionario de Datos. Esto significa que todo tiene que estar almacenado en las tablas.

Toda la información en una base de datos relacional se representa explícitamente en el nivel lógico exactamente de una manera: con valores en tablas. Por tanto los metadatos (diccionario, catálogo) se representan exactamente igual que los datos de usuario. Y puede usarse el mismo lenguaje (ej. SQL) para acceder a los datos y a los metadatos (regla 4)

Regla No. 2 - La regla del acceso garantizado

Cada ítem de datos debe ser lógicamente accesible al ejecutar una búsqueda que combine el nombre de la tabla, su clave primaria, y el nombre de la columna.

Esto significa que dado un nombre de tabla, dado el valor de la clave primaria, y dado el nombre de la columna requerida, deberá encontrarse uno y solamente un valor. Por esta razón la definición de claves primarias para todas las tablas es prácticamente obligatoria.

Regla No. 3 - Tratamiento sistemático de los valores nulos

La información inaplicable o faltante puede ser representada a través de valores nulos

Un RDBMS (Sistema Gestor de Bases de Datos Relacionales) debe ser capaz de soportar el uso de valores nulos en el lugar de columnas cuyos valores sean desconocidos.

Regla No. 4 - La regla de la descripción de la base de datos

La descripción de la base de datos es almacenada de la misma manera que los datos ordinarios, esto es, en tablas y columnas, y debe ser accesible a los usuarios autorizados.

La información de tablas, vistas, permisos de acceso de usuarios autorizados, etc, debe ser almacenada exactamente de la misma manera: En tablas. Estas tablas deben ser accesibles igual que todas las tablas, a través de sentencias de SQL (o similar).

Regla No. 5 - La regla del sub-lenguaje Integral

Debe haber al menos un lenguaje que sea integral para soportar la definición de datos, manipulación de datos, definición de vistas, restricciones de integridad, y control de autorizaciones y transacciones.

Esto significa que debe haber por lo menos un lenguaje con una sintaxis bien definida que pueda ser usado para administrar completamente la base de datos.

Regla No. 6 - La regla de la actualización de vistas

Todas las vistas que son teóricamente actualizables, deben ser actualizables por el sistema mismo.

La mayoría de las RDBMS permiten actualizar vistas simples, pero deshabilitan los intentos de actualizar vistas complejas.

Regla No. 7 - La regla de insertar y actualizar

La capacidad de manejar una base de datos con operandos simples aplica no sólo para la recuperación o consulta de datos, sino también para la inserción, actualización y borrado de datos'.

Esto significa que las cláusulas para leer, escribir, eliminar y agregar registros (SELECT, UPDATE, DELETE e INSERT en SQL) deben estar disponibles y operables, independientemente del tipo de relaciones y restricciones que haya entre las tablas o no.

Regla No. 8 - La regla de independencia física

El acceso de usuarios a la base de datos a través de terminales o programas de aplicación, debe permanecer consistente lógicamente cuando quiera que haya cambios en los datos almacenados, o sean cambiados los métodos de acceso a los datos.

El comportamiento de los programas de aplicación y de la actividad de usuarios vía terminales debería ser predecible basados en la definición lógica de la base de datos, y éste comportamiento debería permanecer inalterado, independientemente de los cambios en la definición física de ésta.

Regla No. 9 - La regla de independencia lógica

Los programas de aplicación y las actividades de acceso por terminal deben permanecer lógicamente inalteradas cuando quiera que se hagan cambios (según los permisos asignados) en las tablas de la base de datos.

La independencia lógica de los datos especifica que los programas de aplicación y las actividades de terminal deben ser independientes de la estructura lógica, por lo tanto los cambios en la estructura lógica no deben alterar o modificar estos programas de aplicación.

Regla No. 10 - La regla de la independencia de la integridad

Todas las restricciones de integridad deben ser definibles en los datos, y almacenables en el catálogo, no en el programa de aplicación.

Las reglas de integridad

Ningún componente de una clave primaria puede tener valores en blanco o nulos (ésta es la norma básica de integridad).

Para cada valor de clave foránea deberá existir un valor de clave primaria concordante. La combinación de estas reglas aseguran que haya integridad referencial.

Regla No. 11 - La regla de la distribución

El sistema debe poseer un lenguaje de datos que pueda soportar que la base de datos esté distribuida físicamente en distintos lugares sin que esto afecte o altere a los programas de aplicación.

El soporte para bases de datos distribuidas significa que una colección arbitraria de relaciones, bases de datos corriendo en una mezcla de distintas máquinas y distintos sistemas operativos y que esté conectada por una variedad de redes, pueda funcionar como si estuviera disponible como en una única base de datos en una sola máquina.

Regla No. 12 - Regla de la no-subversión

Si el sistema tiene lenguajes de bajo nivel, estos lenguajes de ninguna manera pueden ser usados para violar la integridad de las reglas y restricciones expresadas en un lenguaje de alto nivel (como SQL).

Algunos productos solamente construyen una interfaz relacional para sus bases de datos No relacionales, lo que hace posible la subversión (violación) de las restricciones de integridad. Esto no debe ser permitido.

7.1.5. Motor de Base de Datos

7.1.5.1. Conceptos

El Motor de base de datos es el servicio principal para almacenar, procesar y proteger datos. Proporciona, además, acceso controlado y procesamiento de transacciones para cumplir con los requisitos de las aplicaciones.

7.1.5.2. Tareas del motor de base de datos

- Diseñar y crear una base de datos que contenga las tablas relacionales
- Implementar sistemas para obtener acceso y cambiar los datos almacenados en la base de datos
- Aplicar los sistemas implementados en la organización o en los clientes.
- Proporcionar soporte técnico para optimizar el rendimiento de la base de datos.

7.1.5.3. Algunos motores de base de datos

- Microsoft SQL Server.
- MYSQL Server
- Oracle Server
- SQL lite

Tecnología para el desarrollo del sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.

7.1.6. Lenguaje de Programación

7.1.6.1. VB.net

Visual Basic .Net es uno de los lenguajes de programación que más entusiasmo despiertan entre los programadores de PC, tanto expertos como novatos. En el caso de los programadores expertos por la facilidad con la que desarrollan aplicaciones complejas en poquísimo tiempo (comparado con lo que cuesta programar en Visual C++, por ejemplo). En el caso de los programadores novatos por el hecho de ver de lo que son capaces a los pocos minutos de empezar su aprendizaje.

El precio que hay que pagar por utilizar Visual Basic 6.0 es una menor velocidad o eficiencia en las aplicaciones por ejemplo. Visual Basic 6.0 es un lenguaje de programación visual, también llamado lenguaje de 4ª generación. Esto quiere decir que un gran número de tareas se realizan sin escribir código, simplemente con operaciones gráficas realizadas con el ratón sobre la pantalla.

Visual Basic .Net es también un programa basado en objetos, aunque no orientado a objetos como C++ o Java. La diferencia está en que Visual Basic .Net utiliza objetos con propiedades y métodos, pero carece de los mecanismos de herencia y polimorfismo propios de los verdaderos lenguajes orientados a objetos. Aunque la tendencia general es "meterlo en el mismo tarro" de los lenguajes orientados a objetos.

7.1.6.1.1 Ventajas

- Visual Basic es un lenguaje simple y por tanto fácil de aprender.
- Su mayor facilidad radica en el dibujado de formularios, mediante el arrastre de controles.
- La sintaxis está cercana al lenguaje humano.
- Las ramas de Visual Basic X ha muerto. Microsoft ha comenzado con su versión
- .NET, completamente diferente del Visual Basic original. Visual Basic .NET compila hacia el mismo código final que C#. Visual Basic .NET es mucho más potente y alberga muchas más funcionalidades que ésta versión.
- Es un lenguaje RAD, centrado en conseguir en el menor tiempo posible los resultados deseados, por eso mismo su mayor uso está en las pequeñas aplicaciones, como gestión de bares, empresas, restaurantes.
- Tiene una ligera implementación de Programación Orientada a Objetos.
- Permite el tratamiento de mensajes de Windows.
- Gran parte del trabajo en el diseño de formularios está realizado, gracias a la gran gama de controles incorporados junto al lenguaje que ahorran costes de tiempo de desarrollo.
- Permite crear controles personalizados fácilmente del mismo modo que el diseño de formularios.
- Permite generar librerías dinámicas (DLL).

7.1.6.1.1 Inconvenientes.

- Es propiedad de Microsoft, por tanto nadie que no sea del equipo de desarrollo de esta compañía decide la evolución del lenguaje.
- Sólo existe un compilador & IDE, llamado igual que el lenguaje.
- Sólo genera ejecutables para Windows.
- La sintaxis es bastante inflexible.
- Los ejecutables generados son relativamente lentos.
- NO es adecuado para aplicaciones grandes, multimedia, de oficina, videojuegos, editores gráficos.
- -NO permite características de programación avanzada.

7.1.6.2. PHP

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

7.1.6.1.1 Características de PHP

- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas, ejemplo que se hace evidente en el uso de php arrays.
- El código fuente escrito en PHP es invisible al navegador web y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos. Incluso aplicaciones como Zend framework, empresa que desarrolla PHP, están totalmente desarrolladas mediante esta metodología.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).

- Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.
- Debido a su flexibilidad ha tenido una gran acogida como lenguaje base para las aplicaciones WEB de manejo de contenido, y es su uso principal.

7.1.6.3. Java

Java es una tecnología que se usa para el desarrollo de aplicaciones que convierten a la Web en un elemento más interesante y útil.

7.1.6.3.1 Característica de java

Lenguaje Simple:

Se lo conoce como lenguaje simple porque viene de la misma estructura de c y c++; ya que c++ fue un referente para la creación de java por eso utiliza determinadas características de c++ y se han eliminado otras.

Orientado a Objeto:

Toda la programación en java en su mayoría está orientada a objeto, ya que al estar agrupados en estructuras en estructuras encapsuladas es más fácil su manipulación.

Distribuido:

Permite abrir sockets, establecer y aceptar conexiones con los servidores o clientes remotos; facilita la creación de aplicaciones distribuidas ya que proporciona una colección de clases para aplicaciones en red.

Robusto:

Es altamente fiable en comparación con c, se han eliminado muchas características con la aritmética de punteros, proporciona numerosas comprobaciones en compilación y en tiempo de ejecución.

Seguro:

La seguridad es una característica muy importante en java ya que se han implementado barreras de seguridad en el lenguaje y en el sistema de ejecución de tiempo real.

Indiferente a la arquitectura:

Java es compatible con los más variados entornos de red, cualquiera que sean estos desde Windows 95, Unix a Windows Nt y Mac, para poder trabajar con diferentes sistemas operativos.

Java es muy versátil ya que utiliza byte-codes que es un formato intermedio que sirve para transportar el código eficientemente o de diferentes plataformas (Hardware - Software).

Portable:

Por ser indiferente a la arquitectura sobre la cual está trabajando, esto hace que su portabilidad sea muy eficiente, sus programas son iguales en cualquiera de las plataformas, ya que java especifica tamaños básicos, esto se conoce como la máquina virtual de java.

Interpretado y compilado a la vez:

Java puede ser compilado e interpretado en tiempo real, ya que cuando se construye el código fuente este se transforma en una especie de código de máquina.

Multihebra o Multihilos:

Java tiene una facilidad de cumplir varias funciones al mismo tiempo, gracias a su función de Multihilos ya que por cada hilo que el programa tenga se ejecutaran en tiempo real muchas funciones al mismo tiempo.

Dinámico:

El lenguaje java es muy dinámico en la fase de enlazado, sus clases solamente actuaran en medida en que sean requeridas o necesitadas con esto permitirá que los enlaces se puedan incluir incluso desde fuentes muy variadas o desde la red.

Produce Applets:

En java se pueden crear aplicaciones independientes y applets. Independientes porque se pueden comportar como cualquier programa escrito en cualquier lenguaje. Por otra parte los applets considerados pequeños programas, tienen la capacidad de ejecutar funciones muy complejas.

Alto rendimiento

Java es considerado de alto rendimiento por ser tan veloz en el momento de correr los programas y por ahorrarse muchas líneas de código.

Lenguaje Simple:

Se lo conoce como lenguaje simple porque viene de la misma estructura de c y c++; ya que c++ fue un referente para la creación de java por eso utiliza determinadas características de c++ y se han eliminado otras.

Orientado a Objeto:

Toda la programación en java en su mayoría está orientada a objeto, ya que al estar agrupados en estructuras en estructuras encapsuladas es más fácil su manipulación.

7.1.6.4. SQL

7.1.6.4.1 Concepto

El lenguaje de consulta estructurado o SQL (por sus siglas en inglés Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en ellas. Una de sus características es el manejo del álgebra y el cálculo relacional que permiten efectuar consultas con el fin de recuperar de forma sencilla información de interés de bases de datos, así como hacer cambios en ellas.

Características generales del SQL

El SQL es un lenguaje de acceso a bases de datos que explota la flexibilidad y potencia de los sistemas relacionales y permite así gran variedad de operaciones.

Es un lenguaje declarativo de "alto nivel" o "de no procedimiento" que, gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros —y no a registros individuales— permite una alta productividad en codificación y la orientación a objetos. De esta forma, una sola sentencia puede equivaler a uno o más programas que se utilizarían en un lenguaje de bajo nivel orientado a registros.

7.1.6.4.1 Características de SQL:

Lenguaje de definición de datos: El LDD de SQL proporciona comandos para la definición de esquemas de relación, borrado de relaciones y modificaciones de los esquemas de relación.

Lenguaje interactivo de manipulación de datos: El LMD de SQL incluye lenguajes de consultas basado tanto en álgebra relacional como en cálculo relacional de tuplas.

Integridad: El LDD de SQL incluye comandos para especificar las restricciones de integridad que deben cumplir los datos almacenados en la base de datos.

Definición de vistas: El LDD incluye comandos para definir las vistas.

Control de transacciones: SQL tiene comandos para especificar el comienzo y el final de una transacción.

SQL incorporado y dinámico: Esto quiere decir que se pueden incorporar instrucciones de SQL en lenguajes de programación como: C++, C, Java, PHP, Cobol, Pascal y Fortran.

Autorización: El LDD incluye comandos para especificar los derechos de acceso a las relaciones y a las vistas.

7.1.6.4.2 Tipos de Datos

Algunos de los tipos de datos básicos de SQL son:

Date: una fecha de calendario que contiene el año (de cuatro cifras), el mes y el día.

Time: La hora del día en horas minutos segundos (el valor predeterminado es 0).

Timestamp: la combinación de Date y Time.

7.1.7. Herramientas de desarrollo.

7.1.7.1. Visual Studio 2010

7.1.7.3.1 Concepto

Visual Studio es un conjunto completo de herramientas de desarrollo para la generación de aplicaciones web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles. Visual Basic, Visual C# y Visual C++ utilizan todos el mismo entorno de desarrollo integrado (IDE), que habilita el uso compartido de herramientas y hace más sencilla la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes utilizan las funciones de .NET Framework, las cuales ofrecen acceso a tecnologías clave para simplificar el desarrollo de aplicaciones web ASP y Servicios Web XML.

7.1.7.1.3 Ventajas de Visual Studio 2010

La facilidad del lenguaje permite crear aplicaciones para Windows en muy poco tiempo. En otras palabras, permite un desarrollo eficaz y menor inversión tanto en tiempo como en dinero. Permite generar librerías dinámicas de forma activa, mediante una reconfiguración en su proceso de colección o codificación.

7.1.7.1.4 Características de Visual Studio 2010

Microsoft Visual Studio 2010 Professional incluye una variedad de características que permite a los desarrolladores crear, depurar, realizar pruebas de unidad e implementar aplicaciones de alta calidad en diferentes plataformas, desde Windows, Internet, Cloud, Office y SharePoint, entre otros muchos.

7.1.7.2. NetBeans

7.1.7.3.1 Concepto

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos.

7.1.7.3. WampServer

7.1.7.3.1 Concepto

WampServer es un entorno de desarrollo web para Windows en el cual se podrán crear aplicaciones web con Apache, PHP y base de datos en MySQL (motor de base de datos). Esta herramienta incluye además con un administrador de base de datos PHP MyAdmin con el cual podremos crear una nueva base de datos e ingresar la data de las tablas creadas en ella, realizar consultas y generar scripts SQL, como exportar e importar scripts de base de datos. WampServer ofrece a los desarrolladores herramientas necesarias para realizar aplicaciones web de manera local, con un sistema operativo (Windows), un manejador de base de datos (MySQL), un software de programación script web PHP. WampServer se caracteriza por que puede ser usado de forma libre es decir no debemos de contar con alguna licencia el cual nos permita el uso de la misma, ya que pertenece a la corriente de "open source".

7.1.7.3.3 Características

Provee a los desarrolladores con los cuatro elementos necesarios para un servidor web: un Sistema Operativo (Windows), un manejador de base de datos (MySQL), un software para servidor web (Apache) y un software de programación script Web (PHP (generalmente), Python o PERL), debiendo su nombre a dichas herramientas. Lo mejor de todo es que WAMP5 es completamente gratuito. WAMP incluye, además de las últimas versiones de Apache, PHP y MySQL, versiones anteriores de las mismas, para el caso de que se quiera testear en un entorno de desarrollo particular.

7.1.7.3.3 Utilidad

El uso de WAMP permite servir páginas HTML a Internet, además de poder gestionar datos en las. al mismo tiempo WAMP, proporciona lenguajes de programación para desarrollar aplicaciones Web.

7.1.7.4. Eclipse

7.1.7.4.1 Concepto

Eclipse es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent o Azureus.

7.1.7.4.2 Características

Eclipse dispone de un Editor de texto con resaltado de sintaxis. La compilación es en tiempo real. Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes (wizards) para creación de proyectos, clases, tests, etc., y refactorización.

Asimismo, a través de "plugins" libremente disponibles es posible añadir control de versiones con Subversion.4 e integración con Hibernate.

7.1.7.4.3 Lenguajes de programación utilizados en Eclipse 3.2.1

Java	1.911.693	92,66%
ANSI C	133.263	6,46%
C++	10.082	0,49%
JSP	3.613	0,18%
SH	2.066	0,10%
PERL	1.468	0,07%
PHP	896	0,04%
SED	2	0,00%

7.1.8. Patrones de desarrollos

7.1.8.1. Modelo Vista Controlador (MVC)

7.1.8.1.2 Concepto

Es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento.

7.1.8.1.3 Descripción del patrón

De manera genérica, los componentes de MVC se podrían definir como sigue:

El Modelo: Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada (típicamente a un usuario). Las peticiones de acceso o manipulación de información llegan al 'modelo' a través del 'controlador'.

El Controlador: Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos). También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta de 'modelo' (por ejemplo, desplazamiento o scroll por un documento o por los diferentes registros de una base de datos), por tanto se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo' (véase Middleware).

La Vista: Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.

7.1.8.1.4 Interacción de los componentes

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

El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace, etc.)

El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.

El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario (por ejemplo, el controlador actualiza el carro de la compra del usuario). Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.

El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se reflejan los cambios en el modelo (por ejemplo, produce un listado del contenido del carro de la compra). El modelo no debe tener conocimiento directo sobre la vista. Sin embargo, se podría utilizar el patrón Observador para proveer cierta indirección entre el modelo y la vista, permitiendo al modelo notificar a los interesados de cualquier cambio. Un objeto vista puede registrarse con el modelo y esperar a los cambios, pero aun así el modelo en sí mismo sigue sin saber nada de la vista. Este uso del patrón Observador no es posible en las aplicaciones Web puesto que las clases de la vista están desconectadas del modelo y del controlador. En general el controlador no pasa objetos de dominio (el modelo) a la vista aunque puede dar la orden a la vista para que se actualice.

Nota: En algunas implementaciones la vista no tiene acceso directo al modelo, dejando que el controlador envíe los datos del modelo a la vista. Por ejemplo en el MVC usado por Apple en su framework Cocoa. Suele citarse como Modelo-Interface-Control, una variación del MVC más puro

La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente...

7.1.8.1.5 Uso en aplicaciones Web

Aunque originalmente MVC fue desarrollado para aplicaciones de escritorio, ha sido ampliamente adaptado como arquitectura para diseñar e implementar aplicaciones web en los principales lenguajes de programación. Se han desarrollado multitud de frameworks, comerciales y no comerciales, que implementan este patrón (ver apartado siguiente "Frameworks MVC"); estos frameworks se diferencian básicamente en la interpretación de como las funciones MVC se dividen entre cliente y servidor.

Los primeros frameworks MVC para desarrollo web planteaban un enfoque de cliente ligero en el que casi todas las funciones, tanto de la vista, el modelo y el controlador recaían en el servidor. En este enfoque, el cliente manda la petición de cualquier hiperenlace o formulario al controlador y después recibe de la vista una página completa y actualizada (u otro documento); tanto el modelo como el controlador (y buena parte de la vista) están completamente alojados en el servidor. Como las tecnologías web han madurado, ahora existen frameworks como JavaScriptMVC, Backbone o jQuery14 que permiten que ciertos componentes MVC se ejecuten parcial o totalmente en el cliente (véase AJAX).

(wikipedia, 2014)

7.1.8.1.6 Frameworks de desarrollo de aplicaciones

.Net

.NET Framework es un entorno para construir, instalar y ejecutar servicios Web y otras aplicaciones. Se compone de tres partes principales:

- El Common Language Runtime
- Las clases Framework
- ASP.NET.

El .NET Framework es un entorno multilenguaje para la construcción, distribución y ejecución de Servicios Webs y aplicaciones." "El .NET Framework es una nueva plataforma diseñada para simplificar el desarrollo de aplicaciones en el entorno distribuido de Internet." "El .NET Framework consta de dos componentes principales: el Common Language Runtime y la librería de clases .NET Framework.

7.1.8.1.6.1 Características de implementación de .NET Framework

.NET Framework proporciona algunas características básicas que facilitan la implementación de una amplia gama de aplicaciones. Entre estas características se incluyen:

Aplicaciones carentes de impacto.

Esta característica permite aislar la aplicación y eliminar conflictos de archivos DLL. De forma predeterminada, los componentes no afectan a otras aplicaciones

Componentes privados predeterminados.

De forma predeterminada, los componentes se implementan en el directorio de la aplicación y sólo son visibles para la aplicación en la que están incluidos.

Uso compartido de código controlado.

Para compartir código es necesario hacer que éste quede disponible para compartir explícitamente, mediante la modificación del comportamiento predeterminado.

Control de versiones simultáneas.

Es posible que coexistan varias versiones de un componente o de una aplicación; el usuario puede elegir las versiones que desea utilizar, y Common Language Runtime impone la directiva de control de versiones.

Implementación y duplicación mediante XCOPY.

Los componentes y aplicaciones autodescriptivos e independientes pueden implementarse sin entradas del Registro o dependencias.

Actualizaciones inmediatas.

Los administradores pueden utilizar servidores host, como ASP.NET, para actualizar programas de archivos DLL, incluso en equipos remotos.

Integración con Microsoft Windows Installer.

A la hora de implementar la aplicación, estarán disponibles las características de anuncio, edición, reparación e instalación a petición.

Implementación de empresa.

Esta característica proporciona una distribución de software sencilla, que incluye el uso de Active Directory.

Descarga y almacenamiento en caché.

El incremento de descargas reduce el tamaño de las mismas, y los componentes pueden aislarse a fin de que sólo los utilice la aplicación para una implementación de impacto cero.

Código que no es de plena confianza.

La identidad se basa en el código en lugar de basarse en el usuario, el administrador establece la directiva y no aparecen cuadros de dialogo de certificado

7.1.9. Codeigniter

Codeigniter es un framework para desarrollo de aplicaciones – un conjunto de herramientas – para gente que construye sitios web usando PHP.

Su objetivo es permitirle desarrollar proyectos mucho más rápido que lo que podría hacer si escribiera el código desde cero, proveyéndole un rico conjunto de bibliotecas para tareas comunes, así como y una interfaz sencilla y una estructura lógica para acceder a esas bibliotecas.

Codeigniter le permite enfocarse creativamente en su proyecto al minimizar la cantidad de código necesaria para una tarea dada. La arquitectura de Codeigniter se basa en el paradigma MVC (modelo, vista, controlador), lo que permite una ágil separación de las distintas capas que configuran la funcionalidad de un software.

(Codeigniter, 2012)

7.1.10. WORDPRESS

7.1.10.1. Introducción

WordPress es una avanzada plataforma semántica de publicación personal orientada a la estética, los estándares web y la usabilidad. WordPress es el sistema que utilizas cuando deseas trabajar con tu herramienta de publicación en lugar de pelearte con ella.

7.1.10.2. Acceder en modo Editor

Se os ha entregado un usuario, que permite administrar la web. Este usuario permite escribir nuevas entradas y secciones, y modificarlos pudiendo publicarlos directamente.

Para llegar al portal se entra en `http://[nombre-del-dominio]/wp-admin`. Al entrar veréis una pantalla en la que solicita usuario y contraseña, que deberéis rellenar con los facilitados al entregaros este tutorial:

WordPress



Usuario

Contraseña

Acceder

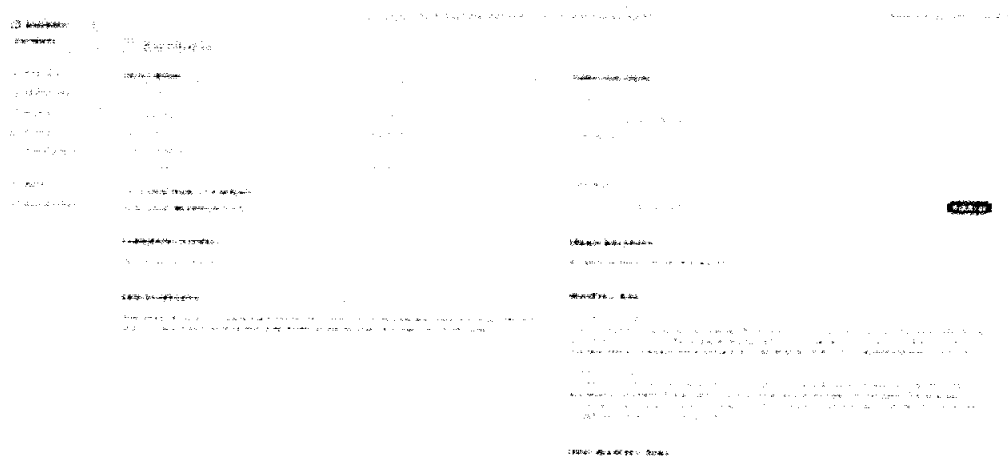
Acceder

WordPress

Después de incluir el nombre de usuario y contraseña, accederemos al panel de administración.

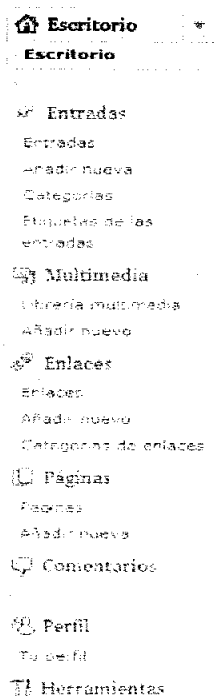
7.1.10.3. Explorando el escritorio de wordpress.com

El panel de administración permite configurar las características del sitio web. En el escritorio (primera pantalla que se ve tras iniciar sesión) se muestra información acerca de la actividad reciente en el sitio web, así como avisos de actualizaciones o información relevante de la comunidad Wordpress:

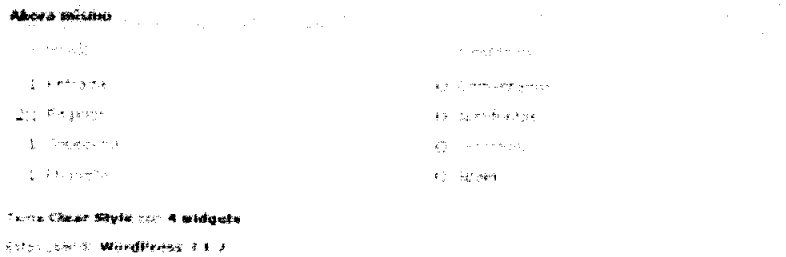


El menú para navegar en el panel de administración se encuentra en la parte izquierda de la pantalla. Es un menú de navegación intuitivo y flexible, permitiendo llegar a las funciones con pocos clics.

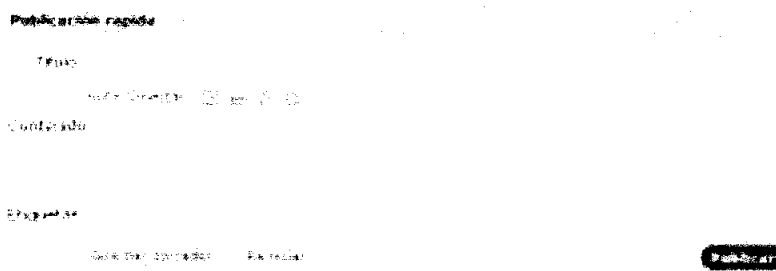
Al pasar el cursor al lado de cada objeto de contenido se mostrará un pequeño triángulo y al oprimir sobre el mismo se mostrarán las sub-secciones:



En la segunda columna del tablero tienes por defecto las estadísticas del blog (Número de entradas, páginas, comentarios, categorías, etc):



La columna de la izquierda comienza con el módulo de Publicación Rápida, una herramienta que te permite publicar entradas (noticias) de una manera sencilla:

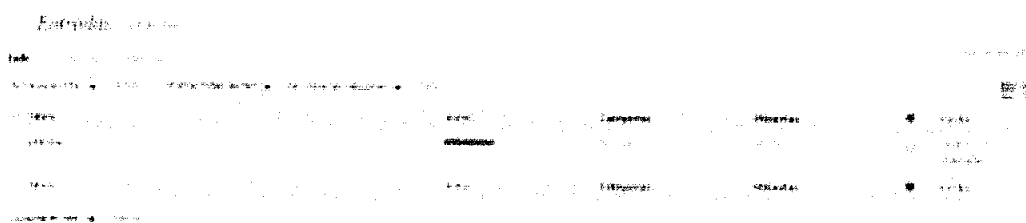


Y por último el módulo de últimos borradores:



7.1.10.4. Administración de Entradas

Las entradas son noticias que se mostrarán en orden cronológico inverso en la página de inicio de la web. Normalmente son las que más se comentan, y se incluyen en el feed RSS de la web.



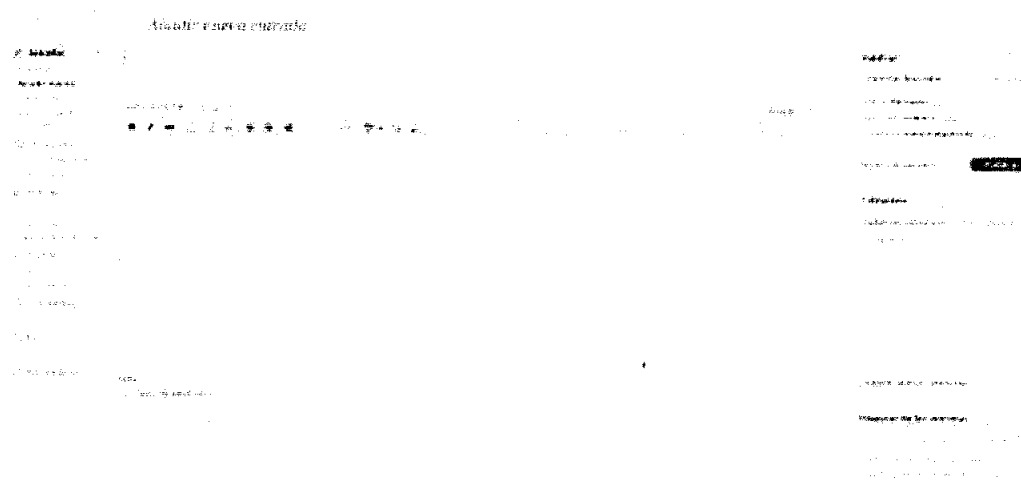
Cada entrada tiene:

- **Título:** el título de la noticia que queremos publicar.
- **Texto:** el texto de la noticia. Para añadirlo disponemos de las opciones de editores similares a Microsoft Word.
- **Categoría:** cada entrada se clasifica bajo una o varias categorías.
- **Etiquetas:** temas de la noticia. Cada tema puede tener hasta cinco.
- **(opcional):** fotografías o enlaces.

A continuación comentaremos las acciones a realizar con las entradas:

Añadir nueva entrada




Publicar una noticia es tan sencillo como mandar un e-mail. Seleccionamos Entradas -> Nueva entrada:



Disponemos un espacio en donde es posible redactar mediante un editor que tiene varias funciones.

El editor tiene dos pestañas principales: Visual y HTML. La primera permite obviar los códigos que están detrás de lo que se publica, puesto que para publicar contenido en Internet es necesario utilizar lo que se conoce como lenguaje HTML. El editor facilita el que puedas incluir las etiquetas sin que tener conocimiento de ello. Es lo que se conoce como WYSIWYG (en inglés: What You See Is What You Get). Para usuarios avanzados, con conocimientos en dicho lenguaje, en la pestaña HTML se tiene acceso al código que se está utilizando para el formato del texto.

Algunas opciones destacadas de ese editor:

-  Este botón es para poner en "negrita" el contenido seleccionado.
-  Mostraremos un texto en "cursiva".
-  Para mostrar un texto "tachado".

Para publicar una entrada sencilla son los siguientes:

1. Selecciona un título para la entrada.
2. En la caja de texto comienza a escribir el mensaje. Puedes añadir enlaces, negritas, sangrías, entre otros.
3. Añade las etiquetas y categorías que definen a esta entrada.

Una de las dudas más frecuentes gira en torno a las diferencias entre las categorías y las etiquetas. Las etiquetas son más específicas, mientras que las categorías son más generales. Por ejemplo para clasificar un artículo sobre la teoría de Sigmund Freud sobre mecanismos de defensa puedo utilizar la categoría de "Freud" y la etiqueta de mecanismos de defensa.

Si no se selecciona una categoría, se clasifica automáticamente en la categoría por defecto. Añadir etiquetas a una entrada es opcional pero recomendable.

4. Visibilidad: se puede seleccionar si la entrada estará protegida por contraseña, que sea una entrada pegajosa (que se mantenga en la parte superior del blog) o que sea privada (solo tú puedes verla).

Publicar

[Guardar borrador](#)

[Vista previa](#)

Estado: **Borrador** [Editar](#)

Visibilidad: **Público** [Editar](#)

[Publicar inmediatamente](#) [Editar](#)

[Mover a la papelera](#)

Publicar

Puedes seleccionar la fecha en que aparecerá publicada la entrada si deseas que aparezca en el futuro, o si por el contrario se publica con algo de retraso.

En la mayoría de los casos sólo es necesario añadir el título y el texto de la noticia, y seleccionar categorías y/o etiquetas, los demás campos pueden quedarse tal y como están.

Cuando tu entrada esté preparada pulsa el botón Publicar.

En varias ocasiones tendrás la necesidad de complementar una entrada con fotos e imágenes para complementar la misma. Hay dos formas principales de insertar una imagen: subiéndola al blog desde el ordenador o añadiendo la dirección de internet en la cual se encuentra la misma. Comencemos por subir una imagen desde nuestra computadora:

En el editor selecciona el icono de "Añadir una imagen" (está al lado de "Subir/Insertar").



5. Pulsas sobre "Elegir archivos", seleccionas la imagen y pulsas en "Abrir". Puedes cambiar el título, incluir una leyenda debajo de la imagen y añadir una descripción (opcional). Se te ofrece la opción de enlazar esa imagen a su URL o a la dirección de la entrada. Además, puedes seleccionar la alineación de la imagen y su tamaño. Finalmente, pulsas sobre "insertar entrada" y la imagen aparecerá en el editor.

Notarás que al pulsar sobre la imagen, tendrás la oportunidad de editarla seleccionado el icono de imagen. Puedes eliminar la imagen la imagen oprimiendo en el icono de "borrar imagen".

La segunda manera de añadir una imagen es siguiendo los mismos pasos que hemos descrito hasta llegar al cuadro de diálogo y seleccionar "Desde URL":

Ahora puedes añadir el URL de la imagen, su título, una leyenda, el tipo de alineación que prefieres y si deseas que enlace a la imagen original. Recuerda dar siempre crédito al autor de la misma, si la imagen no te pertenece. Más aún, esto solo debe hacerse cuando el dueño de la imagen lo permite.

Una entrada será más completa si le añades imágenes, videos, grabaciones o cualquier otro tipo de archivos. Ahora veremos cómo enlazar contenido subido previamente (para insertar el contenido ver sección 7 Administración de contenido multimedia):

1. Añadir un enlace de texto:
2. Ennegrece el texto que quieras enlazar.
3. Pulsa sobre el botón de enlace en la barra del editor.
4. Aparecerá una ventana emergente en donde podrás configurar el enlace.
5. Pulsa sobre "Insertar y verás que el texto cambiará a color azul y estará subrayado.
6. Enlazar desde una imagen: puede hacerse utilizando una de la

Biblioteca de Medios:

1. Pulsa sobre el botón de "Añadir una imagen". b. Selecciona Biblioteca de Medios.
2. Encuentra la imagen que quieres utilizar como objeto de enlace y pulsa sobre el enlace de "Mostrar".
3. Incluye el URL al cual quieres que enlace la imagen. e. Pulsa sobre "Insertar en entrada".
4. La imagen será insertada en la entrada y enlazada al URL que especificaste.

Editar una entrada existente

Para editar una entrada existente, desde el Escritorio seleccionamos Entradas. Nos aparecerá una lista de noticias publicadas:

Escritorio **Entradas** Añadir nueva Opciones de pantalla Ayuda

Escritorio Buscar entradas

Todo Publicadas Papelera

Acciones en lote Aplicar

Mostrar todas las fechas Ver todas las categorías Filtrar

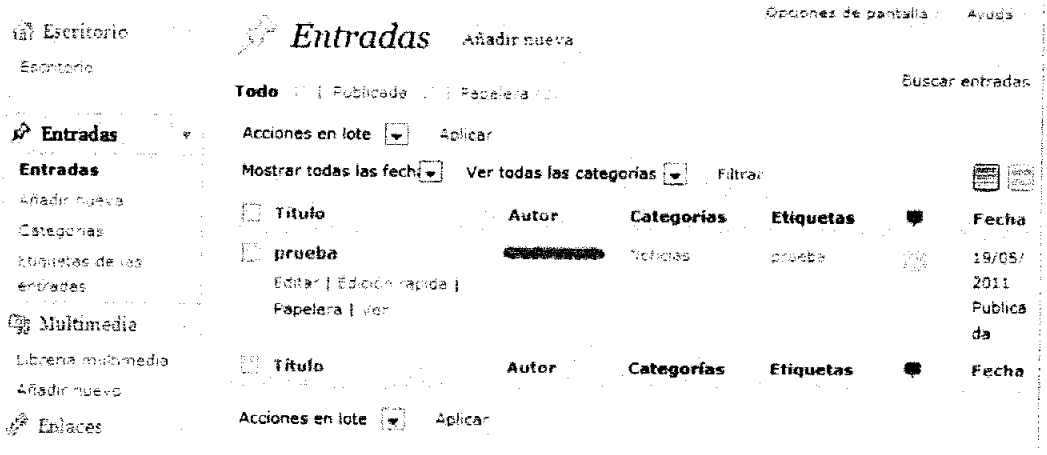
Titulo	Autor	Categorías	Etiquetas	Fecha
prueba	prueba	Noticias	prueba	19/05/2011 Publicada

Acciones en lote Aplicar

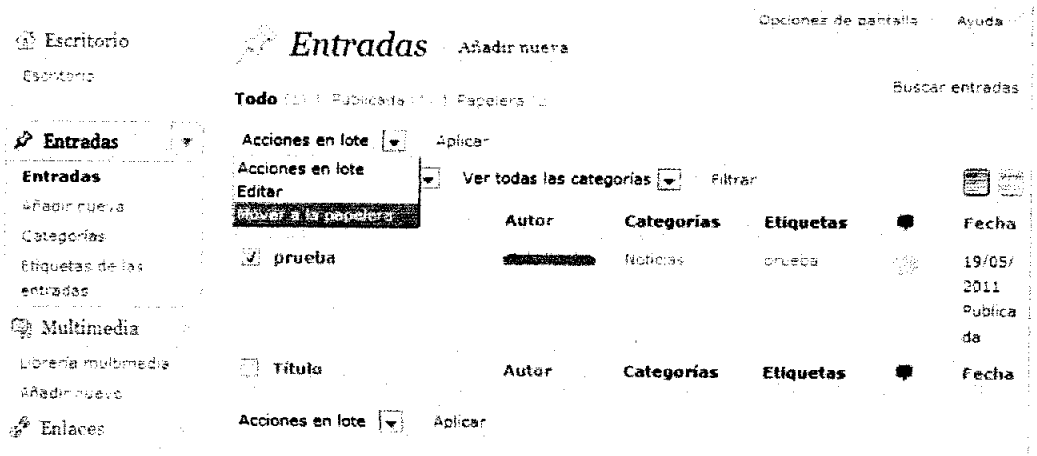
Si hacemos clic en el título de la noticia, se abrirá la ventana de edición. Podemos realizar entonces los cambios del mismo modo que cuando creamos la entrada correspondiente.

Eliminar una entrada

Si queremos eliminar una entrada existente, desde el Escritorio seleccionamos Entradas. Nos aparece una lista de noticias publicadas. Si situamos el cursor encima del título de la noticia que queremos eliminar, nos saldrá un pequeño menú en el que seleccionaremos "Papelera":



También podemos hacer clic en el cuadrado a la izquierda del título para seleccionarlo, y de la parte de arriba, donde pone "Acciones en lote" seleccionar la opción "Mover a la papelera" y pinchamos en Aplicar:



Si eliminamos la entrada por equivocación también podemos restaurarla, puesto que los mensajes eliminados van a la papelera de entradas.

7.1.10.5. Administración de Categorías

Cada entrada se clasifica bajo una o varias categorías. Las categorías permiten la clasificación de las noticias y páginas en grupos y subgrupos, de tal manera que ayuden al visitante en la navegación y uso del sitio web.

Cada categoría puede tener una categoría "padre" para crear una jerarquía dentro de la estructura de categorías.

Crear una categoría

Para crear una categoría, seleccionamos Entradas -> Categorías:

Opciones de pantalla / Ayuda

Escritorio

Entradas

Categorías

Multimedia

Enlaces

Páginas

Comentarios

Perfil

Categorías

Buscar categorías

Añadir nueva categoría

Acciones en lote Aplicar

<input type="checkbox"/> Nombre	Descripción	Slug	Entr
Noticias		noticias	2

Acciones en lote Aplicar

Nota:
Al borrar una categoría no borrarás las entradas que hay en ella. En su lugar, las entradas que sólo estén asignadas a esa categoría se asignarán a la categoría **Noticias**.

Nombre

El nombre es como aparecerá en tu sitio.

Slug

El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene solo letras, números y guiones.

Superior

Ninguna

Las categorías, a diferencia de las etiquetas, pueden tener jerarquías. Podrás tener una categoría de Jazz, y por debajo las categorías Bebop y Big Band. Totalmente opcionales.

En la parte central de la pantalla, aparece un menú para la nueva categoría y la opción de asignar qué categoría estaría por encima en el "árbol de categorías". Una vez rellenos esos datos, pulsáramos en el botón "Añadir nueva categoría" que aparece al final de la pantalla.

Editar una categoría

Para editar el nombre o la categoría superior de una categoría, desde el Escritorio seleccionamos Entradas -> Categorías. Nos aparecerá el listado de categorías en la parte de la derecha:

Si hacemos clic en el nombre de la categoría, se abrirá la ventana de edición. Podremos realizar entonces los cambios del mismo modo que cuando creamos la categoría correspondiente.

Eliminar una categoría

Si queremos eliminar una categoría, desde el Escritorio seleccionamos Entradas -> Categorías. Nos aparece el listado de categorías. Si situamos el cursor encima del nombre de la categoría que queremos eliminar, nos saldrá un pequeño menú en el que seleccionaremos "Borrar":

Entradas

Entradas
Añadir nueva

Categorías

Etiquetas de las
entradas

Multimedia

Librería multimedia

Añadir nuevo

Enlaces

Enlaces

Añadir nueva categoría

Nombre

El nombre es cómo aparecerá en tu sitio.

Slug

El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene sólo letras, números y guiones.

Acciones en lote Aplicar

Nombre	Descripción	Slug	Entradas
<input type="checkbox"/> nueva categoría	Editar Edición rápida Borrar	nueva-categoría	0
<input type="checkbox"/> Noticias		noticias	2

Acciones en lote Aplicar

Acciones en lote Aplicar

Buscar categorías

También podemos hacer clic en el cuadrado a la izquierda del nombre para seleccionarlo, y de la parte de arriba, donde pone "Acciones en lote" seleccionar la opción "Borrar" y pinchamos en Aplicar:

Entradas

Entradas
Añadir nueva

Categorías

Etiquetas de las
entradas

Multimedia

Librería multimedia

Añadir nuevo

Enlaces

Añadir nueva categoría

Nombre

El nombre es cómo aparecerá en tu sitio.

Slug

El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene sólo letras,

Acciones en lote Aplicar

- Acciones en lote
- Borrar

Nombre	Descripción	Slug	Entradas
<input checked="" type="checkbox"/> nueva categoría		nueva-categoría	0
<input type="checkbox"/> Noticias		noticias	2

Acciones en lote Aplicar

NOTA: si se elimina una categoría, no se eliminarán las entradas que hay en ella. En su lugar, las entradas que sólo estén asignadas a esa categoría se asignarán a la categoría por defecto (la primera de las categorías existentes).

7.1.10.6. Administración de Etiquetas de las entradas

Cada noticia se clasifica bajo una o varias categorías. A diferencia de las categorías, las etiquetas no tienen jerarquía de lo que no hay relación de padres e hijos como el de las categorías, pero al igual que las categorías, los nombres de etiqueta deben ser únicos.

Crear una etiqueta

Para crear una categoría, seleccionamos Entradas -> Etiquetas de las entradas:

Etiquetas de las entradas

Etiquetas populares

Acciones en lote [v] Aplicar

<input type="checkbox"/>	Nombre	Descripción	Slug	Entr
<input type="checkbox"/>	[REDACTED]		[REDACTED]	1
<input type="checkbox"/>	prueba		prueba	0

Añadir nueva etiqueta

Nombre

El nombre es cómo aparecerá en tu sitio.

Slug

El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene sólo letras, números y guiones.

Acciones en lote [v] Aplicar

En la parte central de la pantalla, aparece un menú para la nueva etiqueta. Una vez rellenos los datos, pulsaríamos en el botón "Añadir nueva etiqueta" que aparece al final de la pantalla.

Editar una etiqueta

Para editar el nombre de una etiqueta, desde el Escritorio seleccionamos Entradas -> Etiquetas de las entradas. Nos aparecerá el listado de etiquetas en la parte de la derecha:

Escritorio

Escritorio

Entradas

Añadir nueva

Categorías

Etiquetas de las entradas

Multimedia

Librería multimedia

Añadir nuevo

Enlaces

Enlaces

Añadir nuevo

Categorías de enlaces

Páginas

Páginas

Añadir nueva

Opciones de pantalla: Ayuda

Buscar etiquetas

Acciones en lote Aplicar

Nombre	Descripción	Slug	Entr
[REDACTED]		[REDACTED]	1
prueba		prueba	0

Acciones en lote Aplicar

Nombre Descripción Slug Entr

Nombre Descripción Slug Entr

Acciones en lote Aplicar

Añadir nueva etiqueta

Nombre

El nombre es como aparecerá en la URL.

Slug

El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene solo letras, números y guiones.

Si hacemos clic en el nombre de la etiqueta, se abrirá la ventana de edición. Podremos realizar entonces los cambios del mismo modo que cuando creamos la etiqueta correspondiente.

Eliminar una etiqueta

Si queremos eliminar una etiqueta, desde el Escritorio seleccionamos Entradas

-> Etiquetas de las entradas. Nos aparece el listado de etiquetas. Si situamos el cursor encima del nombre de la etiqueta que queremos eliminar, nos saldrá un pequeño menú en el que seleccionaremos "Borrar":

Escritorio

Etiquetas de las entradas

Opciones de pantalla Ayuda

Buscar etiquetas

Entradas

Añadir nueva

Categorías

Etiquetas de las entradas

Multimedia

Librería multimedia

Etiquetas populares

Añadir nueva etiqueta

Nombre

El nombre es cómo aparecerá en tu sitio.

Acciones en lote

Aplicar

Nombre	Descripción	Slug	Entradas
[Redacted]	[Redacted]	[Redacted]	1
prueba		prueba	0

Editar | Edición rápida | Borrar

También podemos hacer clic en el cuadrado a la izquierda del nombre para seleccionarlo, y de la parte de arriba, donde pone "Acciones en lote" seleccionar la opción "Borrar" y pinchamos en Aplicar:

Escritorio

Etiquetas de las entradas

Opciones de pantalla Ayuda

Buscar etiquetas

Entradas

Añadir nueva

Categorías

Etiquetas de las entradas

Multimedia

Librería multimedia

Añadir nuevo

Enlaces

Etiquetas populares

Añadir nueva etiqueta

Nombre

El nombre es cómo aparecerá en tu sitio.

Slug

Acciones en lote

Aplicar

Acciones en lote

Borrar

Nombre	Descripción	Slug	Entradas
[Redacted]	[Redacted]	[Redacted]	1
prueba		prueba	0
Nombre		Slug	Entradas

Acciones en lote

Aplicar

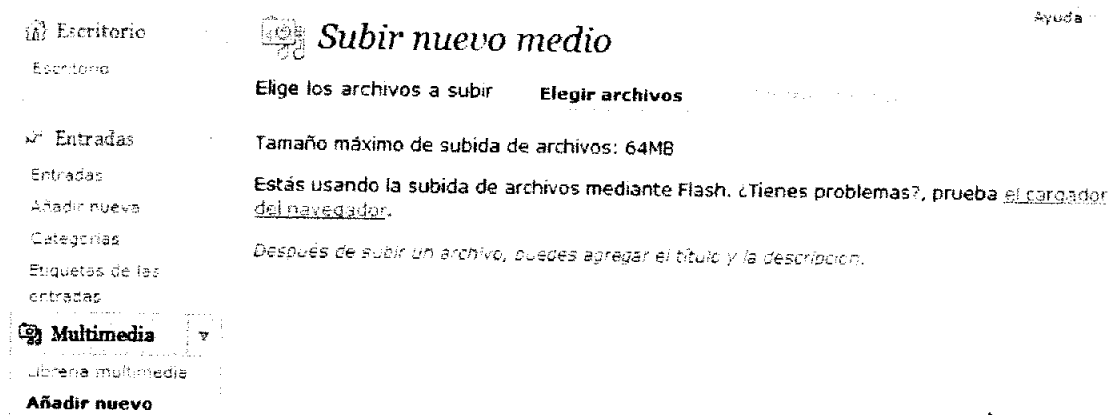
NOTA: si se elimina una etiqueta, no se eliminarán las entradas que hay en ella. El cambio no es reversible.

7.1.10.7. Administración de contenido multimedia

La pantalla de Administración de contenido multimedia permite subir contenido multimedia para después utilizarlo en entradas y páginas.

Subir un contenido multimedia

Para subir un contenido multimedia a nuestro sitio web, seleccionamos Multimedia -> Añadir nuevo:



The screenshot shows a web interface for uploading multimedia. On the left is a sidebar menu with the following items: Escritorio, Entradas, Multimedia (highlighted), and Librería multimedia. The main content area is titled 'Subir nuevo medio' and contains the following text and buttons:

- Buttons: 'Elige los archivos a subir' and 'Elegir archivos'.
- Text: 'Tamaño máximo de subida de archivos: 64MB'.
- Text: 'Estás usando la subida de archivos mediante Flash. ¿Tienes problemas?, prueba [el cargador del navegador](#).'
- Text: 'Después de subir un archivo, puedes agregar el título y la descripción.'
- Buttons: 'Añadir nuevo' and 'Librería multimedia'.

Elegimos el archivo pulsando en "Elegir archivos". Automáticamente subirá el contenido. Cuando haya terminado, nos mostrará una ventana en la que podremos elegir el título de la imagen, una leyenda si queremos que lleve pie, y la descripción:

Escritorio

Escritorio

Entradas

Entradas

Añadir nueva

Categorías

Etiquetas de las

entradas

Multimedia

Multimedia

Añadir nuevo

Etiquetas

Etiquetas

Añadir nueva

Categorías de etiquetas

Temas

Temas

Añadir nuevo

Experiencias

Experiencias

Perfiles

Perfiles

Reservata

Subir nuevo medio

Elige los archivos a subir

Elige archivos

Formato máximo de subida de archivos: 64MB

Este asistente ayuda de archivos mediante flujos de trabajo predefinidos para el contenido multimedia.

Deposita los archivos multimedia en el sistema de archivos de WordPress.



Cancelar



Nombre de archivo: Penguins.jpg

Tipo de archivo: image/jpeg

Fecha de subida: mayo 30, 2011

Dimensiones: 1024 x 768

Formato original

Título

Texto alternativo

Leyenda

Descripción

URL del archivo <http://wp-content/uploads/2011/05/Penguins.jpg>
Ubicación del archivo original
Estat.

Guardar todo los cambios

Editar un contenido multimedia

Para editar el título o descripción de un contenido multimedia, desde el Escritorio seleccionamos Multimedia. Nos aparecerá una lista de contenidos multimedia:

Escritorio

Escritorio

Librería multimedia

Añadir nuevo

Opciones de pantalla

Ayuda

Entradas

Entradas

Añadir nueva

Categorías

Etiquetas de las
entradas

Multimedia

Librería

Todos (1) | Imágenes (1) | Sin adjuntar (1)

Buscar medios

Acciones en lote Aplicar

Mostrar todas las fechas Filtrar



Archivo

Autor

Adjunto a

Fecha



Penguins

[Redacted]

(Sin adjuntar)

hace

JPG

Adjuntar

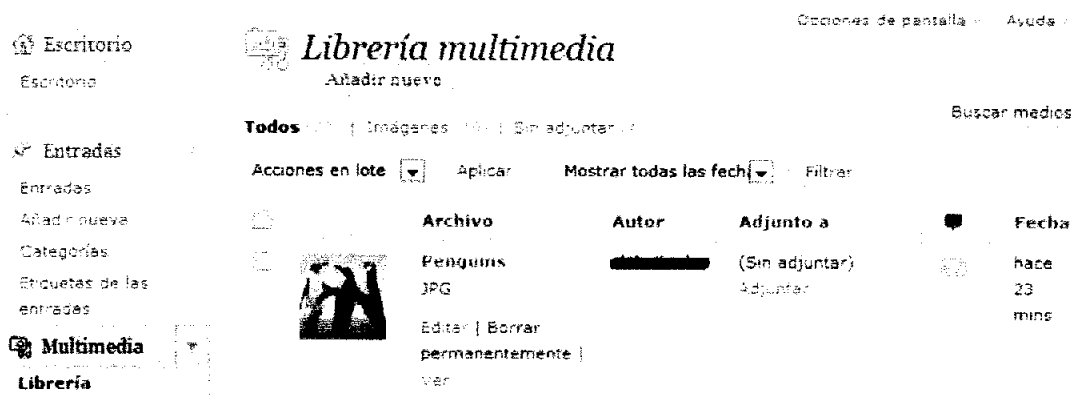
23

mins

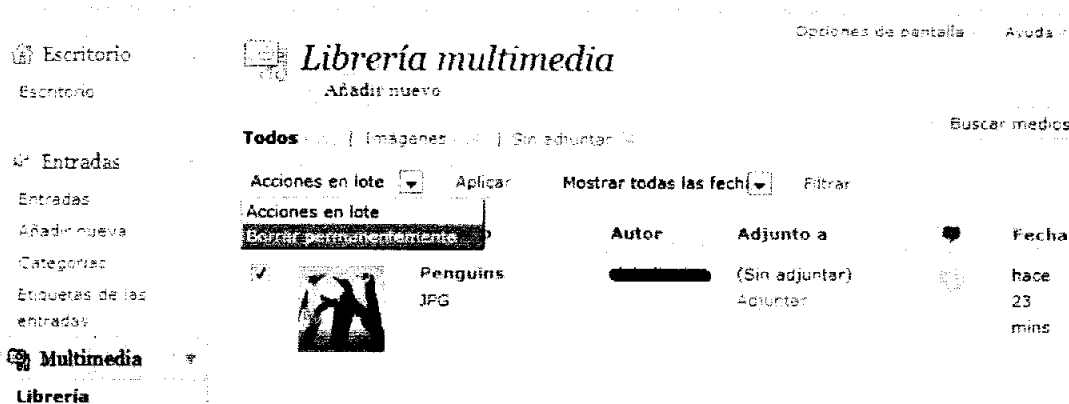
Si hacemos clic en el título del contenido, se abrirá la ventana de edición. Podremos realizar entonces los cambios del mismo modo que cuando creamos el contenido multimedia correspondiente.

Eliminar un contenido multimedia

Si queremos eliminar un contenido existente, desde el Escritorio seleccionamos Multimedia. Nos aparece una lista de contenido multimedia. Si situamos el cursor encima del título del contenido que queremos eliminar, nos saldrá un pequeño menú en el que seleccionaremos "Borrar permanentemente":



También podemos hacer clic en el cuadrado a la izquierda del título para seleccionarlo, y de la parte de arriba, donde pone "Acciones en lote" seleccionar la opción "Borrar permanentemente" y pinchamos en Aplicar:



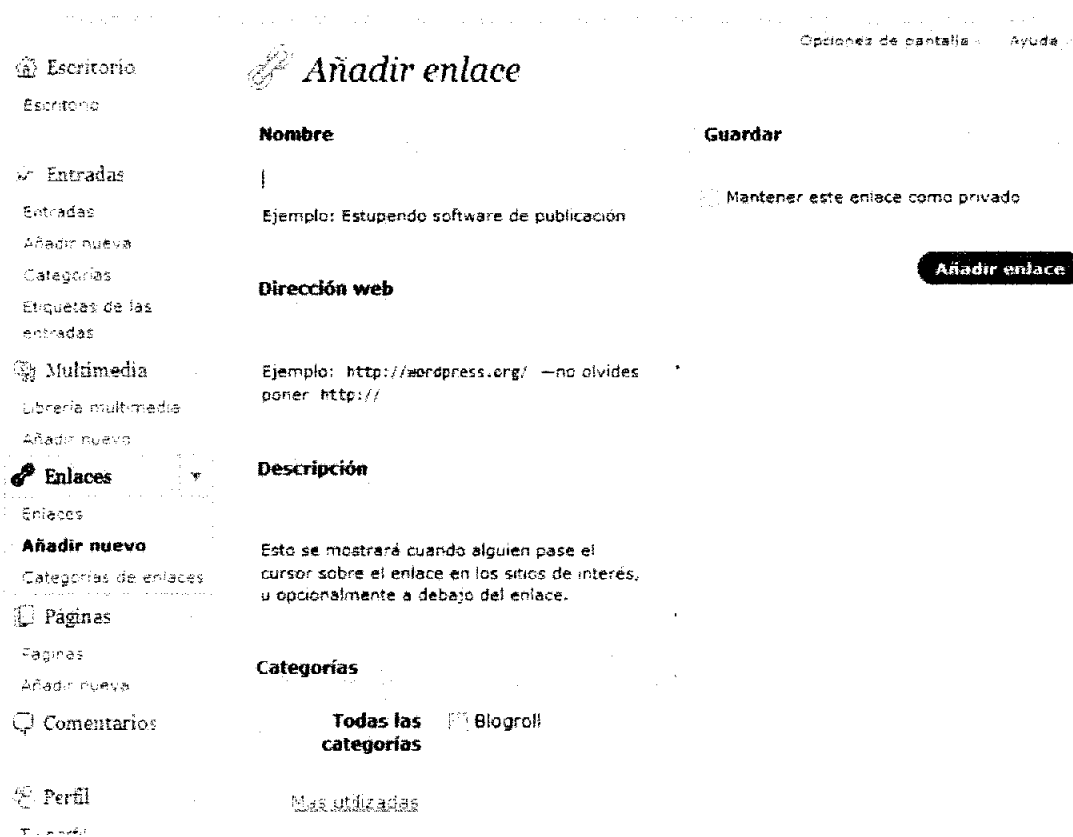
NOTA: este cambio no es reversible, por lo que hay que estar muy seguro de que se quiere eliminar el contenido antes de realizar esta acción.

7.1.10.8. Administración de Enlaces

Los enlaces que se hayan añadido previamente a alguna página o noticia web se pueden administrar de manera sencilla desde una ventana de administración.

Añadir un enlace

Para crear un enlace, seleccionamos Multimedia -> Añadir nuevo:



The screenshot shows the WordPress administration interface for adding a new link. On the left is a sidebar menu with categories like Escritorio, Entradas, Multimedia, Enlaces, Páginas, Comentarios, and Perfil. The 'Enlaces' menu item is highlighted, and a sub-menu is open showing 'Añadir nuevo'. The main content area is titled 'Añadir enlace' and contains the following fields and options:

- Nombre:** A text input field with a placeholder example: 'Ejemplo: Estupendo software de publicación'.
- Dirección web:** A text input field with a placeholder example: 'Ejemplo: http://wordpress.org/ —no olvides poner http://'.
- Descripción:** A text area with a placeholder: 'Esto se mostrará cuando alguien pase el cursor sobre el enlace en los sitios de interés, u opcionalmente a debajo del enlace.'
- Categorías:** A dropdown menu with 'Todas las categorías' selected and 'Blogroll' as an option.
- Guardar:** A section with a checkbox for 'Mantener este enlace como privado'.
- Buttons:** A 'Añadir enlace' button and a 'Cancelar' button.

Podremos seleccionar el nombre por el que conoceremos el enlace para su uso posterior, la URL a la que hace referencia, categoría a la que pertenece, y alguna otra cuestión avanzada.

Una vez que hemos rellenado todas estas cuestiones, pulsaremos en "Añadir enlace" para que quede creado.

Editar un enlace

Para editar el nombre, URL o descripción de un enlace, desde el Escritorio seleccionamos Enlaces. Nos aparecerá una lista de enlaces:

Escritorio **Enlaces** Añadir nuevo Opciones de pantalla Ayuda

Escritorio Buscar enlaces

Acciones en lote Aplicar Ver todas las categorías Filtrar

Nombre	URL	Categorías	Relación con el enlace (XFN)	Visible	Valor
Documentation	codex.wordpress.org	Blogroll		Sí	0
Plugins	wordpress.org/extend/plugins	Blogroll		Sí	0
Suggest Ideas	wordpress.org/extend/ideas	Blogroll		Sí	0
Support Forum	wordpress.org/support	Blogroll		Sí	0
Themes	wordpress.org/extend/themes	Blogroll		Sí	0
WordPress Blog	wordpress.org/news	Blogroll		Sí	0
WordPress Planet	planet.wordpress.org	Blogroll		Sí	0

Acciones en lote Aplicar

Si hacemos clic en el título del enlace, se abrirá la ventana de edición. Podremos realizar entonces los cambios del mismo modo que cuando creamos el enlace correspondiente.

Eliminar un enlace

Si queremos eliminar un enlace, desde el Escritorio seleccionamos Enlaces. Nos aparece una lista de enlaces. Si situamos el cursor encima del título del enlace que queremos eliminar, nos saldrá un pequeño menú en el que seleccionaremos "Borrar":

Escritorio

Buscar enlaces

Entradas

Acciones en lote

Aplicar

Ver todas las categorías

Filtrar

Entradas

Añadir nueva

Categorías

Etiquetas de las entradas

Multimedia

Biblioteca multimedia

Añadir nuevo

Enlaces

Enlaces

Añadir nuevo

Categorías de enlaces

Páginas

Páginas

Añadir nueva

Comentarios

Perfil

Tu perfil

Herramientas



Nombre

URL

Categorías

Relación con el enlace (XFN)

Visible

Valora



Documentation

codex.wordpress.org

Blogroll

Sí

0

Editar | Borrar



Plugins

wordpress.org/extensiones/plugins

Blogroll

Sí

0



Suggest Ideas

wordpress.org/extensiones/ideas

Blogroll

Sí

0



Support Forum

wordpress.org/support

Blogroll

Sí

0



Themes

wordpress.org/extensiones/themes

Blogroll

Sí

0



WordPress Blog

wordpress.org/news

Blogroll

Sí

0



WordPress Planet

planet.wordpress.org

Blogroll

Sí

0



Nombre

URL

Categorías

Relación con el enlace (XFN)

Visible

Valora

Acciones en lote

Aplicar

También podemos hacer clic en el cuadrado a la izquierda del nombre para seleccionarlo, y de la parte de arriba, donde pone "Acciones en lote" seleccionar la opción "Borrar permanentemente" y pinchamos en Aplicar:

Escritorio

Buscar enlaces

Entradas

Acciones en lote

Aplicar

Ver todas las categorías

Filtrar

Entradas

Añadir nueva

Categorías

Etiquetas de las entradas

Multimedia



Nombre

URL

Categorías

Relación con el enlace (XFN)

Visible

Valora



Documentation

codex.wordpress.org

Blogroll

Sí

0



Plugins

wordpress.org/extensiones/plugins

Blogroll

Sí

0

NOTA: este cambio no es reversible, por lo que hay que estar muy seguro de que se quiere eliminar el enlace antes de realizar esta acción.

7.1.10.9. Administración de Páginas

Las entradas o noticias son objetos orientados en el tiempo. Se escriben en un tiempo específico, y ese tiempo define su contexto. Ejemplos de noticias serían las actividades del mes que viene, boletín mensual, etc.

Las páginas, por el contrario, se usan la mayor parte de las veces para presentar la información atemporal sobre usted o su sitio web (información atemporal). Ejemplos de páginas a incluir serían "Acerca de", "Contacto", "copyright", etc.

Resumiendo, las páginas:

- Son para el contenido que es menos dependiente del tiempo que las

Entradas.

- Pueden organizarse en páginas y SubPáginas.
- Pueden usar diferentes Plantillas de Página que pueden incluir Archivos de Plantilla, Etiquetas de Plantilla y otro código PHP.

A continuación comentaremos las acciones a realizar con las páginas:

Añadir nueva página

Publicar una página es tan sencillo como mandar un e-mail. Seleccionamos Páginas -> Añadir nueva:

The screenshot shows a CMS interface with a sidebar on the left and a main content area. The sidebar contains a menu with items like 'Inicio', 'Entradas', 'Páginas', and 'Añadir nueva'. The main content area is titled 'Añadir nueva página' and contains a form with the following fields and options:

- Publicar:** A section with 'Enviar borrador' and 'Publicar ahora' buttons.
- Plantilla:** A dropdown menu with 'Plantilla: [Seleccionar plantilla]' and a 'Publicar' button.
- Mostrar la página:** A 'Publicar' button.
- Administrador de página:** A dropdown menu with 'Superior' selected.
- Temas:** A dropdown menu with 'Temas [Seleccionar tema]'.
- Origen:** A dropdown menu with '0' selected.
- Comentarios:** A section with 'Comentarios en esta página' and 'La página se publicará automáticamente'.
- Etiquetas:** A section with 'Etiquetas: [Etiquetas]'.

Disponemos un espacio en donde es posible redactar mediante un editor que tiene varias funciones.

El editor tiene dos pestañas principales: Visual y HTML. La primera permite obviar los códigos que están detrás de lo que se publica, puesto que para publicar contenido en Internet es necesario utilizar lo que se conoce como lenguaje HTML. El editor facilita el que puedas incluir las etiquetas sin que tener conocimiento de ello. Es lo que se conoce como WYSIWYG (en inglés: What You See Is What You Get). Para usuarios avanzados, con conocimientos en dicho lenguaje, en la pestaña HTML se tiene acceso al código que se está utilizando para el formato del texto.

Algunas opciones destacadas de ese editor:

■ Este botón es para poner en "negrita" el contenido seleccionado. Mostraremos un texto en "cursiva".

* Para mostrar un texto "tachado".

Para publicar una entrada sencilla son los siguientes:

1. Selecciona un título para la entrada.
2. En la caja de texto comienza a escribir el mensaje. Puedes añadir enlaces, negritas, sangrías, entre otros.
3. Añade las etiquetas y categorías que definen a esta entrada.

Una de las dudas más frecuentes gira en torno a las diferencias entre las categorías y las etiquetas. Las etiquetas son más específicas, mientras que las categorías son más generales. Por ejemplo para clasificar un artículo sobre la teoría de Sigmund Freud sobre mecanismos de defensa puedo utilizar la categoría de "Freud" y la etiqueta de mecanismos de defensa.

Si no se selecciona una categoría, se clasifica automáticamente en la categoría por defecto. Añadir etiquetas a una entrada es opcional pero recomendable.

Visibilidad: se puede seleccionar si la entrada estará protegida por contraseña, que sea una entrada pegajosa (que se mantenga en la parte superior del blog) o que sea privada (solo tú puedes verla).

Publicar

Guardar borrador

Vista previa

Estado: **Borrador** [Editar](#)

Visibilidad: **Público** [Editar](#)

 **Publicar inmediatamente** [Editar](#)

[Mover a la papelera](#)

Publicar

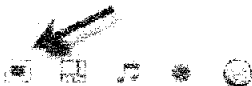
En la mayoría de los casos sólo es necesario añadir el título y el texto de la página, y seleccionar categorías y/o etiquetas, los demás campos pueden quedarse tal y como están.

Cuando tu página esté preparada pulsa el botón Publicar.

En varias ocasiones tendrás la necesidad de complementar una página con fotos e imágenes para complementar la misma. Hay dos formas principales de insertar una imagen: subiéndola al blog desde el ordenador o añadiendo la dirección de internet en la cual se encuentra la misma. Comencemos por subir una imagen desde nuestra computadora:

En el editor selecciona el icono de "Añadir una imagen" (está al lado de "Subir/Insertar").

Subir/Insertar



Pulsas sobre "Elegir archivos", seleccionas la imagen y pulsas en "Abrir". Puedes cambiar el título, incluir una leyenda debajo de la imagen y añadir una

Descripción (opcional). Se te ofrece la opción de enlazar esa imagen a su URL o a la dirección de la entrada. Además, puedes seleccionar la alineación de la imagen y su tamaño. Finalmente, pulsas sobre "insertar entrada" y la imagen aparecerá en el editor.

Notarás que al pulsar sobre la imagen, tendrás la oportunidad de editarla seleccionado el icono de imagen. Puedes eliminar la imagen la imagen oprimiendo en el icono de "borrar imagen".

La segunda manera de añadir una imagen es siguiendo los mismos pasos que hemos descrito hasta llegar al cuadro de diálogo y seleccionar "Desde URL":

Ahora puedes añadir el URL de la imagen, su título, una leyenda, el tipo de alineación que prefieres y si deseas que enlace a la imagen original. Recuerda dar siempre crédito al autor de la misma, si la imagen no te pertenece. Más aún, esto solo debe hacerse cuando el dueño de la imagen lo permite.

Una página será más completa si le añades imágenes, videos, grabaciones o cualquier otro tipo de archivos. Ahora veremos cómo enlazar contenido subido previamente (para insertar el contenido ver sección 7 Administración de contenido multimedia):

Añadir un enlace de texto:

1. Ennegrece el texto que quieras enlazar.
2. Pulsa sobre el botón de enlace en la barra del editor.
3. Aparecerá una ventana emergente en donde podrás configurar el enlace.
4. Pulsa sobre "Insertar y verás que el texto cambiará a color azul y estará subrayado.
5. Enlazar desde una imagen: puede hacerse utilizando una de la Biblioteca de Medios:
6. Pulsa sobre el botón de "Añadir una imagen". b. Selecciona Biblioteca de Medios.
7. Encuentra la imagen que quieres utilizar como objeto de enlace y pulsa sobre el enlace de "Mostrar".
8. Incluye el URL al cual quieres que enlace la imagen. e. Pulsa sobre "Insertar en entrada".
9. La imagen será insertada en la página y enlazada al URL que especificaste.



Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Inicio

Comentarios

Todos

Aprobados en el momento

Aprobados

En moderación

En spam

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Comentarios

Al poner el curso en algún comentario aparecerá un menú con varias opciones, las cuales te permitirán rechazar el comentario, identificarlo como spam, editarlo, editarlo rápidamente o responder al mismo. Si utilizas el menú de acciones puedes marcar varios comentarios y aplicar la acción correspondiente.

Los comentarios están organizados en varias categorías: pendientes de moderación, aprobados y de spam.

WordPress incluye un sistema muy efectivo para manejar spam llamado Akismet. En nuestras páginas dejamos este sistema activado de forma automática.

7.1.10.11. Administración del Perfil

Cada cuenta de usuario incluye un perfil del mismo. Para ver tu perfil pulsa en "Perfil":

7.2. La Empresa

7.2.1. Misión

COOCI es una empresa de servicios de construcción en el ámbito público y privado, cuya misión es satisfacer las necesidades de sus clientes antes, durante y después de finalizado el proyecto. Lo anterior dando cumplimiento a los estándares de calidad y plazos fijados por éste, generando con ello relaciones duraderas, basadas en sus años de experiencia y profesionalismo.

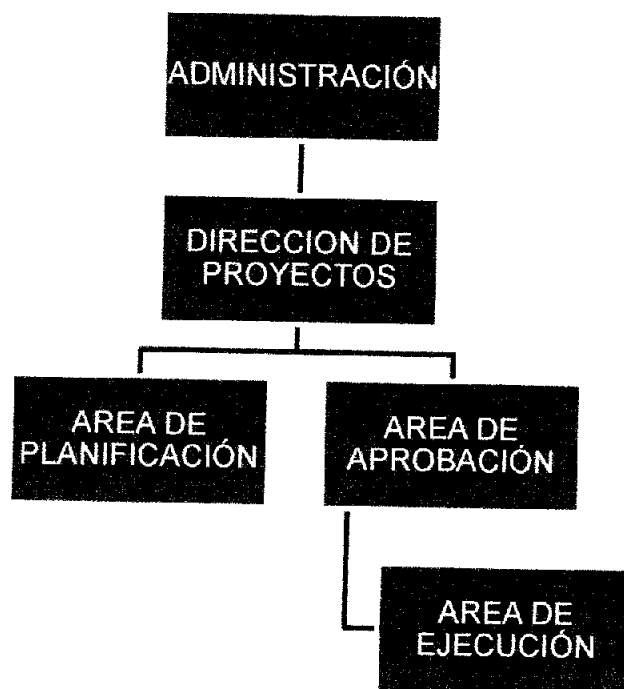
7.2.2. Visión

Lograr ser una empresa líder y competitiva en el ramo de la construcción a nivel nacional, tomando como estandarte "calidad" "responsabilidad" y "servicio". Ser reconocidos y estar posicionados como una de las mejores empresas constructoras de nuestro país.

7.2.3. Sector a que se dirige

Se dirige al sector de la construcción.

Organigrama



7.3. Marco Conceptual

Es necesario identificar los conceptos más comunes manejados y conocidos dentro de los distintos Sistemas existentes tales como sistema de escritorio o aplicación de escritorio y sistema web o aplicación web , pues estos conceptos tienden a ser confundidos.

Para efectos de análisis es necesario establecer las siguientes definiciones:

7.3.1. Sistema Informático:

Un sistema informático es un sistema que permite almacenar y procesar información; como todo sistema, es el conjunto de partes interrelacionadas: en este caso, hardware, software y recursos humanos

Sistema de escritorio o aplicación de escritorio:

El sistema o aplicación de escritorio, también conocido como un recurso o un "app", es un programa informático diseñado para ayudar al usuario a realizar singulares o múltiples tareas específicas relacionadas. Ayuda a resolver problemas en el mundo real. Los ejemplos incluyen software empresarial, software de contabilidad, suites de oficina, software de gráficos, y reproductores de medios.

7.3.2. Aplicación web:

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

7.3.3. Aplicación informática:

En informática, una aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos. Esto lo diferencia principalmente de otros tipos de programas como los sistemas operativos (que hacen funcionar al ordenador), las utilidades (que realizan tareas de mantenimiento o de uso general), y los lenguajes de programación (con el cual se crean los programas informáticos).

Programa Informático:

Un programa informático es un conjunto de instrucciones que una vez ejecutadas realizarán una o varias tareas en una computadora.

7.3.4. Sistema Operativo:

Un sistema operativo (SO o, frecuentemente, OS —del inglés Operating System—) es un programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes y anteriores próximos y viceversa.

7.3.5. Lenguajes de Programación:

Un lenguaje de programación es un lenguaje formal diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras.

7.3.6. Utilidad (informática):

- En informática, una utilidad es una herramienta que realiza:
- Tareas de mantenimiento
- Soporte para la construcción y ejecución de programas
- Las tareas en general

En donde se incluyen las bibliotecas de sistema, middleware, herramientas de desarrollo, etc.

7.3.7. Middleware:

Middleware es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos.

7.3.8. Framework:

Un framework o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Componente:

Un componente de software es un elemento de un sistema software que ofrece un conjunto de servicios, o funcionalidades, a través de interfaces definidas.

7.3.9. Estándares:

Estándares web es un término muy general utilizado para referirse a estándares y otras especificaciones técnicas que definen y describen aspectos de la World Wide Web. En años recientes, el término ha sido frecuentemente asociado con la tendencia de aprobar un conjunto de mejores prácticas estandarizadas para construir sitios web y a la filosofía del diseño y desarrollo web que incluye esos métodos.

7.3.10. Optimizar:

La optimización de software es el proceso de modificación de un software para hacer que algún aspecto del mismo funcione de manera más eficiente y/o utilizar menos recursos (mayor rendimiento). En general, un programa puede ser optimizado para que se ejecute más rápidamente, o sea capaz de operar con menos memoria u otros recursos, o consuman menos energía.

VIII. Marco metodológico

I.1. Tipo de investigación

La presente investigación es de tipo aplicada ya que podrá brindar solución a un problema determinado. La necesidad para este caso es la creación de un sistema en el área de planificación y ejecución de proyectos de construcción que posee la cooperativa de obras civiles COOCI, en el año 2015; para alcanzar este propósito primeramente se realizó un análisis de los procedimientos que se emplean en esta organización; luego se tomó como base un conjunto de conocimientos teóricos y prácticos adquiridos en la carrera de Licenciatura en ciencias de la computación determinando así cuáles serían las estrategias y herramientas más apropiadas a utilizar; por último esta investigación permitirá una solución oportuna a las debilidades de la organización.

I.2. Área de estudio

El área de estudio es precisamente donde se implementa dicho sistema, en este caso La cooperativa de obras civiles COOCI.

I.3. Universo y Muestra

I.3.1. Universo

Es una empresa Nicaragüense legalmente constituida desde el año 2003 con experiencia comprobada en el ramo de la Arquitectura y la Ingeniería aportada por el personal calificado que la compone. La empresa está siendo vanguardista en el uso de la tecnología y la técnica para proveer soluciones en el área del diseño arquitectónico y las construcciones verticales y horizontales acordes a las crecientes necesidades, con enfoques de eficiencia en el uso de los recursos humanos y materiales y la adaptabilidad a las nuevas demandas del mercado.

8.3.2. Ubicación geográfica

La empresa COOCI se encuentra en: Managua, Managua Resd. Los Robles, del hotel Colón, 1 cuadra al Este, 3 cuabras al Sur, # 95

8.3.3. Muestra

Dirección de Proyectos es el área que se encarga de la administración y control de los proyectos, desarrollan proyectos sin límite máximo o mínimo en las áreas a desarrollar cuenta con el respaldo de la satisfacción de los clientes en cada uno de los proyectos desarrollados.

8.4. Método de recolección de datos

Los métodos utilizados para la recolección de datos fueron:

- Entrevista a director del Área de planificación de proyectos, área encargada en brindar apoyo en la elaboración de proyectos de obras civiles.
- Revisión de documentos existentes.

La planificación de las entrevistas fue realizada tomando en cuenta los siguientes elementos:

- Establecer el objetivo de la entrevista
- Selección de los entrevistados
- Preparación del entrevistado
- Selección del tipo y la estructura de las preguntas.

8.5. Hipótesis

Un sistema de gestión de proyectos de construcción para la cooperativa de obras civiles (COOCI) le permitirá a la Dirección mejorar el control de las obras civiles y tomar decisiones en el momento preciso de la ejecución.

8.6. Operacionalización de las variables

OBJETIVO	VARIABLE	INDICADORES	INSTRUMENTO
<p>Determinar los procesos principales del sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.</p>	<p>Los procesos principales del sistema para la planificación y ejecución de proyectos de construcción en la cooperativa de obras civiles COOCI.</p>	<p>Cantidad de procesos automatizados en el tiempo previsto.</p>	<p>Entrevista</p>
		<p>% de proyectos planificados</p>	<p>Observación</p>
		<p>Cantidad de obras solicitadas por los clientes</p>	<p>Observación</p>
<p>Utilizar los marcos de trabajo gráfico como Visual Studio 2010, Neatbean, codeigniter para el diseño de las interfaces de usuario para el sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.</p>	<p>interfaces de usuario para el sistema</p>	<p>Cantidad de pantallas de entrada terminadas en el tiempo</p>	<p>Microsoft Visual Studio 2010 express , Neatbean, codeigniter</p>
<p>Cantidad de informes terminados</p>			
<p>Aplicar el Modelo Vista Controlador (MVC) como método de desarrollo de las rutinas del sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015.</p>	<p>las rutinas del sistema automatizado</p>	<p>% de avance en la codificación</p>	<p>Microsoft Visual Studio 2010, neatbean</p>

8.7. Cronograma de trabajo

PROYECTO: Sistema de gestión de proyectos de construcción											
Items	Actividades	TIEMPO DE EJECUCION EN SEMANAS								Días Calendario	
		Septiembre	Octubre	Noviembre	Enero	Febrero	Marzo	Abril			
	Recolección de la Información	■	■	■							21
	Análisis de la Información		■	■	■						35
	Requerimientos Funcionales		■	■	■						30
	Requerimientos No Funcionales		■	■	■						30
	Identificación de Usuarios y Roles			■	■						30
	Casos de Uso			■	■	■					35
	Diagramas de Clase			■	■	■					30
	Diseño y Creación de la Base de datos				■	■					19
	Codificación					■	■	■			60
	Catálogos/Registros					■	■	■			60
	Proceso Principal					■	■	■	■		98
	Consulta/Informes					■	■	■	■	■	65

8.8. Estudio de Factibilidad

8.8.2. Estudio de Factibilidad

Estudio de Factibilidad

Cantidad	Personal	Duración	Costo
1	Analista de Sistema	3 meses	U\$ 500
1	Diseñador Grafico	3 Meses	U\$ 100
3	Programadores	1 Año	U\$ 500
		Total	U\$ 1100

Cantidad	Descripcion	Costo Unidad	Costo total
3	Alquiler de maquinas	U\$ 1.25	U\$ 3.75
250	Impresión de papel	U\$ 0.03	U\$ 9.25
		Total	U\$ 13.00

Software			
Cantidad	Equipo	Costo Unidad	Costo total
1	Mysql	U\$ 0	U\$ 0
3	Aptana Studio	U\$ 0	U\$ 0
3	Visual Studio 2010 Express	U\$ 0	U\$ 0
3	Gimp	U\$0	U\$ 0
3	App Server, WampServer	U\$0	U\$ 0
		Total	U\$ 0

IX. Metodología a implementar

9.2 ¿Qué es un modelo para el desarrollo de software?

Un modelo para el desarrollo de software es una perspectiva de las actividades que ocurren durante el diseño y el desarrollo del software, se pretende determinar el orden de las etapas implicadas en el sistema y los criterios de transición asociadas entre estas etapas. Un modelo de ciclo de vida del software:

- Describe las etapas primordiales del desarrollo de software.
- Define las etapas primarias esperadas para ser aplicadas durante esas etapas.
- Ayuda a administrar el progreso del desarrollo, y
- Provee un espacio de trabajo para la definición de un detallado proceso de desarrollo de software.

Así, los modelos por una parte proveen una guía a los ingenieros de software con el fin de establecer las diversas actividades técnicas en el proyecto, por otra parte suministran un marco para la administración del desarrollo y el mantenimiento del software, en el sentido en que permiten estimar recursos, definir puntos de control intermedios, monitorear el avance, etc.

El modelo a implementar en el Sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015 será el modelo espiral. (Ver fig#4)

9.3 ¿Qué es un modelo espiral?

El modelo espiral en el desarrollo del software es un modelo meta del ciclo de vida del software donde el esfuerzo del desarrollo es iterativo, tan pronto culmina un esfuerzo del desarrollo por ahí mismo comienza otro; además en cada ejecución del desarrollo se sigue cuatro pasos principales:

1. Determinar o fijar los objetivos.

En este paso se definen los objetivos específicos para posteriormente identificar las limitaciones del proceso y del sistema de software, además se diseña una planificación detallada de gestión y se identifican los riesgos.

2. Análisis del riesgo.

En este paso se efectúa un análisis detallado para cada uno de los riesgos identificados del proyecto, se definen los pasos a seguir para reducir los riesgos y luego del análisis de estos riesgos se planean estrategias alternativas.

3. Desarrollar, verificar y validar.

En este tercer paso, después del análisis de riesgo, se eligen un paradigma para el desarrollo del sistema de software y se lo desarrolla.

4. Planificar.

En este último paso es donde el proyecto se revisa y se toma la decisión si se debe continuar con un ciclo posterior al de la espiral. Si se decide continuar, se desarrollan los planes para la siguiente fase del proyecto.

Con cada iteración alrededor de la espiral, se crean sucesivas versiones del software, cada vez más completas y, al final, el sistema de software ya queda totalmente funcional.

La diferencia principal entre el modelo espiral y otros modelos (ej.: cascada, evolutivo, incremental, etc.) es la evaluación del riesgo. El riesgo es todo lo que pueda salir mal en un proyecto de desarrollo de software es por ello que hemos decidido implementar este modelo a nuestra aplicación.

Un modelo espiral comienza con la determinación de los objetivos tanto funcionales como de rendimiento. Después se enumeran algunas formas posibles de alcanzar estos objetivos identificando las fuentes de riesgos posibles. Luego continuamos con el siguiente paso que es resolver estos riesgos y llevar a cabo

las actividades de desarrollo, para finalizar con la planificación del siguiente ciclo de la espiral.

9.4 Características del modelo en espiral para el desarrollo de software

- Es considerado como un modelo evolutivo ya que combina el modelo clásico con el diseño de prototipos.
- Contiene una nueva etapa que es el análisis de riesgos, no incluida anteriormente.
- Este modelo es el indicado para desarrollar software con diferentes versiones actualizadas como se hace con los programas modernos de PC's.
- La ingeniería puede desarrollarse a través del ciclo de vida clásico o el de construcción de prototipos.
- Este es el enfoque más realista actualmente.

El modelo en espiral esta compartida en varias actividades estructurales, también llamadas regiones de tareas. Existen seis regiones de tareas que son:

Comunicación con el cliente: esta es una tarea requerida para establecer comunicación entre el desarrollador y el cliente.

Planificación: esta tarea es necesaria aplicarla para poder definir los recursos, el tiempo y otras informaciones relacionadas con el proyecto, es decir, son todos los requerimientos.

Construcción y adaptación: esta tarea es requerida en el modelo espiral porque se necesita construir, probar, instalar y proporcionar soporte al usuario.

Evaluación el cliente: esta también es una tarea principal, necesaria para adquirir la reacción del cliente según la evaluación de las representaciones del software creadas durante la etapa de ingeniería y la de implementación creada durante la etapa de instalación.

Análisis de riesgos: esta es una de las tareas principales por lo que se aplica el modelo en espiral, es requerida para evaluar los riesgos técnicos y otras informaciones relacionadas con el proyecto.

Ingeniería: esta es una tarea necesaria ya que se requiere construir una o más representaciones de la aplicación.

9.5 Ventajas del modelo espiral

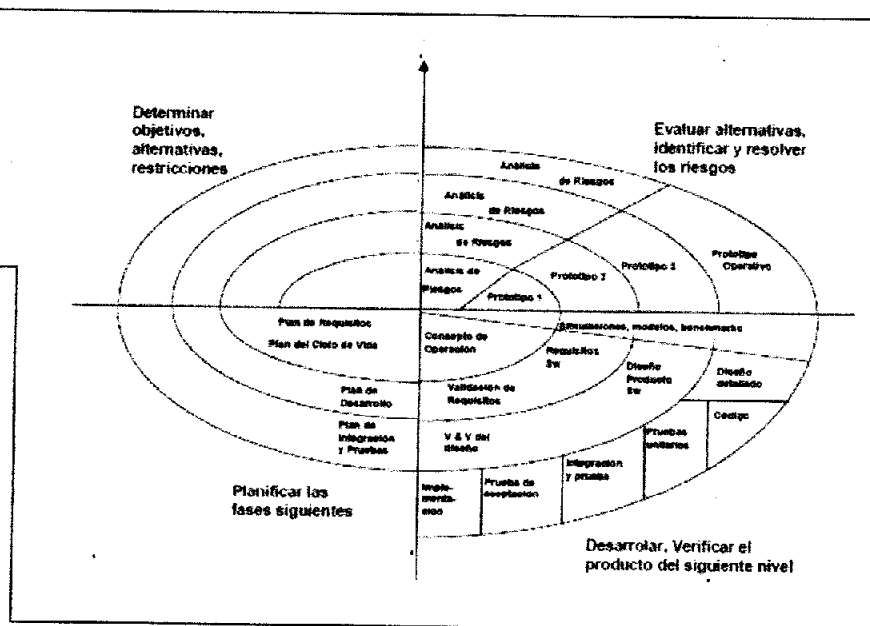
- No requiere una definición completa de los requerimientos del software a desarrollar para comenzar su funcionalidad.
- En la terminación de un producto desde el final de la primera iteración es muy factible aprobar los requisitos.
- Sufrir retrasos corre un riesgo menor, porque se comprueban los conflictos presentados tempranamente y existe la forma de poder corregirlos a tiempo.

9.6 Desventajas del modelo espiral

- Existe complicación cuando se evalúa los riesgos.
- Se requiere la participación continua por parte del cliente.
- Se pierde tiempo al volver producir inicialmente una especificación completa de los requerimientos cuando se modifica o mejora el software.

Figura #4

Ciclo de vida de un modelo espiral



X. Marco de Desarrollo

10.1. Ficha requerimientos

10.1.2. Ficha de requerimientos Funcionales

ID del Requerimiento:	RF-001
Nombre del Requerimiento:	Registro
Descripción:	El sistema debe permitir al administrador guardar un registro o historial de sus actividades, tales como títulos de propiedades, ventas, arriendos, etc.
Prioridad:	Alta

ID del Requerimiento:	RF-002
Nombre del Requerimiento:	Proyecto
Descripción:	El sistema debe permitir al administrador la posibilidad de guardar la información de los proyectos, además de poder realizar los seguimientos y las valoraciones a cada proyecto u obra civil
Prioridad:	Media

ID del Requerimiento:	RF-003
Nombre del Requerimiento:	Informe
Descripción:	El sistema debe permitir al administrador brindar una serie de informes para realizar una toma de decisión precisa.
Prioridad:	Media

ID del Requerimiento:	RF-004
Nombre del Requerimiento:	Consulta
Descripción:	El sistema debe permitir al administrador realizar consulta para la verificación de algunos datos existentes en la bd para realizar una acción al momento.
Prioridad:	Alta

ID del Requerimiento:	RF-005
Nombre del Requerimiento:	Area del Cliente
Descripción:	El sistema debe permitir al cliente visualizar la información del proyecto de obra civil de forma que la pueda ver desde cualquier parte del mundo.
Prioridad:	Alta

11. ID del Requerimiento:	RFP-006
Nombre del Requerimiento:	Agregar Actividad
Descripción:	El sistema debe permitir al Usuario guardar un registro de actividades a seguir en un proyecto etc.
Prioridad:	Alta

ID del Requerimiento:	RFP -007
Nombre del Requerimiento:	Agregar Fase
Descripción:	El sistema debe permitir al Usuario guardar un registro de Fases a seguir en una Actividad determinada etc.
Prioridad:	Media

ID del Requerimiento:	RFP -008
Nombre del Requerimiento:	Cumplimiento del Proyecto
Descripción:	El sistema debe permitir al Usuario guardar información referente al avance o nivel de cumplimiento de una Fase o Actividad.
Prioridad:	Media

ID del Requerimiento:	RFP-009
Nombre del Requerimiento:	Datos del Empleado
Descripción:	El sistema debe permitir al Usuario guardar un listado de los empleados asignados a una Actividad.
Prioridad:	Media

ID del Requerimiento:	RFP-010
Nombre del Requerimiento:	Retraso de Actividades
Descripción:	El sistema debe permitir al Usuario Crear un registro de los Retrasos de Proyectos indicando así que Fase y que Actividad fueron afectadas, etc.
Prioridad:	Alta

10.1.2. Ficha requerimientos no funcionales:

ID del Requerimiento:	RNF-001
Nombre del Requerimiento:	Seguridad del Sistema
Descripción:	El sistema no debe ser vulnerado, pues la información debe ser de carácter confidencial.
Prioridad:	Alta

ID del Requerimiento:	RNF-002
Nombre del Requerimiento:	Simplicidad
Descripción:	El sistema debe ser fácil de usar.
Prioridad:	Media

ID del Requerimiento:	RNF-003
Nombre del Requerimiento:	Validaciones
Descripción:	El sistema debe validar cada dato que sea ingresado en los formularios.
Prioridad:	Alta

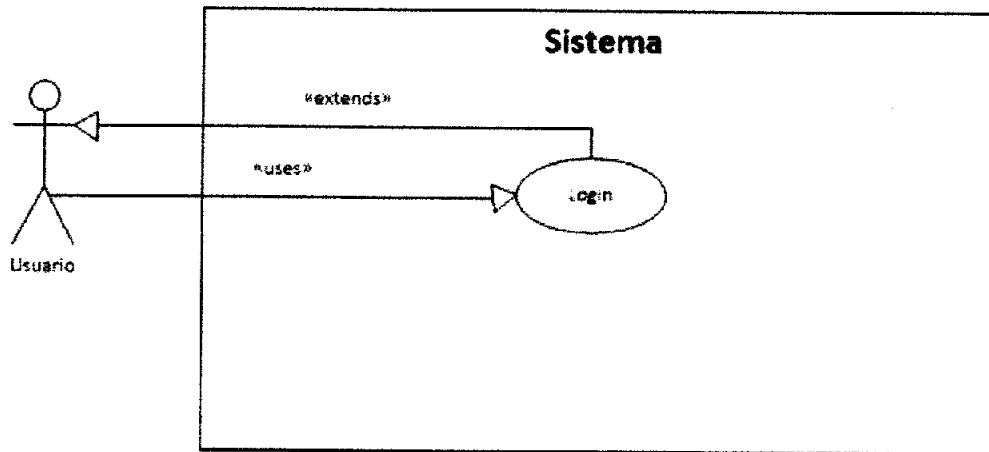
ID del Requerimiento:	RNF-004
Nombre del Requerimiento:	Generar Bitácora
Descripción:	El sistema debe realizar una bitácora de seguimiento para mantener un registro de actividades en el sistema
Prioridad:	Alta

ID del Requerimiento:	RNF-005
Nombre del Requerimiento:	Hardware
Descripción:	El sistema debe correr en el hardware existente.
Prioridad:	Alta

10.2. Casos de usos

10.2.1. Diagramas de Casos de uso

Caso de uso Login (Acceso al Sistema)



Caso de uso Login(Salir del Sistema)

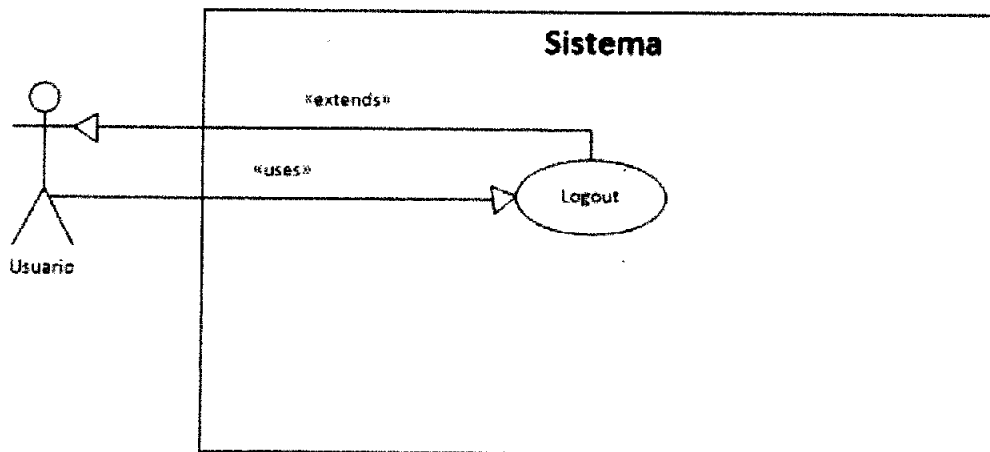


Diagrama de Casos de usos Sistema

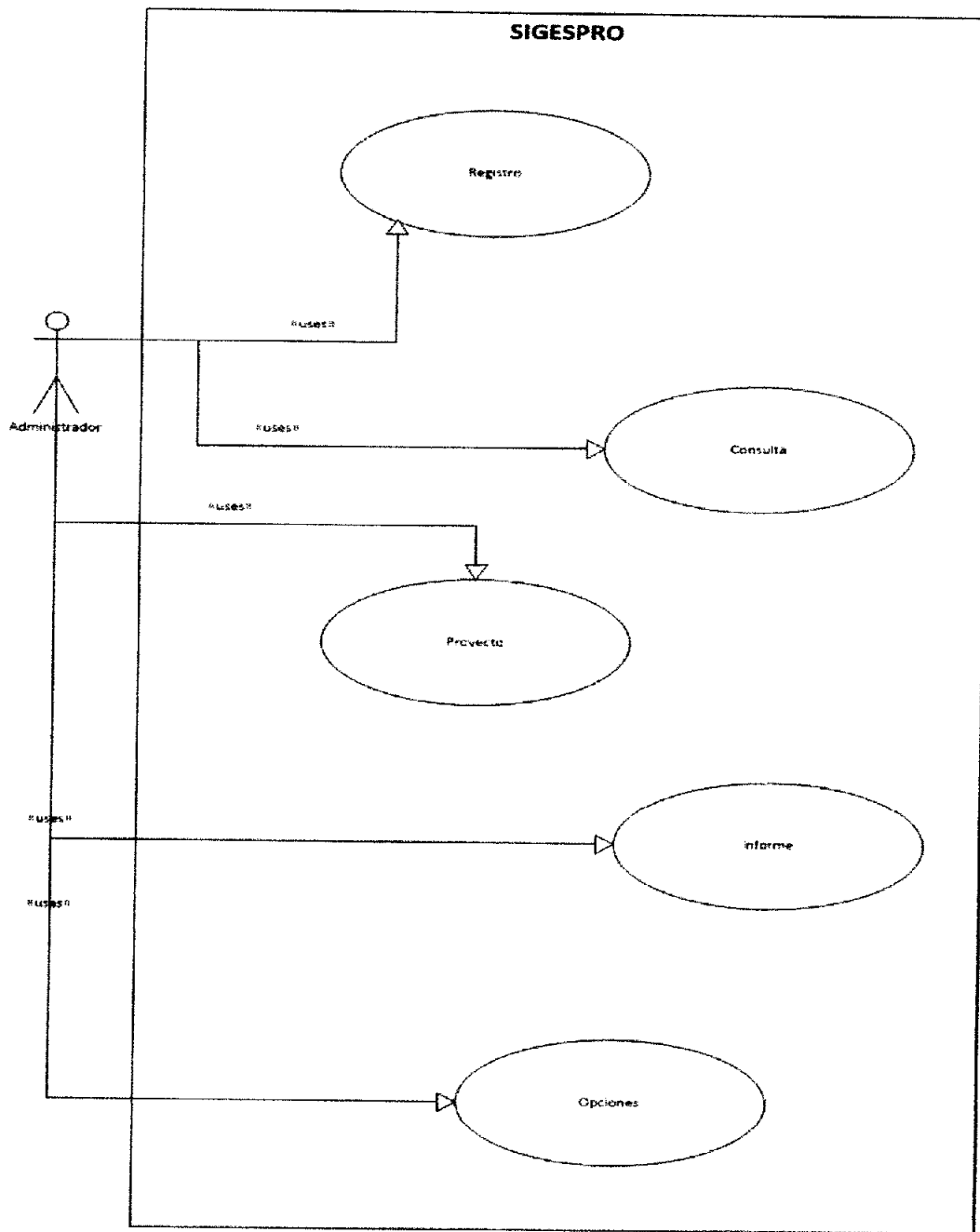


Diagrama de Casos de usos Registro

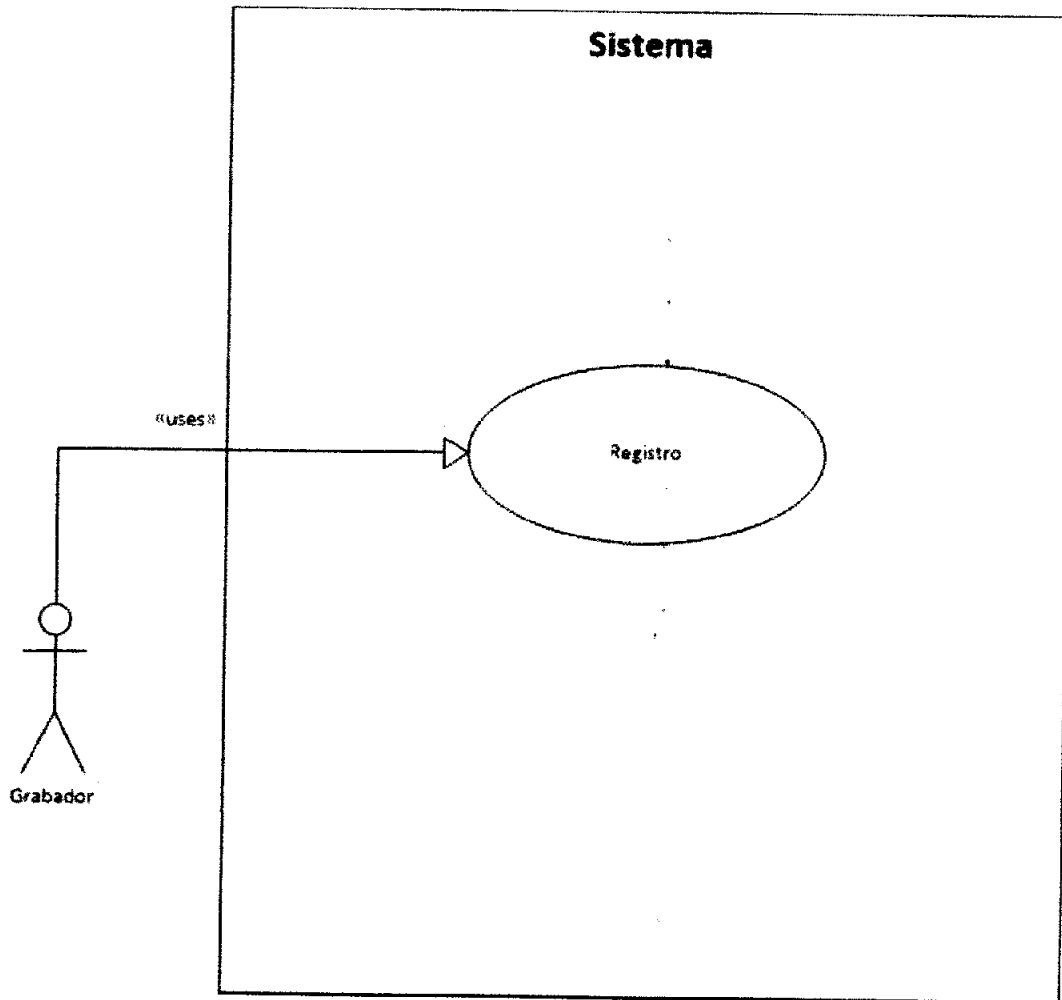


Diagrama de Casos de usos Proyecto - Consulta

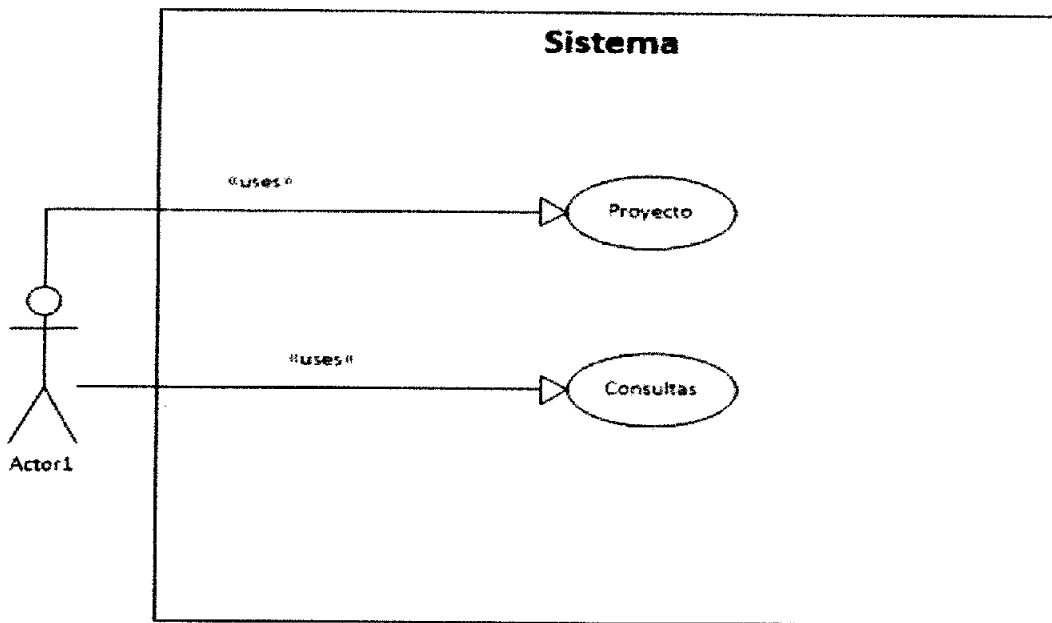


Diagrama de Casos de usos Área del cliente

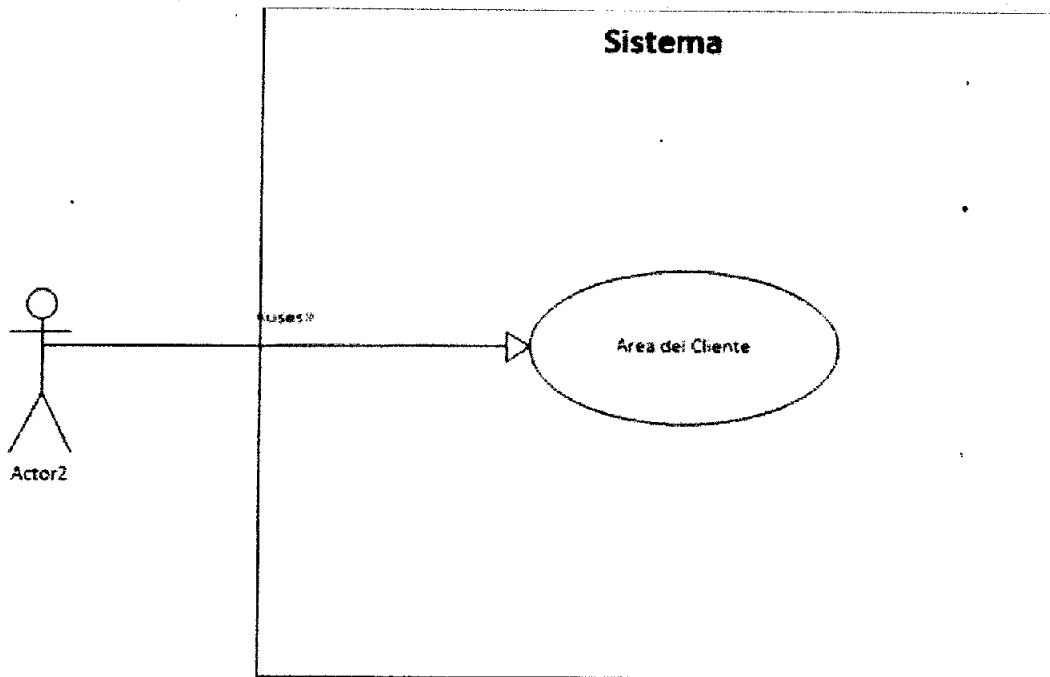


Diagrama de Casos de usos Área del cliente

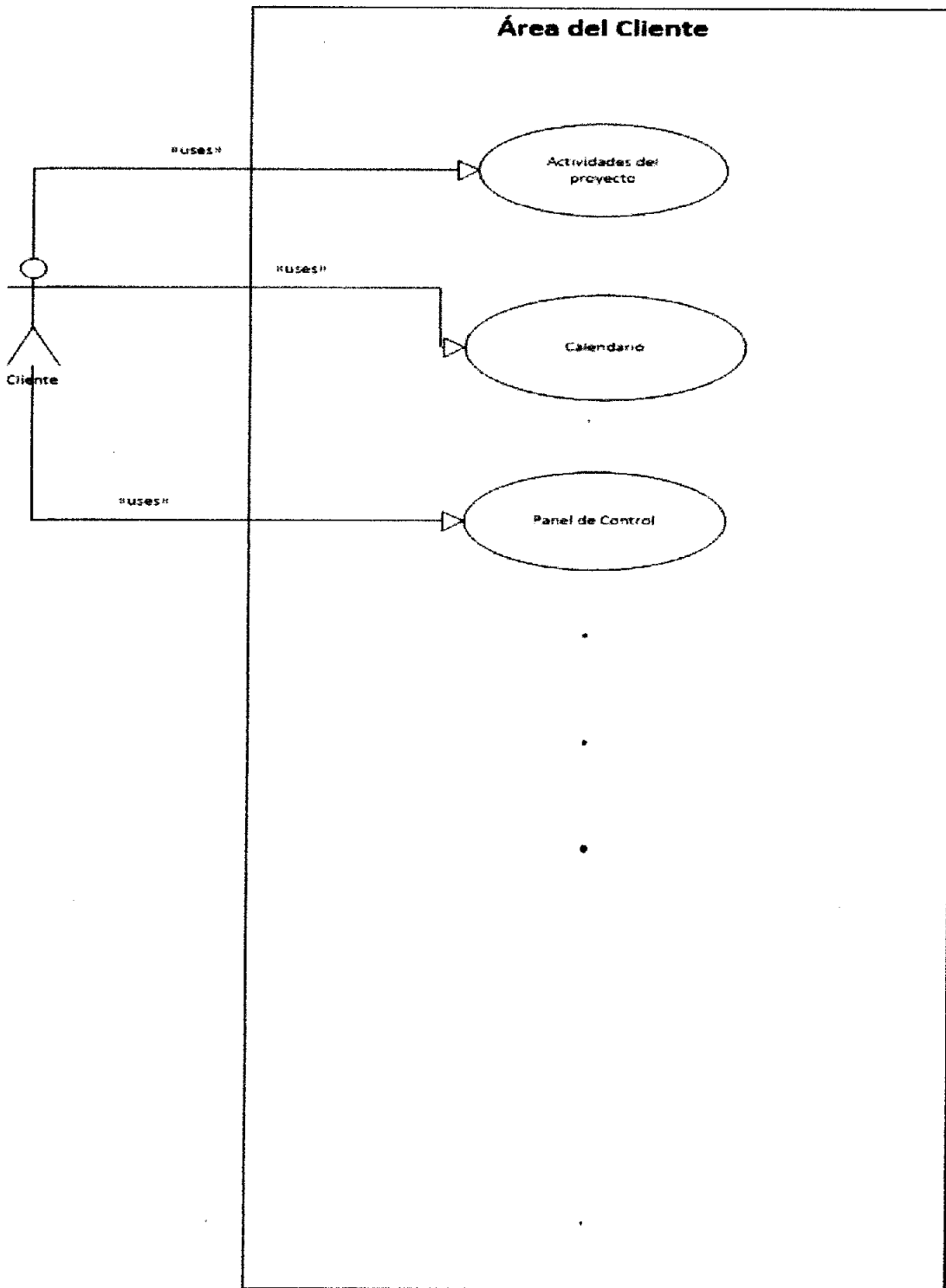
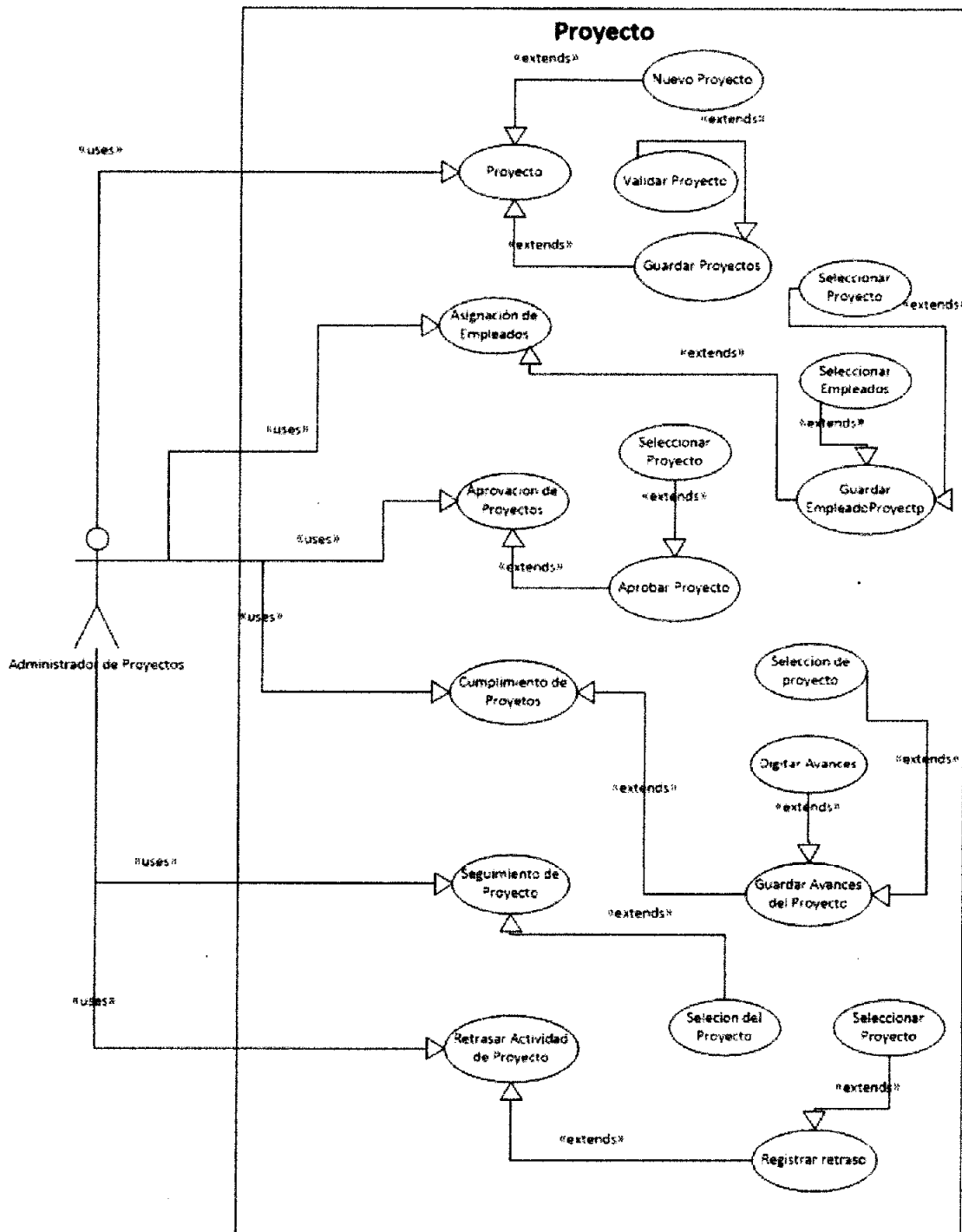
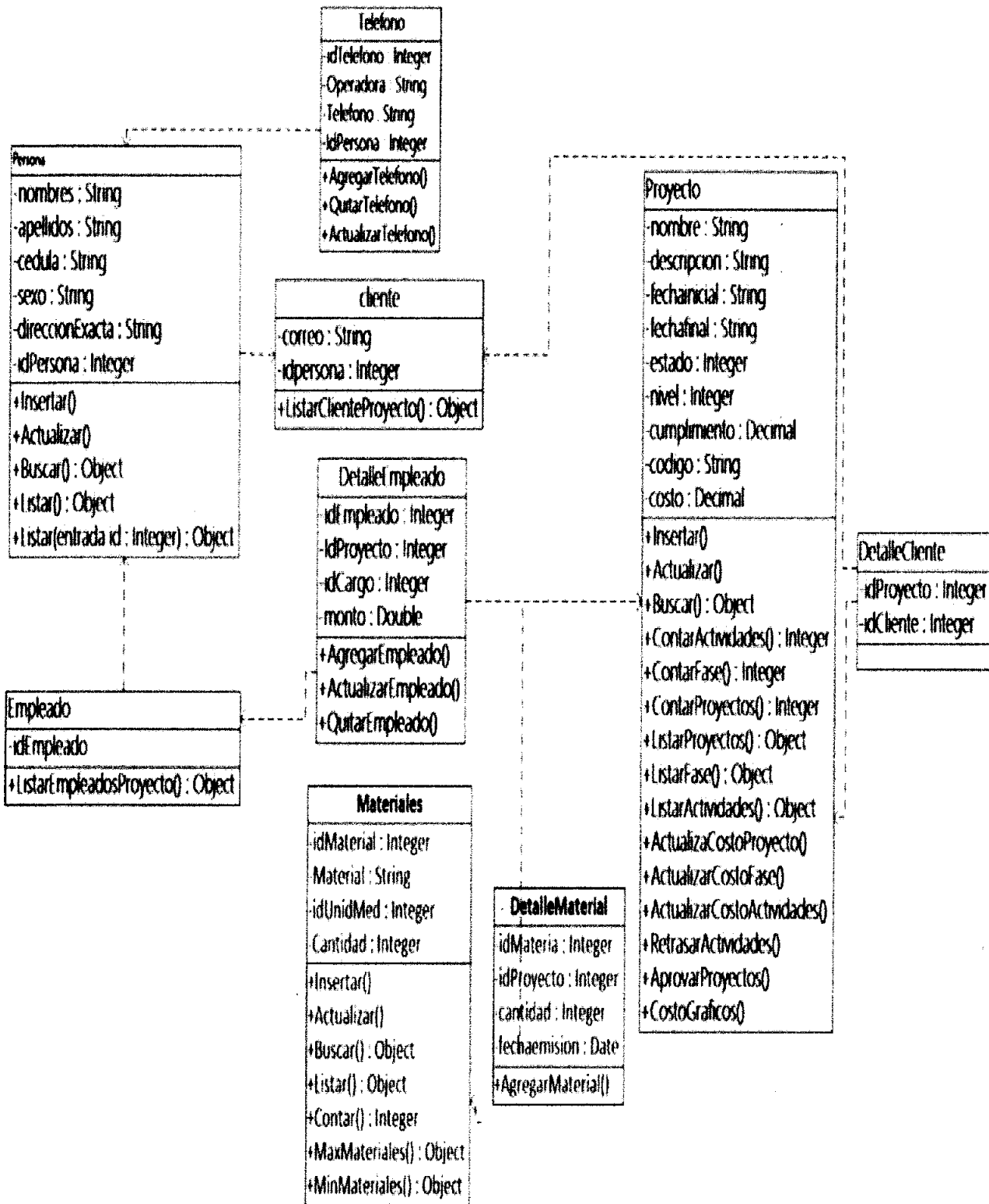


Diagrama de Casos de usos Proyecto



10.3. Diagrama de Clase



10.4. Fichas de Casos de uso

CU-0001	Proyecto	
Versión	V1 22/01/2015	
Autores	Administrador, Grabador	
Objetivos asociados	Registro de Proyecto	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso { concreto cuando <evento de activación> , abstracto durante la realización de los casos de uso <lista de casos de uso> }	
Precondición	<precondición del caso de uso>	
Secuencia	Paso	Acción
Normal	1	Presiona el botón de proyecto y se le presenta una ventana con un listado de proyecto junto con sus fases y actividades, donde usted podrá agregar un nuevo proyecto o editar la información.
Postcondición	Haberse logueado en el sistema	
Excepciones	Paso	Acción
	1	Ninguna
Rendimiento	Paso	Cota de tiempo
	1	2 segundos
Importancia	Alta	
Urgencia	Inmediatamente	
Comentarios	ninguno	

CU-0002	Nuevo Proyecto	
Versión	V1 22/01/2015	
Autores	Administrador, Grabador	
Objetivos asociados	Registro de Proyecto	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso { concreto cuando <evento de activación> , abstracto durante la realización de los casos de uso <lista de casos de uso>}	
Precondición	<precondición del caso de uso>	
Secuencia Normal	Paso	Acción
	1	Si desea agregar un nuevo elemento presionar en nuevo y se le presentara un formulario.
	2	Llenar el formulario y presionar guardar para que los registro sean guardados
	3	Se le cargara la información de ese registro y podrás modificar los datos que desees y luego presionar guardar para que los cambios sean guardados
Postcondición	Haberse logueado en el sistema	
Excepciones	Paso	Acción
	1	Si uno de los datos ingresado en los formularios del registro son errores se le enviara un aviso de error para que corrija la información ingresada
Rendimiento	Paso	Cota de tiempo
	1	3 segundos

	2	3 segundos
Importancia	Alta	
Urgencia	Inmediatamente	
Comentarios	ninguno	

CU-0003	Asignación de Empleados	
Versión	V1 24/01/2015	
Autores	Administrador, Grabador	
Objetivos asociados	Asignacion de Empleado	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso { concreto cuando <evento de activación> , abstracto durante la realización de los casos de uso <lista de casos de uso>}	
Precondición	<precondición del caso de uso>	
Secuencia	Paso	Acción
Normal	1	Si desea Asignar un nuevo empleado al proyecto presionar asignación de empleado y se le presentara un listado de proyectos.
	2	Seleccionar al empleado y el proyecto al cual desea asignarlo
	3	Se le cargara la información de ese registro y podrás modificar los datos que desees y luego presionar guardar para que los cambios sean guardados
Postcondición	Haberse logueado en el sistema, Haber registrado mínimo un proyecto, Haber registrado mínimo un empleado.	
Excepciones	Paso	Acción
	1	Si uno de los datos ingresado en los formularios del registro son errores se le enviara un aviso de error para que corrija la información ingresada
Rendimiento	Paso	Cota de tiempo
	1	3 segundos
	2	3 segundos
Importancia	Alta	
Urgencia	Inmediatamente	

Comentarios	ninguno
-------------	---------

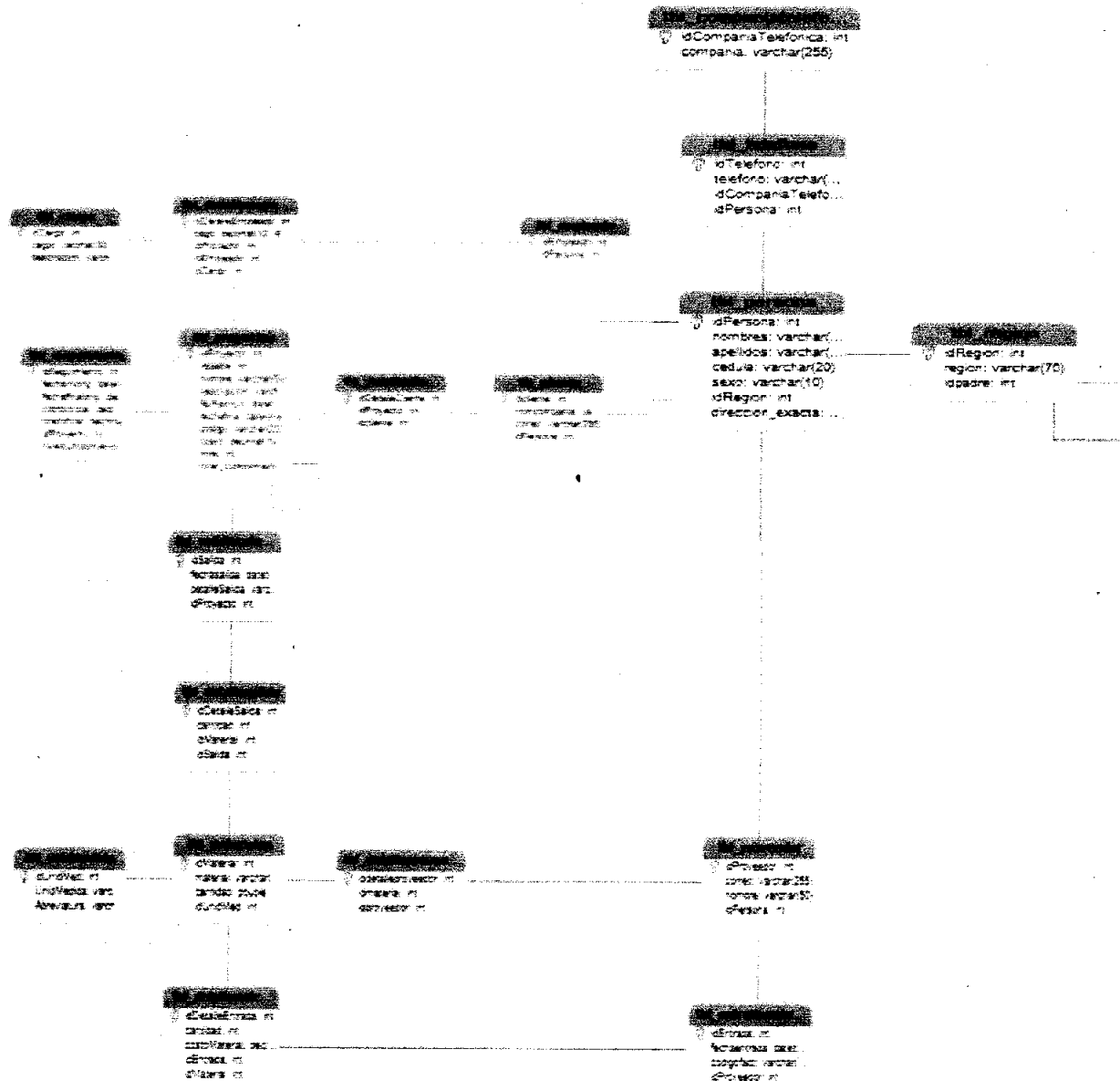
CU-0004	Aprobación de Proyectos	
Versión	V1 24/01/2015	
Autores	Administrador, Grabador	
Objetivos asociados	Asignación de Empleado	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso { concreto cuando <evento de activación> , abstracto durante la realización de los casos de uso <lista de casos de uso>}	
Precondición	<precondición del caso de uso>	
Secuencia	Paso	Acción
Normal	1	Si desea Aprobar un proyecto presionar Aprobación de proyecto y se le presentara un listado de proyectos sin aprobar.
	2	Seleccionar el proyecto que desea aprobar presionar guardar para que los cambios sean guardados
Postcondición	Haberse logueado en el sistema, Haber registrado mínimo un proyecto sin aprobar.	
Excepciones	Paso	Acción
	1	Ninguna
Rendimiento	Paso	Cota de tiempo
	1	3 segundos
	2	3 segundos
Importancia	Alta	
Urgencia	Inmediatamente	
Comentarios	ninguno	

CU-0005	Cumplimiento de Proyectos	
Versión	V1 24/01/2015	
Autores	Administrador, Grabador	
Objetivos asociados	Asignación de Empleado	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso { concreto cuando <evento de activación> , abstracto durante la realización de los casos de uso <lista de casos de uso>}	
Precondición	<precondición del caso de uso>	
Secuencia Normal	Paso	Acción
	1	Si desea Actualizar el porcentaje de cumplimiento de un proyecto presionar Cumplimiento de Proyecto y se le presentara un listado de proyectos y su porcentaje de cumplimiento.
	2	Seleccionar el proyecto que desea Actualizar su porcentaje de cumplimiento presionar guardar para que los cambios sean guardados
Postcondición	Haberse logueado en el sistema, Haber registrado mínimo un proyecto sin aprobar.	
Excepciones	Paso	Acción
	1	El porcentaje del proyecto no debe ser del 100%
Rendimiento	Paso	Cota de tiempo
	1	3 segundos
	2	3 segundos

Importancia	Alta
Urgencia	Inmediatamente
Comentarios	ninguno

10
11
12
13
14

10.5. Base de Datos



10.6. Interfaz

Se diseñaron 41 interfaces para la administración y desarrollo de nuestro sistema de gestión de proyectos; obteniendo así una interfaz amigable y dinámica para el usuario.

Listado de Proyectos existentes

Proyectos Existentes

Nombre	Proyecto	Descripción	Inicio	Fin	Costo
Administración de un autopercepción	Administración de un autopercepción				

Ventana de nuevo

Proyecto

Nombre del Proyecto Fecha de Inicio lunes , 22 de junio

Costo Fecha de Culminación lunes , 22 de junio

Descripción

Guardar

Fase

Nombre de la fase Fecha de Inicio lunes , 22 de junio

Asignar Proyecto Fecha de Culminación lunes , 22 de junio

Costo

Descripción

Actualizar

Actividad +

Nombre de la Actividad Fecha de Inicio lunes , 22 de junio

Asignar Fase Fecha de Culminación lunes , 22 de junio

Costo

Descripción

Actualizar

Listado de empleado en los proyectos

Administración de Empresas y Recursos Humanos

Empleado

Buscar

Empleado	Cargo	Departamento	Proyecto	Fecha de Inicio	Fecha de Culminación	Estado
Empleado	Cargo	Departamento	Proyecto	Fecha de Inicio	Fecha de Culminación	Estado
Empleado	Cargo	Departamento	Proyecto	Fecha de Inicio	Fecha de Culminación	Estado

Filtros

Proyecto

Construcción de un autosericio (Mas)

Fase

Actividad

100% de los datos de este sistema son confiables y están actualizados.

Nueva asignación de empleado

Administración de Empresas y Recursos Humanos

Empleado

Empleado

Asignación de Empleado

Proyecto

Construcción de un aut...

Fase

Actividad

Asignar

Listado de Empleados en la Actividad

Proyecto	Empleado	Asignar Cargo	Pago del Empleado

Listado de Cliente

Inicio > Gestión de Clientes > Listado de Clientes

Clientes Asignados a los proyectos

Buscar:

Filtros:

Proyectos: **Proyectos de los departamentos de...**

ID	NOMBRE	PROYECTO	ESTADO
1
2

Nueva Asignación de Cliente a proyectos

Administración de Clientes y Proyectos

Clientes Asignados a los proyectos: Datos del Cliente

Seleccionar la Actividad: Limpio

Detalle de Cliente

Listado de Clientes asignados al proyecto

CONTACTO	COMPANIA	PROYECTO	ESTADO	PERSONA
...

Actualizar | Eliminar | Imprimir

Listado de aprobación de proyecto

Administración de Clientes y Proyectos

Proyectos Aprobados: Aprobación de Proyectos

Búsqueda

Proyecto	Fecha de inicio	Fecha que finaliza	Costo (Miles de \$)	Estado (Inicio)
...

Nueva Aprobación de proyectos

Administración de Clientes y Proyectos

Proyectos Aprobados: Aprobación de Proyectos

Proyecto

Fecha de inicio

Fecha que finaliza

Costo (Miles de \$)

Estado (Inicio)

✓ Finalizar

Ventana de cumplimiento

Completado por: [Nombre]

Selección de Proyecto:
[Construcción de un autoserivicio (Masa)]

Selección de Fase:
[Planificación]

Selección de Actividad:
[Empaquetado]

Nivel de cumplimiento planificado:
[100%]

Nivel de cumplimiento ejecutado:
[0%]

Guardar

Ventana de retraso de actividades

Retraso de Actividades

Selección de Proyecto	Inicio y Fin de Actividades (Planned)
[Construcción de un autoserivicio (Masa)]	[Inicio: 01/01/2015 - Fin: 01/01/2015]
Selección de Fase	[Inicio y Fin de la Actividad (Planned)]
[]	[Inicio: 01/01/2015 - Fin: 01/01/2015]
Selección de Actividad	[Inicio y Fin de Actividad]
[]	[Inicio: 01/01/2015 - Fin: 01/01/2015]
Selección de Actividad	[]
[]	[]

Guardar

Area del Cliente online

Bienvenido al panel de Cooci



Usuario



Contraseña

Recordar contraseña

Entrar

Panel de Usuario

Inicio

Inicio

Inicio

Inicio

Introducción

Inicio

Inicio

Construcción de un autoservicio (Masaya)

Inicio

Inicio

Inicio

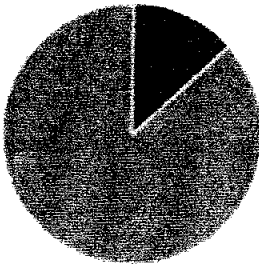
Inicio

Inicio

Inicio

Inicio del Proyecto

Inicio



Actividades del proyecto online

InfoPro Inicio Inicio Inicio

Actividades del Proyecto

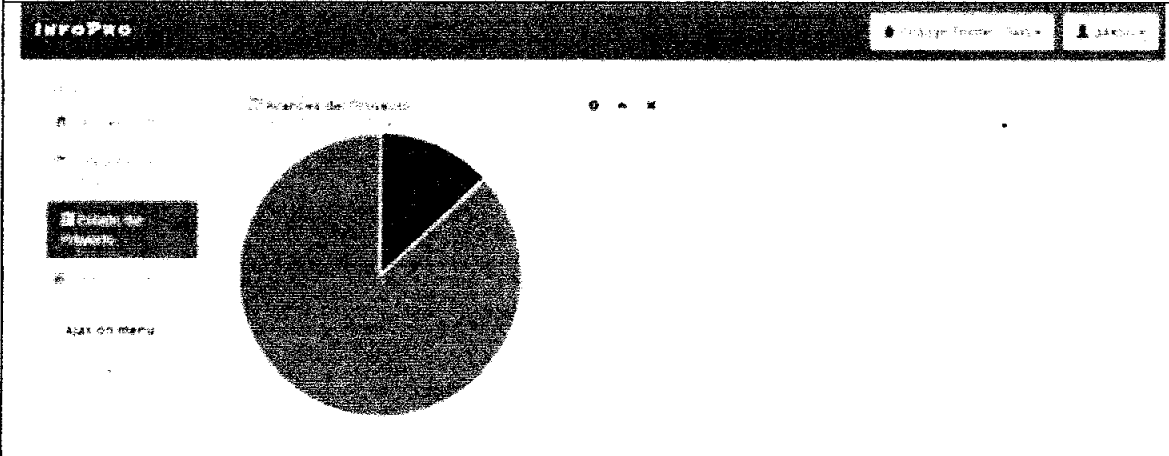
15 records per page

Search

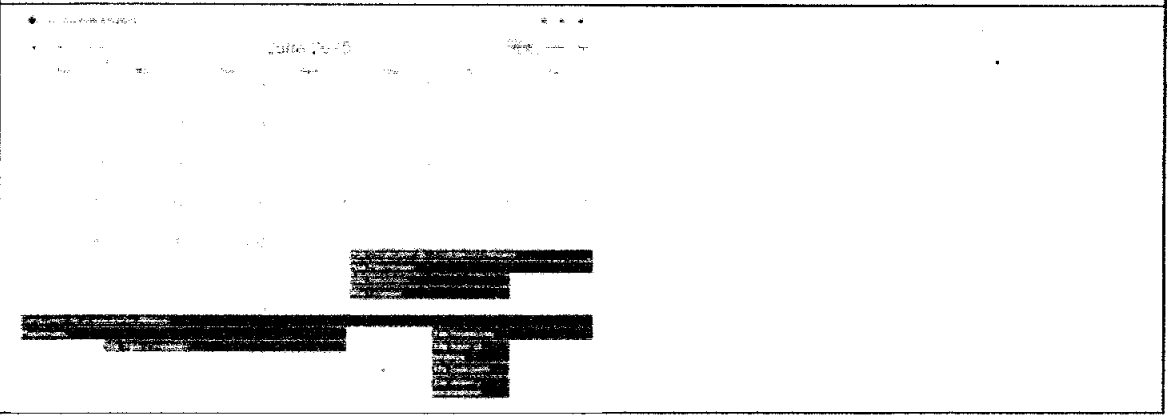
Actividad	% Completado	Inicio de la Actividad	Terminación de la Actividad	Fase	Proyecto
Alcance de recursos	75% Completado	11/02/15	11/02/15	2. planeación	Creación de un subproyecto (Mozambique)
Asignación de recursos	75% Completado	11/02/15	11/02/15	4. ejecución	Creación de un subproyecto (Mozambique)
Identificación	75% Completado	11/02/15	11/02/15	3. monitoreo	Creación de un subproyecto (Mozambique)
Control de cambios	75% Completado	11/02/15	11/02/15	5. cierre	Creación de un subproyecto (Mozambique)
Comunicación	75% Completado	11/02/15	11/02/15	6. cierre	Creación de un subproyecto (Mozambique)

Invest:001/croci/page id = 9

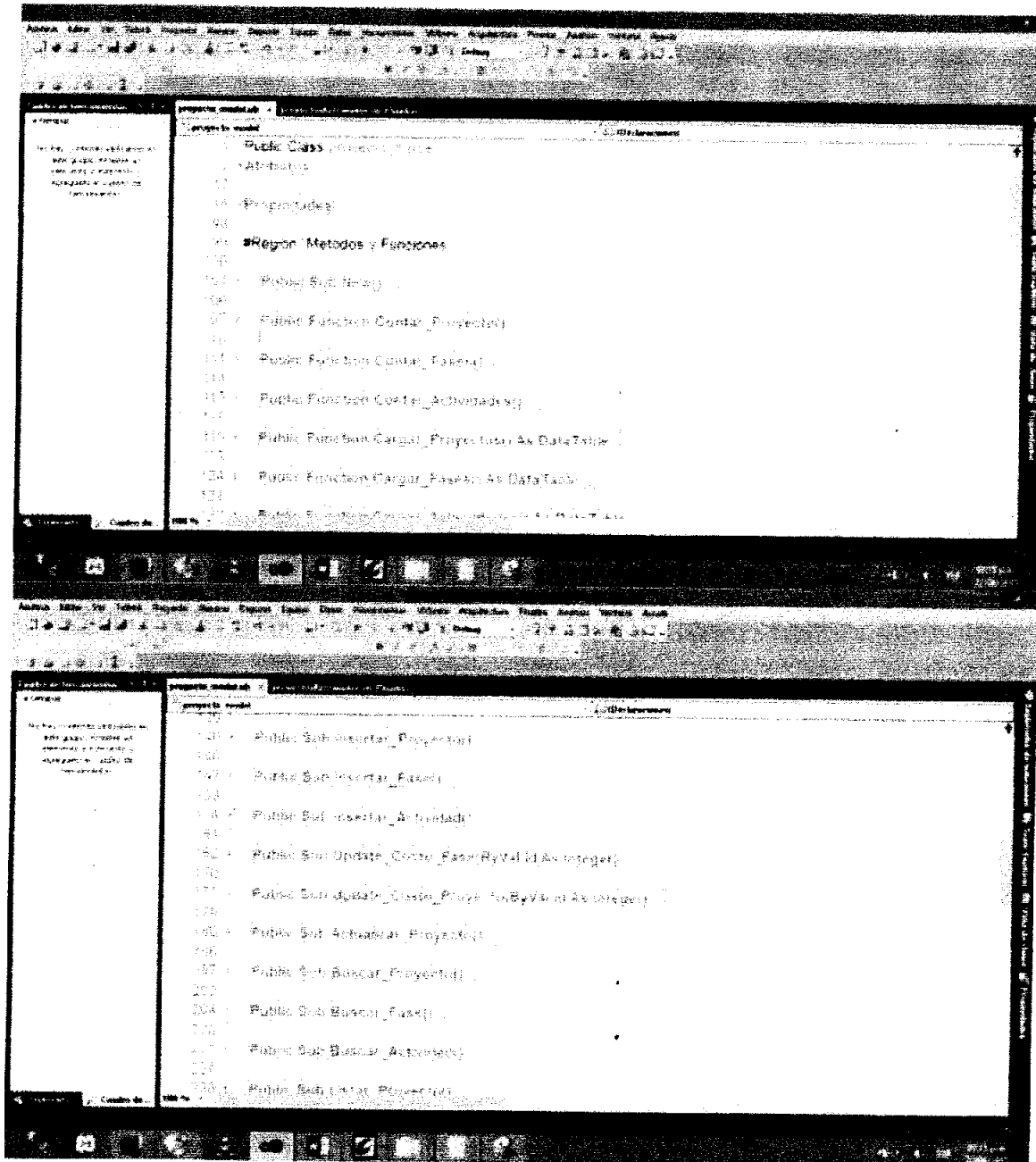
Porcentaje de avances del proyecto



Calendario de Actividades



Entre los principales Procesos se encuentran los siguientes métodos:



```
Public Sub Borrar_Fax()
End Sub

Public Sub Borrar_Aprobado()
End Sub

Public Sub Borrar_Proyectos()
End Sub

Public Function TotalFasesPorProyecto() As DataTable
End Function

Public Function Total_BienesPorProyecto() As DataTable
End Function

Public Function ProyectosAprobados() As DataTable
End Function

Public Sub Aprobar()
End Sub

Public Function ListaProyectosAprobados() As DataTable
End Function

Public Function ProyectosPlanificados() As DataTable
End Function

Public Function ProyectosEjecutandose() As DataTable
End Function

Public Function ComplementoActividad() As Double
End Function

Public Sub UpdateComplemento()
End Sub

Public Sub UpdateComplementoFase(ByVal idProyecto As Integer, ByVal Fase As Integer)
End Sub

Public Sub UpdateComplementoProyecto(ByVal idProyecto As Integer)
End Sub

Public Function TotalComplemento() As Integer
End Function
```

```
Public Function ProyectosPorParametro() As DataTable
End Function

Public Function ProyectosEjecutandose() As DataTable
End Function

Public Function ComplementoActividad() As Double
End Function

Public Sub UpdateComplemento()
End Sub

Public Sub UpdateComplementoFase(ByVal idProyecto As Integer, ByVal Fase As Integer)
End Sub

Public Function TotalProyectos() As Integer
End Function

Public Sub GestionarPorValParametro() As Object()
End Sub

Public Sub GestionarPorParametro(ByVal parametro As Object)
End Sub

Public Sub CompletarProyectos(ByVal parametro As Integer)
End Sub

Public Function ProyectosEjecutandose() As DataTable
End Function

Public Sub RegistrarActividad(ByVal nombre As String, ByVal fecha As Date, ByVal estado As Integer)
End Sub

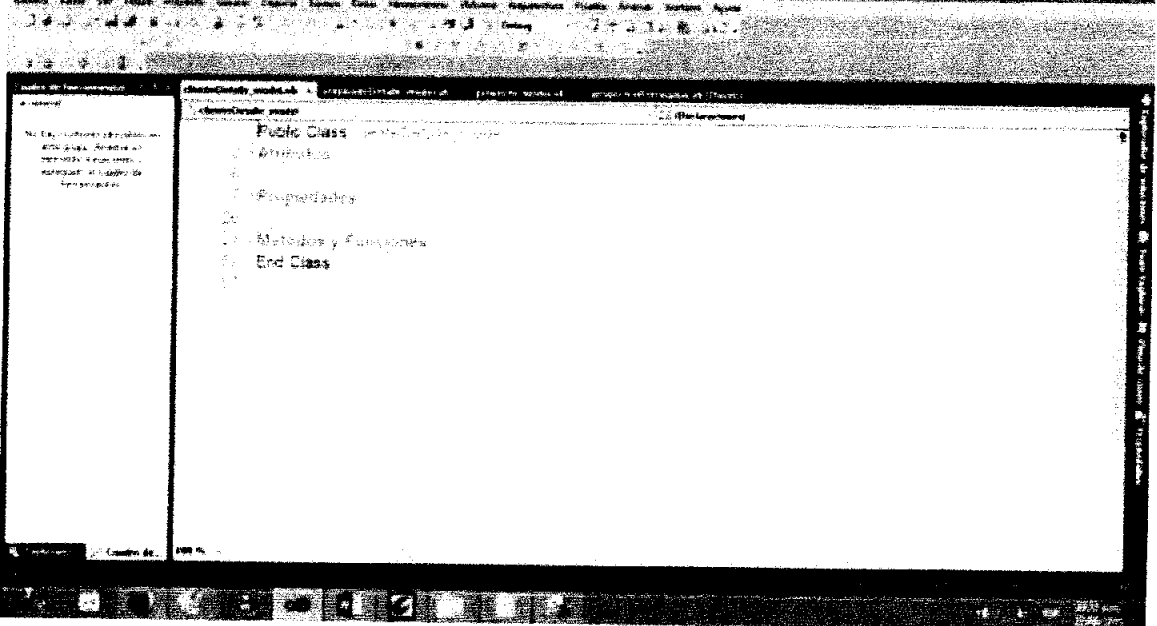
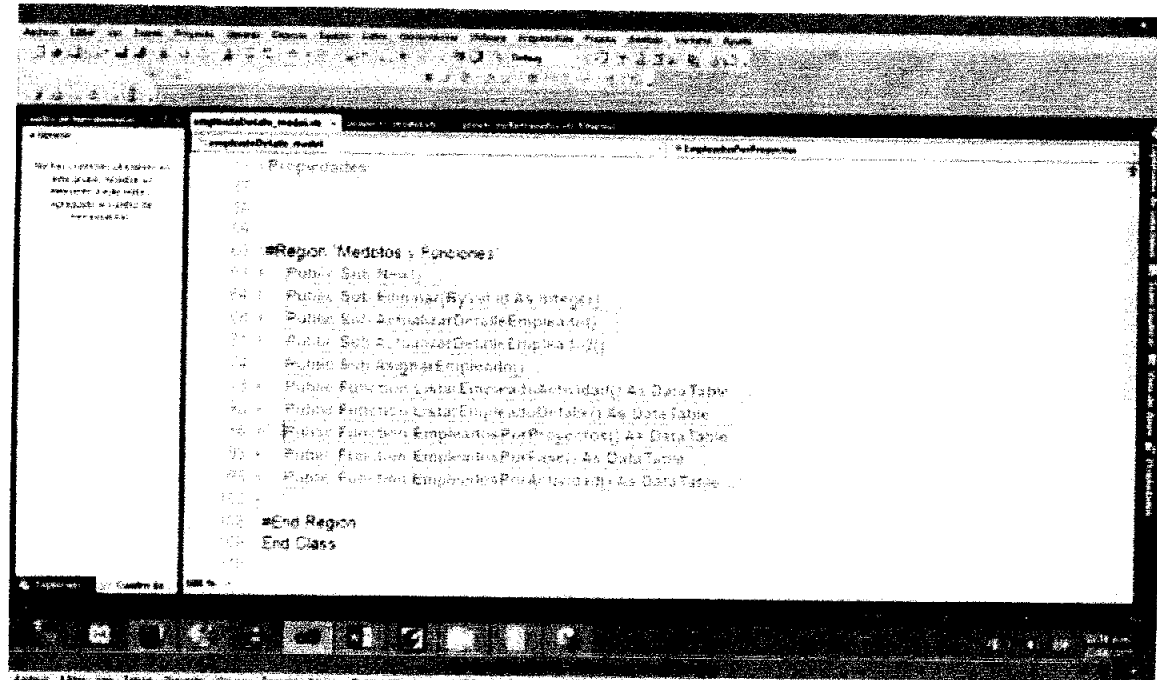
Public Function ActividadRegistrada() As Integer
End Function

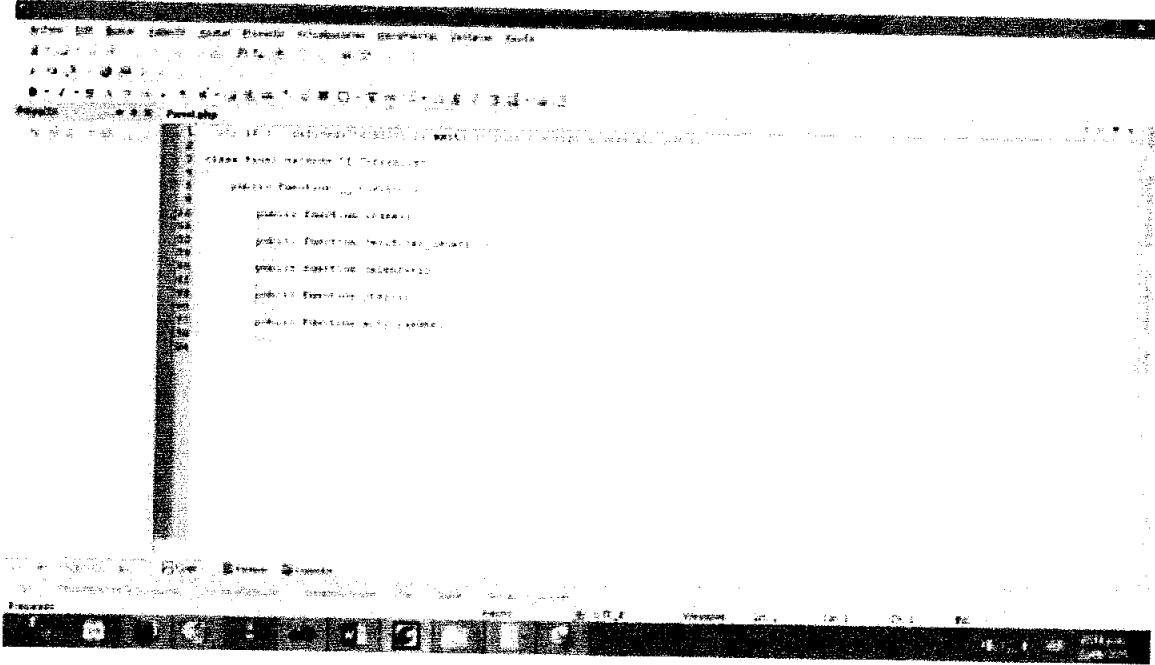
Public Function ActividadRegistradaPorActividad(ByVal parametro As String) As DataTable
End Function

Public Function ActividadRegistradaPorActividad(ByVal parametro As String) As Double
End Function

End Region

INSERT INTO tbl_proyectos (nombre, descripcion, referencia, fecha, estado) VALUES
```





XI. Conclusiones

Mediante la determinación y descubrimiento de los procesos principales del sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015 hemos logrado la coordinación necesaria y adecuada para el desarrollo de cada una de las ejecuciones a ser realizadas en el proyecto; este proyecto consta de 10 procesos principales y 5 no funcionales.

Gracias a la utilización de diferentes herramientas de trabajo (framework) como Visual Studio 2010, APTANA Studio y Neatbean para el diseño de las interfaces de usuario para el sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015; hemos logrado brindar al usuario una interfaz sencilla, amigable y de fácil manejo; además de que permite un mejor manejo y control sobre las tareas o actividades que se encuentran en asignación o ejecución.

Mediante a la aplicación del Modelo Vista Controlador (MVC) como patrón de desarrollo de las rutinas del sistema de gestión de proyectos de construcción para la cooperativa de obras civiles COOCI, en el año 2015; se logró alcanzar un 85 % del proyecto gracias a la finalización de la mayoría de los requerimientos exigidos por el cliente tales como: asignación de proyecto, retraso de proyecto, planificación de proyecto, actividades pendientes del proyecto, etc. el otro 15 % es quedado pendiente debido a que el cliente posee una aplicación para el manejo de bodega de cada proyecto de construcción realizado por la empresa.

XII. Recomendaciones

Para el correcto funcionamiento y ejecución de este sistema se brindan las siguientes recomendaciones:

Contar con los equipos y software necesarios para la ejecución del sistema:

Aplicación-escritorio

- PC de 32 bit o 64 bit.
- SO (Windows seven, o versiones superiores).
- WamppServer si la aplicación no se conectara online.
- Conexión a internet o red local para la conexión de la base de dato remota.

Aplicación-web

- PC
- Conexión a internet
- Contar con cualquiera de los siguientes navegadores para disfrutar la amigable interfaz : Mozilla, Google Chrome, Opera.

XIII. Anexos

13.1. Formato de planificación de proyectos

Formato de planificación de proyectos

		Fecha de inicio	Fecha Final
Nombre del proyecto			
Descripción:			
Fases del proyecto	Fecha de inicio	Fecha Final	

		Fecha de inicio	Fecha Final
Nombre de la fase			
Descripción:			
Actividades de la Fase	Fecha de inicio	Fecha Final	

		Fecha de inicio	Fecha Final
Nombre de la Actividad			
Descripción:			
Emprendedor	Fecha de inicio	Fecha Final	

13.2. Ejemplo de formato de planificación de proyectos

	Fecha de inicio: 15/01/2015	Fecha final:06/06/2015
Nombre del proyecto:	Pliego de bases y condiciones remodelación y ampliación de edificio de servicios consulares	
Descripción:		
Fases del proyecto	Fecha de inicio	Fecha final
Preliminares	15/01/2015	30/01/2015
Fundaciones	01/02/2015	15/02/2015
Estructura	14/02/2015	22/02/2015
Paredes	22/02/2015	04/03/2015
Techo y fascias	04/03/2015	15/03/2015
Acabados	16/03/2015	19/03/2015
Cielos	20/03/2015	30/03/2015
Pisos	01/04/2015	09/04/2015
Particiones	10/04/2015	18/04/2015
Muebles fijos	19/04/2015	27/04/2015
Puertas	27/04/2015	30/04/2015
Ventanas	01/05/2015	05/05/2015
Obra hidrosanitarias	06/05/2015	18/05/2015
Electricidad	19/05/2015	27/05/2015
Obras exteriores	26/05/2015	30/05/2015
Pinturas	01/06/2015	05/06/2015

	Fecha de inicio: 15/01/2015	Fecha Final: 30/01/2015
Nombre de la fase:	Preliminares	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Limpieza inicial	15/01/2015	20/01/2015
Champa y obras temporales	20/01/2015	23/01/2015
Demoliciones	23/01/2015	26/01/2015
Trazo y nivelación	26/01/2015	30/01/2015

		Fecha de inicio: 01/02/2015	Fecha Final: 15/02/2015
Nombre de la fase:	Fundaciones		
Descripción:			
Actividades de la Fase		Fecha de inicio	Fecha final
Excavación estructural		01/02/2015	04/02/2015
Relleno y compactación		05/02/2015	10/02/2015
Desalojo de material de excavación		11/02/2015	12/02/2015
Concreto 3000 psi (mezcla y fundición)		12/02/2015	15/02/2015

		Fecha de inicio: 14/02/2015	Fecha Final: 22/02/2015
Nombre de la fase:	Estructura		
Descripción:			
Actividades de la Fase		Fecha de inicio	Fecha final
Acero de refuerzo (armado y colocado)		14/02/2015	17/02/2015
Formaleta de vigas y columnas		18/02/2015	19/02/2015
Concreto 3000 PSI (Mezcla y fundación)		20/02/2015	21/02/2015
Columnas metálicas (instalada y pintadas)		20/02/2015	22/02/2015

		Fecha de inicio: 22/02/2015	Fecha Final: 04/03/2015
Nombre de la fase:	Paredes		
Descripción:			
Actividades de la Fase		Fecha de inicio	Fecha final
Pared de mampostería confinada 6"X8"X16"		22/02/2015	25/02/2015
Pared de panel covitec semiestructural, chilateada y repellada		25/03/2015	04/03/2015

	Fecha de inicio: 04/03/2015	Fecha Final: 15/03/2015
Nombre de la fase:	Techos y fachas	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Estructura metalica de techos	04/03/2015	05/03/2015
Cubiertas de zinc	06/03/2015	09/03/2015
Flashing y canales de zinc	10/03/2015	11/03/2015
Fascia de denglass	11/03/2015	12/03/2015
Alero de lámina MR (de más de 50cm de ancho)	12/03/2015	13/03/2015
Losa de concreto (repellada e impermeabilizada en ambas cara)	13/03/2015	14/03/2015
Losa de acceso de panel covitec	14/03/2015	14/03/2015
Gárgolas de concreto	15/03/2015	15/03/2015

	Fecha de inicio: 16/03/2015	Fecha Final: 19/03/2015
Nombre de la fase:	Acabados	
Descripción:		
Actividades de la Fase	Fecha de Inicio	Fecha final
Piqueteo de vigas y columnas	16/03/2015	17/03/2015
Repello corriente	17/03/2015	17/03/2015
Fino y arenillado con cal hidratada	18/03/2015	18/03/2015
Enchape de azulejos	18/03/2015	18/03/2015
Roda pie de cerámica	18/03/2015	19/03/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Cielos	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Cielo de gypsum regular con acabado liso, estructura de aluminio	20/03/2015	23/03/2015
Cielo de lámina MR con acabado liso, estructura de aluminio	23/03/2015	26/03/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Particiones	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Particiones de gypsum regular o denglass, forro ambas caras	26/03/2015	30/03/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Muebles fijos	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Mueble de cocina	31/03/2015	05/04/2015
Mueble de tesorería	06/04/2015	11/04/2015
Mueble de lavamanos publico		
Divisiones modulares para inodoros de baños publico	11/04/2015	17/04/2015
Espejo de baños públicos	17/04/2015	23/04/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Puertas	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Puerta tipo 1	23/04/2015	23/04/2015
Puerta tipo 2,3 y 7	23/04/2015	24/04/2015
Puerta Tipo 3	24/04/2015	24/04/2015
Puerta tipo 4	25/04/2015	25/04/2015
Puerta tipo 5 y 6	25/04/2015	26/04/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Paredes	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Ventanas de aluminio y vidrio	26/04/2015	30/04/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Obras hidrosanitarias	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Tubería y accesorios de agua potable	02/05/2015	11/05/2015
Tubería y accesorios de aguas negras	02/05/2015	15/05/2015
Aparatos sanitarios con sus accesorios	15/05/2015	17/05/2015
Trampa de grasa	17/05/2015	20/05/2015
Grifería	18/05/2015	20/05/2015
Prueba de presión	20/05/2015	25/05/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Electricidad	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Canalización eléctrica, del TEL y RED	25/05/2015	31/05/2015
Alambrado eléctrico	29/05/2015	05/06/2015
Panel Eléctrico y sus accesorios	03/06/2015	05/06/2015
Lámparas y accesorios eléctricos	04/06/2015	05/06/2015
Acometida	06/06/2015	09/06/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Obras exteriores	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Andenes de concreto E=2"	10/06/2015	20/06/2015
Rampa	20/06/2015	30/06/2015

	Fecha de inicio	Fecha Final
Nombre de la fase:	Pinturas y Limpieza	
Descripción:		
Actividades de la Fase	Fecha de inicio	Fecha final
Pintura acrílica	30/06/2015	05/07/2015
Limpieza regular	05/07/2015	10/07/2015
Limpieza final	10/07/2015	15/07/2015

XIV. Bibliografía

Bibliografía

- Codelgniter. (2012). Guía del Usuario en Español. España: Autoedición.
- Daniel Valdivieso. (14 de 06 de 2007). sistemas3.wordpress.com. Recuperado el 03 de 10 de 2013, de sistemas3.wordpress.com:
<http://sistemas3.wordpress.com/2007/06/14/Ingenieria-web/>
- Ing. Edwin Joao Merchán Carreño. (13 de 06 de 2011). blogereducativo.wordpress.com. Recuperado el 03 de 10 de 2013, de blogereducativo.wordpress.com:
<http://blogereducativo.wordpress.com/2011/09/06/ventajas-y-desventajas-de-utilizar-s-i/>
- Jorge Sánchez. (2004). Diseño Conceptual de Bases de Datos (guía de aprendizaje). California, 94305, USA.: Creative Commons.
- Kendal y Kendal. (2009). Análisis y Diseño de Sistemas de Información 6ta Edición. Prentice-Hall.
- sheyla88. (02 de Marzo de 2005). <http://sheyla88.blogspot.es/>. Recuperado el 05 de Mayo de 2014, de <http://sheyla88.blogspot.es/>: <http://sheyla88.blogspot.es/>
- Wikipedia. (05 de 05 de 2012). wikipedia.org. Recuperado el 05 de 05 de 2014, de <http://es.wikipedia.org/>: http://es.wikipedia.org/wiki/Base_de_datos
- Wikipedia. (15 de Junio de 2012). wikipedia.org. Recuperado el 02 de Mayo de 2014, de es.wikipedia.org:
http://es.wikipedia.org/wiki/Sistema_de_informaci%C3%B3n
- wikipedia. (7 de Mayo de 2014).
<http://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>.
Recuperado el 16 de Mayo de 2014, de
<http://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>.
<http://www.vidadigital.net/> <http://ayudawordpress.com/> <http://es.wordpress.org/>