



INSTITUTO POLITÉCNICO NACIONAL
Escuela Superior de Ingeniería Mecánica y Eléctrica



SISTEMA DE ADMINISTRACIÓN MÚLTIPLE (SAM)

TESIS

**QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMUNICACIONES Y ELECTRÓNICA**

PRESENTAN:

- ❖ Canales López Erika
- ❖ Martínez Ricaño Antonio

ASESORES:

- ❖ Ing. Catalina Patiño Gallegos
- ❖ M. en C. Gregorio García Pérez

México, DF. Mayo 2010

AGRADECIMIENTOS

A Dios, porque con él todo lo puedo, sin él nada. Por el regalo de la vida, por todo lo que me ha dado, por permitirme llegar hasta aquí, por ayudarme a alcanzar esta meta y por guiar cada uno de mis pasos.

A mis padres, por su confianza y su apoyo en mis años de estudios, por todas las enseñanzas que no se obtienen en ninguna escuela, que me inculcaron con amor, gracias por la comprensión que siempre me brindaron.

A mis hermanas Mary, Fer, Ale y a mis sobrinos, por su apoyo incondicional, por estar siempre para mí, por su voto de confianza y sobre todo por creer en mí.

A la profesora Catalina Patiño Gallegos, por su paciencia, por su orientación y asesoría para el desarrollo de la misma, por su interés y disposición en todo momento.

A mi profesor Gregorio García Pérez, por su apoyo, por sus consejos, por la asesoría para la elaboración de la presente y lo más importante por su ayuda desinteresada.

A Humberto que siempre ha estado a mi lado apoyándome, brindándome su cariño, aconsejándome, simplemente gracias por todo, TA.

Finalmente a todas las personas que se cruzaron en mi camino y que siempre me dieron palabras de aliento y apoyo.

ÍNDICE GENERAL

Índice de tablas	4
Índice de figuras	5
Glosario de términos	6
Introducción	8
Justificación	11
Objetivo General	13
Objetivos Específicos	14
I. Marco Teórico y Metodológico	
1.1 Ciclo de vida del desarrollo de sistemas	16
1.1.1 Identificación de problemas, oportunidades y objetivos	17
1.1.2 Determinación de los requerimientos de información	17
1.1.3 Análisis de las necesidades del sistema	17
1.1.4 Diseño del sistema recomendado	18
1.1.5 Desarrollo y documentación del software	19
1.1.6 Pruebas y mantenimiento del sistema	19
1.1.7 Implementación y evaluación del sistema	19
1.2 Modelo Conceptual	19
1.2.1 Modelo cliente-servidor	20
II. Análisis y Diagnóstico	
2.1 Análisis y diagnóstico del sistema actual	23
2.1.1 Técnicas para la obtención de información	24
2.1.2 Identificación de las necesidades de información	24
2.1.3 Tabla Sistemática	26
2.2 Conocimiento de Procesos en Ollin Studio	27
2.3 Forma de trabajo en Ollin Studio	31
2.4 Modelo entidad-relación	32
2.4.1 Limitantes de Mapeo	33
2.4.2 Cardinalidades de mapeo	33
2.4.3 Cardinalidades de mapeo para conjuntos binarios de relaciones	33
2.5 Diagrama entidad-relación SAM	36
2.5.1 Descripción de relaciones	39

2.6 Entidades SAM	39
2.7 Llaves	41
2.8 Recursos de almacenamiento para la administración de datos	42
2.8.1 Herramientas para la Administración del Acervo Digital	43
2.9 Propuesta general de solución	44

III. Diseño y construcción

3.1 Estructura de datos	47
3.2 Diseño preliminar	49
3.2.1 Asignación de nombre de los archivos	49
3.2.2 Descripción de los archivos	50
3.2.3 Requerimientos de personal	50
3.2.4 Análisis económico (Costos y beneficios del sistema)	50
3.3 Diseño de salida	51
3.3.1 Diseño lógico de salidas	51
3.4 Diseño de Entradas	51
3.5 Creación de bases de datos	54
3.5.1 Estructura de modelo relacional para SAM	55
3.5.2 Arquitectura General de SAM	56
3.5.3 Servidor Central	57
3.5.4 Servicio de Administración de Metadata (SAM)	57
3.5.5 Clientes SAM	58
3.6 Desarrollo	59
3.6.1 Módulos de Software	59
3.6.2 Desarrollo del Núcleo del Servidor	60
3.6.3 Recepción de EDL's y Creación de la estructura de Carpetas	60
3.6.4 Herramientas para el Data Wrangler	60
3.6.5 Utilidades para el Northlight	61
3.6.6 Administración de Vfx	61
3.6.7 Utilidades para Ilusión	61
3.6.8 Administración de Proyectos	62
3.6.9 Generación de Estadísticas	62
3.6.10 Importación y Exportación a Extranet	62

IV. Implementación y Operación

4.1 Modelo de Interfaz	64
4.2 Interfaz de Usuario	64
4.3 Menú Principal	67
4.4 Captura y Edición de Proyectos y Archivos	68
4.5 Administración de Usuarios y clientes	72
Conclusiones	80
Bibliografía	82

ÍNDICE DE TABLAS

Tabla 1.- Tabla Sistemática	26
Tabla 2.- Descripción de Entidad Proyectos	39
Tabla 3.- Descripción de Entidad Usuarios	40
Tabla 4.- Descripción de Entidad Perfil_maestro	41
Tabla 5.- Descripción de Entidad Archivos	41
Tabla 6.- Principales recursos de almacenamiento	42
Tabla 7.- Departamentos responsables de la administración del espacio en los servidores.	43
Tabla 8.- Consulta estatus del proyecto	69
Tabla 9.- Consulta para modificar usuario	75
Tabla 10.- Conclusiones (Comparativo antes y después del sistema SAM)	80

ÍNDICE DE FIGURAS

Figura 1.- Estructura Orgánica de la Empresa Ollin Studio	10
Figura 2.- Las siete fases del ciclo de vida para el desarrollo de sistemas.	16
Figura 3.- Presentación de Páginas PHP	21
Figura 4.- Flujo de trabajo en Ollin Studio	31
Figura 5.- Diagrama Entidad-Relación	38
Figura 6.- Esquema de la relación entre un elemento-dato y una estructura de datos.	48
Figura 7.- Modelos relacional de SAM	55
Figura 8.- Arquitectura General de SAM	56
Figura 9.- Interfaz Principal	65
Figura 10.- Menú	67
Figura 11.- Alta de proyecto	68
Figura 12.- Edición de proyecto	69
Figura 13.- Eliminar Proyectos	70
Figura 14.- Subir Archivo(s)	71
Figura 15.- Separación de usuarios	73
Figura 16.- Eliminar Usuario	73
Figura 17.- Modificar usuario	74
Figura 18.- Agregar usuario	75
Figura 19.- Administrar clientes	76
Figura 20.- Modificar cliente	76
Figura 21. -Agregar cliente	77
Figura 22.- Menú clientes	78
Figura 23.- Consulta de estatus del proyecto	78
Figura 24.- Cambiar contraseña	79

GLOSARIO DE TÉRMINOS

A	B
<p>Alfanumérico: Característica que indica un conjunto de caracteres que incluye letras, números y signos de puntuación</p> <p>Archivo: Conjunto de registros relacionados.</p> <p>Archivo de datos: son los que guardan el contenido.</p> <p>Atributo: Características de definen a una entidad.</p>	<p>Base de Datos: Conjunto de archivos o tablas o entidades o clases, sobre un mismo concepto y con una asociación lógica y física bien definida, también es llamada como Sistema de Bases de Datos.</p>
C	D
<p>Campo: Conjunto de caracteres(es el valor de un dato).</p> <p>Campo Alfanumérico: el que puede almacenar cualquier carácter(dígito, letra, símbolo especial).</p> <p>Campo Numérico: el que solo puede almacenar valores (dígitos).</p> <p>Carácter: Conjunto de bits (puede ser de 6, 8, 12 también se le conoce como byte).</p> <p>Código Fuente: Programa en su forma original, tal y como fue escrito por el programador, el código fuente no es ejecutable directamente por el computador, debe convertirse en lenguaje de maquina mediante compiladores, ensambladores o interpretes.</p>	<p>Dato: Es una unidad mínima de información, sin sentido en sí misma, pero que adquiere significado en conjunción con otras precedentes de la aplicación que las creó.</p> <p>Diagrama de flujo: Es la representación gráfica de una secuencia de instrucciones de un programa que ejecuta un computador para obtener un resultado determinado.</p> <p>Dominio: Es un rango de valores permitidos que existe para un atributo.</p>
E	F
<p>Entidad: Objeto que existe y puede distinguirse de otros objetos se identifica de los demás de forma única.</p> <p>Evento: Es la reacción que puede desencadenar un objeto, es decir la acción que genera</p>	<p>FTP (File Transfer Protocol): Protocolo de transferencia de archivos de una computadora a otra a través de una red (usado en Internet).</p>
H	I
<p>Hipervínculo: Conexión en distintos puntos de una página de Internet, que lleva a otro punto determinado del mismo sitio o de otro dentro de la red.</p> <p>HTML (HiperText Markup Language): Lenguaje de composición de páginas de hipertexto para la WWW.</p> <p>HTTP (HiperText Transfer Protocol). Protocolo de transferencia de hipertexto. Es el protocolo que permite navegar por la www.</p>	<p>Interfaz: Conexión entre dos componentes de hardware, entre dos aplicaciones o entre un usuario y una aplicación. También llamada por el término en inglés interface.</p> <p>Intranet: Denominación utilizada para referirse a la red interna de una empresa o institución también llamada</p>

K	L
<p>Key: Clave utilizada para acceder a datos protegidos por encriptación, también es conocida como llave</p>	<p>Lenguaje de programación: Conjunto de sentencias utilizadas para escribir secuencias de instrucciones para ser ejecutadas en una computadora.</p>
M	P
<p>Manejador de bases de datos: constituye la interfaz entre los datos de bajo nivel almacenados y las consultas que se hacen al sistema (Es una especie de controlador o cerebro de todo el sistema).</p> <p>Multiusuario: Permite que dos o más usuarios utilicen sus programas al mismo tiempo.</p>	<p>Password: Contraseña utilizada para ingresar en una red o en un sistema de manera segura. Conjunto de caracteres alfanuméricos requeridos para acceder a una determinada red, sistema, aplicación o recurso.</p> <p>Programa: Es una colección de instrucciones que indican a la computadora que debe hacer. Un programa se denomina software, por lo tanto, programa, software e instrucción son sinónimos.</p>
R	S
<p>Registro: Conjunto de campos relacionados.</p> <p>Relación: Es una asociación entre varias entidades.</p> <p>Render: unión de capas de cada una de las imágenes formando una sola.</p> <p>Rutina: Es el conjunto de instrucciones dentro del mismo programa, que se puede llamar a ejecución desde diferentes partes del mismo programa.</p>	<p>Servidor de Base de Datos: (Database Server) provee servicios de base de datos a otros programas u otras computadoras, como es definido por el modelo cliente-servidor. También puede hacer referencia a aquellas computadoras (servidores) dedicadas a ejecutar esos programas, prestando el servicio.</p> <p>Sistema: Conjunto de elementos interrelacionados que trabajan juntos para obtener un resultado deseado.</p>
T	U
<p>Tiempo Real: Responde a las entradas inmediatamente. Los sistemas operativos como DOS y UNIX, no funcionan en tiempo real</p> <p>Tupla o tabla: es una forma dentro de una base de datos, que nos sirve para almacenar y ordenar información, también es conocida como tabla.</p>	<p>Usuario: Cualquier individuo que interactúa con la computadora a nivel de aplicación. Los programadores, operadores y otro personal técnico no son considerados usuarios cuando trabajan con la computadora a nivel profesional.</p> <p>URL (Uniform Resource Locator): Localizador uniforme de recursos. Estándar que especifica un tipo de servicio en Internet, así como la localización exacta del archivo correspondiente. Dirección de un sitio web, universal y único a nivel mundial.</p>
V	W
<p>Variables: Es un espacio de memoria reservado para almacenar un valor que corresponde a un tipo de dato soportado por el lenguaje de programación. Una variable es representada y usada a través de una etiqueta (un nombre) que le asigna un programador o que ya viene predefinida.</p> <p>Vfx: efectos especiales que se aplica a imágenes.</p> <p>Video: Es una secuencia de imágenes.</p>	<p>World Wide Web (www): Sistema de organización de la información de Internet a través de enlaces hipertexto. En sentido estricto es el conjunto de servidores que emplean el protocolo HTTP.</p>

Antecedentes

Ollin Studio fundada en los 90's trabaja de la mano con industrias publicitarias y cinematográficas nacionales e internacionales, dedicada directa e indirectamente a la atención al cliente en servicios que van desde copiados de multiformatos, edición, postproducción de audio, video, efectos visuales y procesos de Intermedia Digital.

Actualmente en la Empresa Ollin Studio el proceso de administración de los archivos digitales, se hace manualmente, no se cuenta con ningún software que nos permita reducir tiempos y pérdidas económicas.

Ollin Studio es una empresa mexicana de servicios, más precisamente de efectos visuales y post-producción. Cuando se fundó la compañía, en la década del 90, ésta constituyó una novedad en México. Rápidamente creció y su facturación anual alcanzó niveles muy atractivos.

Parte de la historia de esta firma de animación para películas y comerciales arranca cuando Alejandro Diego decidió que vender software de animación a mediados de los años 90 no era suficiente para un joven ambicioso.

La historia se completa cuando Alejandro conoce a Carlos Iturriaga, un estudiante de preparatoria que creaba animaciones y deciden crear Ollin Studio, al darse cuenta que las casas productoras, sus futuros clientes, encargaban trabajos de animación a estudios en el extranjero. Al proyecto se le sumó Jorge Lizárraga.

Arrancó con un proyecto de animación por el que recibieron 15,000 pesos. Hoy, esta empresa factura 6 MDD y tiene 105 empleados.

A finales de 1997 Ollin empezó a ofrecer servicios de postproducción para retener a los clientes que contrataban proyectos de animación y efectos visuales. Actualmente localizados en Insurgentes Sur, en un edificio de cinco niveles, Ollin guarda la infraestructura para convertir la filmación de un comercial en un archivo digital.

Muestras de su trabajo las encontramos en el comercial de los ‘mariachis chinos’, donde un ejército de hombres de negro amenaza a los guardias de la Muralla China. Transmitido durante la reciente Copa Mundial de Fútbol, el televidente cree estar viendo a miles de mariachis chinos, pero en realidad se trata de una decena, a los que Ollin multiplicó usando un programa de software llamado Massive, vuelto famoso por la película El señor de los anillos.

El siguiente paso para Ollin es el cine. Ya participaron en los efectos especiales de Ladies’s Night, que Miravista –subsidiaria que Walt Disney Company lanzó en América Latina en sociedad con Telefónica– coprodujo con Argos Comunicaciones y Televisa Cine. Otro éxito en puerta es la película de terror Kilómetro 31.

“Ollin tiene la tecnología para producir cine”, menciona Roque Azcuaga, postproductor de Cinismo Films. Su inversión les permitirá atender proyectos de cine mexicano y extranjero, a precios 70% más bajos que los de un estudio en Estados Unidos. El objetivo de la empresa es que, el próximo año, la mitad de sus ingresos provengan de clientes extranjeros y triplicar sus ventas a 2011.

ORGANIGRAMA DE LA EMPRESA OLLIN STUDIO

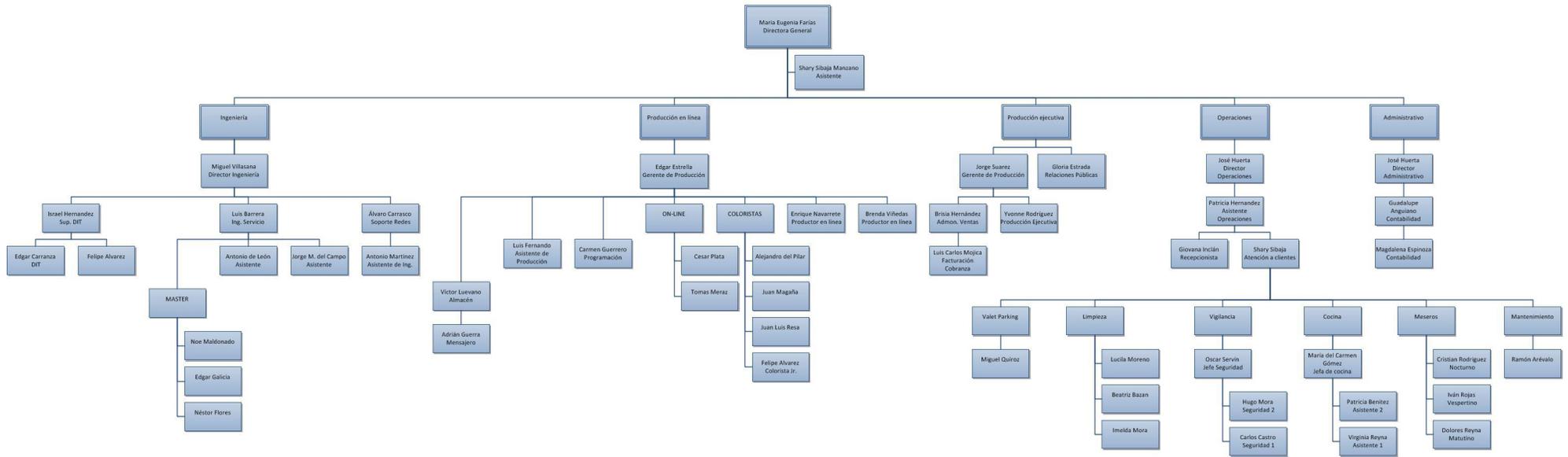


Figura 1.- Estructura Orgánica de la Empresa Ollin Studio

JUSTIFICACIÓN

Una administración eficaz, ahorra tiempo y dinero en cualquier empresa, esto se debe principalmente a la facilidad que se tiene en el manejo, control y acceso a los recursos.

Actualmente en la Empresa Ollin Studio la administración que se tiene de sus archivos digitales, no cuenta con ningún procedimiento que le permita reducir los tiempos en el manejo y control de los archivos que se generan a través de los diferentes procesos que se realizan en las estaciones de trabajo.

Con base en lo anterior, no se cuenta con una base sólida que permita un flujo de trabajo constante, los datos obtenidos no brindan información ni estadísticas oportunas sobre el aprovechamiento de los recursos humanos y técnicos que permitan tomar decisiones adecuadas para lograr una mayor eficiencia en cada uno de los procesos; no se informa el estado que guarda cada elemento del acervo digital, además de que los clientes no cuentan con un seguimiento veraz y oportuno del avance de sus proyectos. No se tiene un historial de los proyectos realizados que sirvan como referencia para su reutilización o desarrollo en proyectos futuros.

La mala administración que se tiene de los procesos ocasiona la pérdida o duplicación de la información que se genera durante la ejecución los procesos, esto trae como consecuencia la saturación y/o pérdida de información en los diferentes servidores, lo que se refleja en la cantidad de horas extras que se tienen que pagar a trabajadores por buscar o generar nuevamente esta información.

Por lo expuesto en lo anterior se deberá diseñar e implementar un sistema de información al que denominaremos “Sistema de Administración Múltiple” (SAM), que permita:

- ❖ Trabajar de forma automatizada,
- ❖ Reducir el tiempo de producción,
- ❖ Optimizar la administración del acervo digital,
- ❖ Controlar el almacenamiento de la información de acuerdo a la capacidad de los servidores.
- ❖ Identificar por medio del nombre el status del archivo, ubicación y detalles del mismo.
- ❖ Consultar los avances y/o status de los proyectos en línea y en tiempo real por parte de los clientes.

Para el desarrollo e implementación de este sistema requerimos:

- ❖ Sistema Operativo
- ❖ Manejador de base de datos
- ❖ Lenguaje de Programación
- ❖ Servidor de Páginas Web

Tomando en cuenta recursos con los que cuenta la empresa y cotizaciones realizadas a la misma, emplearemos Windows y Linux FedoraCore como sistemas operativos ya que será un sistema que puede ser montado en cualquier plataforma, MySQL para el manejador de datos, PHP como lenguaje de programación y APACHE como servidor de páginas web.

****Para mayor referencia acerca del software que se utilizó
ver ANEXO 1****

OBJETIVO GENERAL

Diseñar e implementar Sistema de Administración Múltiple (SAM), para mejorar la administración de acervo digital de la empresa Ollin Studio, con ello lograr una mayor productividad en la empresa. Mejorar estrategias y herramientas para organizar los recursos de medios digitales en su almacenamiento, seguridad, recuperación y reutilización. Permitiendo al usuario manejar toda la información necesaria de una manera automática y sencilla efectuando su trabajo de manera eficiente. Utilizando Windows y el arreglo de FedoraCore Linux como sistemas operativos, MySQL para el manejador de datos, PHP el lenguaje de programación, Web Apache para el servidor de páginas.

OBJETIVOS ESPECÍFICOS

Para el desarrollo del Sistema SAM se ha planteado lo siguiente:

- ❖ Analizar las necesidades de información por elementos sistemáticos, identificar el medio ambiente y recursos con los que se cuenta, identificar funciones que se realizan y procesos actuales que se llevan a cabo, definiendo alcances del sistema actual y futuro.
- ❖ Diseñar propuesta general de solución partiendo de la propuesta del análisis previamente revisada para generar actividades de diseño preliminar y diseño detallado y realizar un reporte del diseño.
- ❖ Construir y desarrollar programas y/o procesos para instalarse en un entorno que sea útil a la empresa, desarrollando procedimiento de usuario, pruebas y aceptación del sistema generando un reporte de la construcción
- ❖ Realizar pruebas de programas hasta su operación, integrando software y hardware, mediante las etapas de: preparación de la implementación, verificar condiciones para implementación y reporte de la implementación del sistema.

FASE I
MARCO TEÓRICO
Y
METODOLÓGICO

Capítulo I

MARCO TEÓRICO Y METODOLÓGICO

1.1 Ciclo de vida del desarrollo de sistemas

Ciclo de vida del desarrollo de un sistema (SDLC), es un enfoque por fases del análisis y diseño que sostienen que los sistemas son desarrollados de la mejor manera mediante el uso de un ciclo específico de actividades del análisis y del usuario.

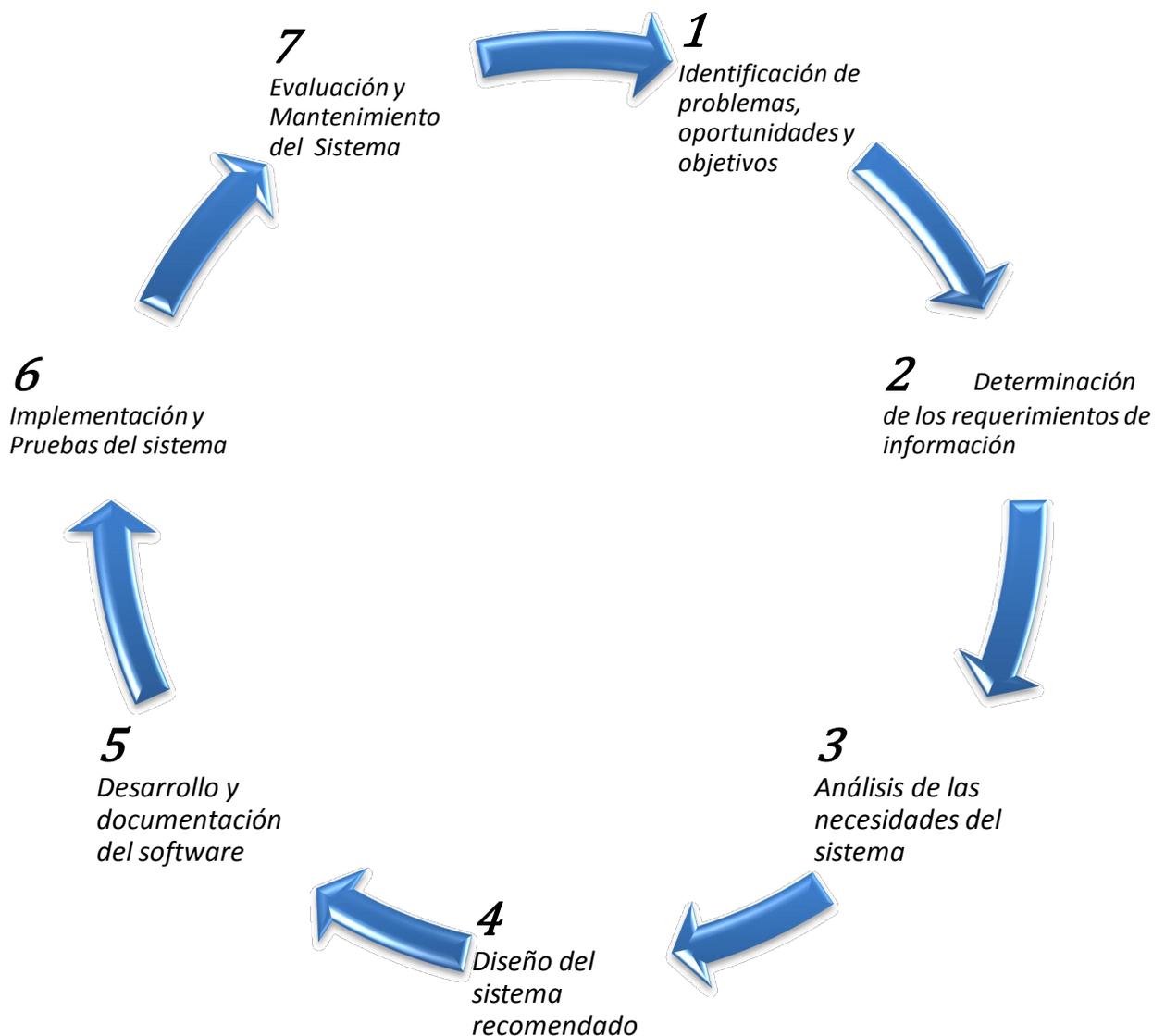


Figura 2.- Las siete fases del ciclo de vida para el desarrollo de sistemas.

Este ciclo se divide en 7 fases:

1.1.1 Identificación de problemas, oportunidades y objetivos

El análisis tiene que ver con la identificación de problemas y objetivos. Esta etapa es crítica para el éxito del resto del proyecto, las personas que participan en esta primera fase son usuarios, analistas y administradores de sistemas que coordinan el proyecto. Las actividades de esta fase consisten en entrevistas a los administradores de los usuarios, operarios, sumarización del conocimiento obtenido, estimación del alcance del proyecto y documentación de resultados.

1.1.2 Determinación de los requerimientos de información

En esta fase se hace el requerimiento de la información por medio de herramientas utilizadas como: muestreo e investigación de los datos relevantes, entrevistas, cuestionarios, el comportamiento de los tomadores de decisiones y su ambiente de oficina y elaboración de prototipos. Para determinar los requerimientos de información se involucra interacción directa con los usuarios.

Esta fase también nos sirve para formar la imagen que el analista tiene la organización y sus objetivos.

Las personas involucradas en esta fase son los analistas, usuarios, administradores de operaciones y trabajadores de operaciones.

1.1.3 Análisis de las necesidades del sistema

Involucra las necesidades del sistema con la ayuda de herramientas y técnicas especiales el analista hace las determinaciones de los

requerimientos. Se hace uso del diagrama de flujo de datos para diagramar la entrada proceso y salida de funciones en forma gráfica estructurada.

A partir de los diagramas de flujo de datos se desarrolla un diccionario de datos, que lista todos los conceptos de datos usados en el sistema, así como sus especificaciones, si son alfanuméricos y que tanto espacio ocupan cuando se imprimen.

Durante esta fase el analista de sistemas también analiza las decisiones estructuradas que se hacen. Las decisiones estructuradas son aquellas para las que pueden ser determinadas las condiciones como alternativas de condición, acciones y reglas de acción.

Hay tres métodos principales para el análisis de decisiones estructurales: lenguaje estructurado, tablas de decisión y arboles de decisión.

1.1.4 Diseño del sistema recomendado

En esta fase el ciclo de vida del desarrollo de sistemas, el análisis usa la información recolectada anteriormente para realizar el diseño lógico del sistema de información el analista diseña procedimientos precisos para la captura de datos, a fin de que los datos que van a entrar al sistema de información sean correctos. Además, el analista también proporciona entrada efectiva para el sistema de información mediante el uso de técnicas para el buen diseño de formas y pantallas.

La fase de diseño incluye el diseño de archivos o bases de datos que guardarán la mayor parte de los datos necesarios para los tomadores de decisiones de la organización. Una base de datos bien organizada es la base para todos los datos de la organización. Se trabaja con los usuarios para diseñar la salida (ya sea en pantalla o impresa) que satisfaga sus necesidades de información.

1.1.5 Desarrollo y documentación del software

El analista trabaja con los programadores para desarrollar cualquier software original que se necesite. Algunas de las técnicas estructuradas para el diseño y documentación de software incluyen diagramas estructurados. También se trabaja con los usuarios para desarrollar documentación efectiva para el software incluyendo manuales de procedimientos. La documentación dice al usuario la manera de usar el software y también que hacer si suceden problemas con el software.

1.1.6 *Implementación y Pruebas del sistema*

El sistema de información debe ser probado antes de ser usado, ya que es mucho menos costoso encontrar problemas antes que el sistema sea entregado a los usuarios. Algunas de las pruebas son realizadas por los programadores y otras por los analistas. Se le da mantenimiento rudimentario a lo largo de la vida del sistema de información, el mantenimiento debe ser mínimo.

1.1.7 Evaluación y mantenimiento del sistema

En esta fase el desarrollo del sistema el analista a implementar el sistema de información. Esto incluye el entrenamiento de los usuarios para que manejen el sistema. Algún entrenamiento es hecho por los proveedores pero la supervisión del entrenamiento es responsabilidad del analista de sistemas.

1.2 Modelo Conceptual

Arquitectura de una estructura organizada como base de datos, esta estructura se divide en tres niveles generales: interno, conceptual y externo.

❖ **Nivel Interno o Físico**

Es el más cercano al almacenamiento físico es decir la manera como los datos se almacenan en realidad. En este nivel se describen en detalle las estructuras de datos complejan el nivel más bajo.

❖ **Nivel conceptual**

En este nivel se describen cuales son los datos reales que están almacenados en la base de datos y que relaciones existen entre los datos.

❖ **Nivel externo o de visión**

En este nivel más cercano a los usuarios, describe la forma en que cada usuario va la parte de la base de datos que le interesa. El sistema puede proporcionar muchas vistas diferentes de la misma base de datos.

1.2.1 Modelo cliente-servidor

La aplicación está basada en el modelo Cliente - Servidor. La tecnología Cliente-Servidor es utilizada por todas las tecnologías de Internet/Intrared: Un servidor es un ordenador remoto (en algún lugar de la red) con dirección conocida que proporciona información según la petición del cliente.

Un cliente funciona en su ordenador local, se comunica con el servidor remoto, y pide a este la información, el servidor por su parte envía de regreso la información solicitada.

En los capítulos subsecuentes vamos a tratar la instalación y configuración de un servidor web, utilizando Apache como servidor de páginas web, PHP como lenguaje interpretado de alto nivel y MYSQL como base de datos.

Con esta combinación podremos crear páginas dinámicas y obtener información de nuestra base de datos para presentarla vía web.

El proceso para presentar las páginas con código PHP es la siguiente:

- 1.- El navegador envía una petición de página al servidor.
- 2.- El servidor responde con una página PHP.
- 3.- El servidor procesa el código PHP.
- 4.- El servidor envía el resultado en una página HTML.



Figura 3.- Presentación de Páginas PHP

FASE II
ANÁLISIS
Y
DIAGNÓSTICO

Capítulo II

ANÁLISIS Y DIAGNÓSTICO

2.1 Análisis y diagnóstico del sistema actual

Determinar lo que el cliente desea que el sistema realice. El Modelo Generado por el Análisis de Requerimientos puede funcionar como un contrato entre el desarrollador y el contratante describiendo la visión del desarrollador de lo que el cliente desea. Para conocer mejor los procesos, la forma de trabajo y necesidades de usuarios de la empresa Ollin Studio se realizó una entrevista a empleados de la misma, con el fin de recopilar información y obtener todo tipo de documentación para el desarrollo de un sistema que cumpla y cubra todas las necesidades del usuario. Se realizó una entrevista a empleados de la empresa Ollin Studio y el resultado fue el siguiente:

Se requiere de un sistema que facilite:

- ❖ La administración de acervo digital, que brinde una solución integral para el manejo del acervo digital que garantice su integridad y una rápida disponibilidad en todo momento.
- ❖ Que sea desarrollado en una base sólida que permita un flujo de trabajo constante, que brinde información y estadísticas oportunas sobre el aprovechamiento de los recursos tanto humanos como técnicos que permitan la toma de decisiones adecuadas para lograr una mayor eficiencia en cada uno de los procesos.
- ❖ Que sea un medio de comunicación entre las distintas etapas del proceso de producción con el fin de informar el estado que guarda cada elemento del acervo digital, que forme un vínculo entre Ollin Studio y los clientes de forma que ellos puedan estar enterados del avance de sus proyectos.
- ❖ Que guarde un historial de los proyectos realizados en el pasado con la finalidad de que puedan ser reutilizados en proyectos futuros.

2.1.1 Técnicas para la obtención de información

1. Actividades realizadas por el área y el sistema actual
2. Relación existente entre las diversas operaciones o flujo de procesos
3. Procedimientos de operación
4. Limitaciones de tiempo y programas de trabajo y/o actividades
5. Controles requeridos
6. Reportes o salidas requeridas y entradas necesarias.
7. Tipos, uso y retención de datos.
8. Costos y beneficios posibles.

Para mayor referencia acerca del formato que se aplicó en la entrevista ver ANEXO 2

2.1.2 Identificación de las necesidades de información

La identificación de necesidades (requerimientos), es el estudio actual de la empresa o área a fin de encontrar como opera y donde puede o debe mejorarse.

Una necesidad o requerimiento, es una característica que debe incluirse en un nuevo sistema y puede consistir en una forma de captar y procesar datos, producir información, controlar una actividad de la empresa, o apoyar La toma de decisiones de la gerencia.

Por lo tanto la identificación de las necesidades implica estudiar el sistema existente ya sea como un enfoque de operación: manual o computarizado). Con este fin, se debe analizar la información con relación al mismo para así encontrar cuales son estas necesidades.



En primera instancia, es necesario ubicar si la identificación de necesidades de información y orientar a sistemas de toma de decisiones, o a sistemas de apoyo a la operación (que posteriormente, pueden servir para generar sistemas para la toma de decisiones).

Con base a lo descrito anteriormente se puede empezar a desarrollar un perfil del futuro sistema, en base a la información recabada. Ese perfil indicará los posibles recursos requeridos: como los son humanos, hardware, software, etc. Para ello se deben emplear los resultados obtenidos con anterioridad.

2.1.3 Tabla Sistemática

Subsistemas	Procesos	Entradas	Salidas	Tiempos	Volúmenes	Datos	Controles	Distribución
Administración de archivos digitales	Cotización del Proyecto	Elaborar de una cotización tentativa, la cual deberá ser aprobada por el cliente.	Cotización aprobada.	1 día	1/día	Precio del proyecto	Aprobación del cliente en la cotización	ServidorSAM1
	Preproducción	Recepción de archivos EDL's y referencias de todo el material filmado y adquirido previo a la cotización.	Material aprobado empieza producción.	4 días	1/día	Archivos EDL's y material adquirido	Revisión y corrección de material.	ServidorSAM2
	Producción	Planeación de los tiempos y forma en cómo se llevará a cabo la filmación	Finalización del film del proyecto.	1 semana	1/día	Film en proceso	Revisión y corrección del material.	ServidorSAM2
	Intermedia Digital	Digitalización del film por el Northlight tomando como base los EDL's y las prioridades indicadas por parte de la producción.	Se complementa film dándole tratamiento.	2 a 3 días	1/día	Tratamiento de film	Revisión y corrección de film	ServidorSAM3
	Film Out y Resultados	El film lleva un proceso de impresión mediante la film-recorder para su revelado, además pasan al editor de HD para llevar a cabo su traspaso a video.	El material es entregado al cliente, no sin antes haber realizado una copia de seguridad.	1 día	1/día	VoBo del cliente.	Firma de conformidad del cliente	ServidorSAM3

*** **NOTA:** Los tiempos y volúmenes va en relación a la duración del film, ya que puede ser de comercial de tv o de película. En la segunda mencionada los tiempos cambian incrementando el tiempo en cada proceso.

2.2 Conocimiento de Procesos en Ollin Studio

Es de gran importancia tener un conocimiento detallado de los procesos por los que pasan los proyectos de cine en Ollin Studio. Esto nos permite el desarrollo de estrategias y soluciones adecuadas para un eficiente flujo de datos. Después de algunos meses dedicados a la investigación en esta área, los procesos se pueden separar en cinco grandes grupos.

❖ Cotización del Proyecto

El principal objetivo de este grupo de procesos es llegar a la elaboración de una cotización tentativa, la cual deberá ser aprobada por el cliente. La cotización tentativa será elaborada tomando como base la estimación de uso de equipo y los tiempos necesarios para su realización. **Cabe destacar que si la cotización es rechazada el proyecto se dará por finalizado y por obvias razones ya no tendría continuidad.**

Para la elaboración de una cotización se debe realizar la lectura del guion e identificar las partes del film que sean candidatas para la incorporación de algún Vfx. Estas candidatas se enriquecen con referencias y ejemplos para que sean aprobadas por el cliente. Con las partes aprobadas y el visto bueno del cliente se realiza la cotización y se pasa a la siguiente etapa.

❖ Preproducción

Al inicio del proceso de preproducción se recibirán los archivos que fueron creados en el proceso de cotización como los EDL's², EDL's y referencias de todo el material filmado y adquirido previo a la cotización.

¹Llamamos referencias a todo tipo de acervo digital que sirva para dar una idea más cercana de lo que se quiere lograr a través de algún proceso de Vfx.

²Edit Decision List, es una forma de representar la edición del film tomando como base el nombre de los rollos y el timecode de cada evento.

Es importante mencionar que los EDL's deben ser verificados ya que una vez aceptados por Ollin Studio **NO** podrán ser modificados, esto se debe principalmente a que en ellos están basados el resto de los procesos de producción y un cambio representa una grave alteración en el flujo de trabajo y en la estructura de los datos resultando en una pérdida de la productividad y eficiencia de los procesos.

❖ Producción

El objetivo de este conjunto de procesos es planear los tiempos y forma en cómo se llevará a cabo la filmación, principalmente en aquellas partes del film que necesiten un tratamiento especial debido a la incorporación de algún Vfx. Aquí también se realizará la supervisión de la filmación, pruebas de apariencia y pre-compuestos además de la elaboración de los Rushes³-Dailies-necesarios para la creación de la Copia de Trabajo. Esta etapa finaliza junto con la filmación de la película.

❖ Intermedia Digital

Sin duda es el conjunto de procesos que constituyen el alma de los proyectos. Su desarrollo consta tanto de procesos creativos como administrativos y es en esta etapa donde los principales recursos de hardware son utilizados. Esta etapa comienza con recepción de los EDL's, referencias y copia de trabajo por parte del Data Wrangler⁴ que se encargará de colocarlas en cada uno de los servidores correspondientes para que puedan ser consultadas por los departamentos adecuados.

³ Se conoce como Rushes a la captura del material sin ningún tipo de tratamiento, usualmente esto se hace día con día durante el proceso de filmación a través de un telecine.

⁴ El Data Wrangler es parte fundamental en la administración del acervo digital ya que es el encargado de dosificar el espacio y el tráfico entre los servidores, para ello debe conocer ampliamente cuales son los recursos con los que se cuentan y los diferentes protocolos de comunicación entre ellos. Además analizar las prioridades entre los diferentes acervos para que el flujo de trabajo sea continuo.

Posteriormente el film será digitalizado por el Northlight⁵ tomando como base los EDL's y las prioridades indicadas por parte de la producción, los archivos digitales serán almacenados en la jerarquía de carpetas previamente elaborada por el Data Wrangler.

Una vez escaneadas las secuencias el Data Wrangler creará los EfectosComplementos⁶ de media y un cuarto de resolución, además también deberá de convertir la copia de trabajo a cuadros, para posteriormente colocar todo en los servidores correspondientes. En este punto el flujo de trabajo recibe una bifurcación, por una parte tenemos aquellas secuencias que no llevarán ningún tipo de compuestos ni animación.

Estas secuencias pasarán directamente al Baselight⁷ para su corrección de color. Por otra parte tenemos las secuencias que involucran algún tipo de trabajo en compuestos y animación. Si la secuencia necesita algún efecto de animación el proceso comenzará al recibir los EfectosComplementos, las referencias y demás datos necesarios según el tipo de trabajo requerido, estos datos pueden ser:

Medidas de los objetos, información de cámara, esferas HDRI, etc. Al terminar la animación, previa aprobación por el supervisor de efectos y el cliente, el render será copiado por el Data Wrangler al área de compuestos para su composición.

Si la secuencia necesita algún tipo de compuesto los EfectosComplementos serán copiados al servidor de la sala de compuestos junto con las referencias e información necesaria para su realización. Al terminar la composición el Data Wrangler copiará el render al Baselight para su

⁵ Northlight es el sistema para realizar el film escáner en alta resolución.

⁶ Los EfectosComplementos son conversiones de menor calidad de las digitalizaciones realizadas en el Northlight que pueden ser usados en las estaciones de trabajo de compuestos y animación.

⁷ Baselight es el sistema para realizar la corrección de color, el armado del film y permite su reproducción en tiempo real a una resolución hasta de 4K.

aprobación en sala por parte del cliente. Si el compuesto es rechazado deberá ser modificado hasta lograr su aprobación, una vez aprobado pasará a la corrección de color.

Al terminar la corrección de color de todo el film se lleva a cabo el render final, este render será copiado a discos duros para realizar la limpieza del film y verificar que se encuentren la totalidad de los cuadros. Con este proceso se da por concluido el conjunto de Intermedia Digital.

❖ **Film Out y Resultados**

En esta etapa el film, hasta ahora una secuencia de imágenes almacenadas en discos duros-pasa al proceso de impresión en donde cada uno de los cuadros serán registrados en la película mediante la film-recorder para su posterior revelado, además las secuencias de cuadros pasan al editor de HD para llevar a cabo su traspaso a video.

Al final de este proceso todo el material es entregado al cliente, no sin antes haber realizado una copia de seguridad. Con esto los proyectos se dan por finalizados. El flujo de trabajo completo se muestra en la Figura 4.

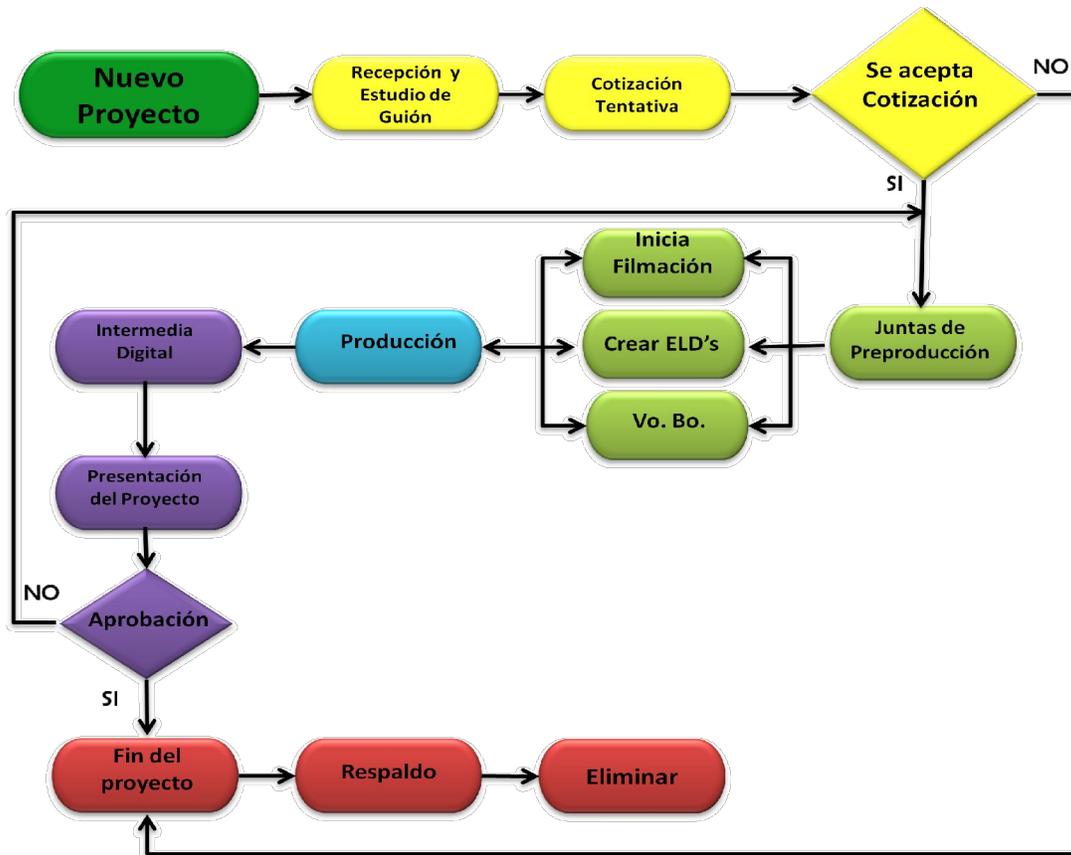


Figura 4.- Flujo de trabajo en Ollin Studio

2.3 Forma de trabajo actual en Ollin Studio

Se crea una carpeta con el nombre del proyecto, después se crean subcarpetas en donde se almacenan los diferentes materiales que llegan de ese proyecto. El nombre de la subcarpeta se asigna de acuerdo a procedencia y uso.

El trabajo se hace por terminal segura del servidor, con diferentes comandos. Cada servidor posee una versión diferente de Linux, ya que cada servidor es de distinto fabricante. Al implementar el sistema SAM la empresa podrá trabajar de forma automatizada, el tiempo invertido en producción se reduciría.

También se logrará que los clientes puedan consultar avances de sus proyectos en línea y en tiempo real, actualmente no se cuenta con ese servicio.

El proceso descrito anteriormente está perjudicando a la empresa en pérdida de tiempo e ingresos, se invierte mucho tiempo a un proyecto por el proceso que actualmente se maneja, se ve reflejado en horas extra que hay que pagar a los trabajadores, además hay duplicidad de información y eso nos crea pérdidas de espacio en capacidad en servidores de almacenamiento. También se elimina información por no tener un buen control sobre la misma.

2.4 Modelo entidad-relación

Es la percepción de un mundo real, consiste en un conjunto de objetos básicos llamados entidades y que relaciones entre estos objetos, facilita el diseño de las bases de datos y permite especificar de una forma clara y gráfica en esquema de información, representa una estructura lógica general de la base de datos, puede ser un esquema lógico o conceptual.

Una entidad puede ser considerada como un objeto que existe y puede distinguirse de otros objetos, es decir, se identifica de los demás en forma única. Una entidad puede ser concreta (una persona. Un libro) o bien puede ser abstracta (un día festivo o un concepto).

Conjunto de entidades (CLASE)

Conjunto de entidades del mismo tipo, algo importante a notar es que no es necesario que las entidades pertenezca exclusivamente (en un momento dado) a un solo conjunto de entidades. Por eso definimos el conjunto de entidades de todos los archivos que se manejan en Ollin Studio (Archivos), así como proyectos, usuarios, perfil_maestro, perfil_detalle y programa.

Atributo: podemos considerarlo como todas las características que definen a una entidad.

De acuerdo a lo anterior cada entidad se describe por medio de un conjunto de parejas (Atributo, valor atributo), un atributo para cada pareja de la entidad.

- ✓ El concepto de conjunto de entidades (Clase o archivo) corresponde a la idea de definición de tipo en un lenguaje de programación-
- ✓ El concepto de entidad corresponde a la idea de variable, en un lenguaje de programación
- ✓ Así una base de datos descrita bajo este modelo de datos, incluye una agrupación de conjuntos de entidades, cada uno de los cuales contiene cualesquier número de entidades del mismo tipo.

2.4.1 Limitantes de mapeo

Es un esquema E-R que representa a un concepto, se puede definir ciertas limitantes con las que deberán cumplir los datos contenidos en la base de datos.

2.4.2 Cardinalidades de Mapeo

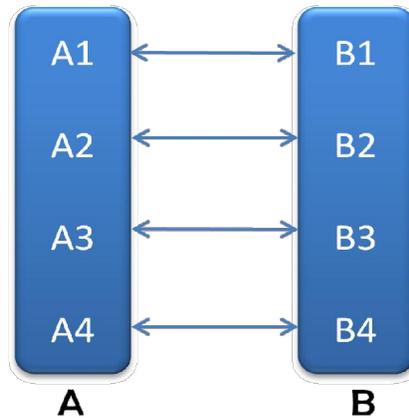
Es una limitante importante que expresará el número de entidades con las que pueden asociarse otra entidad, mediante una relación.

2.4.3 Cardinalidades de mapeo para conjuntos binarios de relaciones

Para un conjunto binario de relaciones entre los conjuntos de entidades A y B, la cardinalidad de mapeo debe ser una de las siguientes:

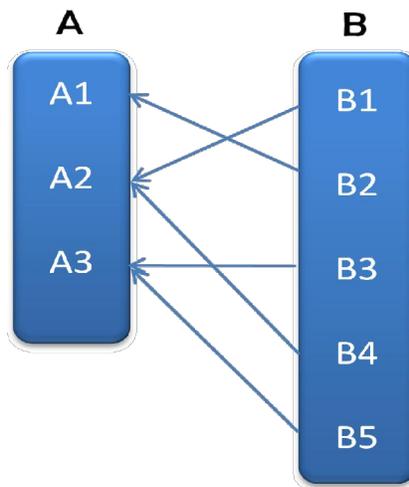
❖ Una A Una

Una entidad A que está asociada únicamente con la entidad en B y una entidad en B asociada solo con una en A.



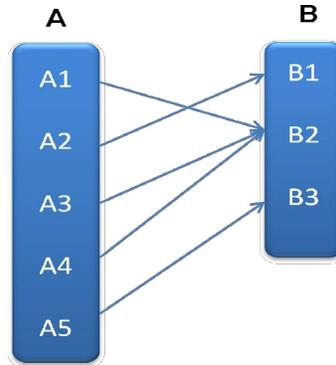
❖ Una A Muchas

Una entidad en A esta relacionada con cualquier número de entidades en B, pero una entidad en B puede asociarse únicamente con una entidad en A.



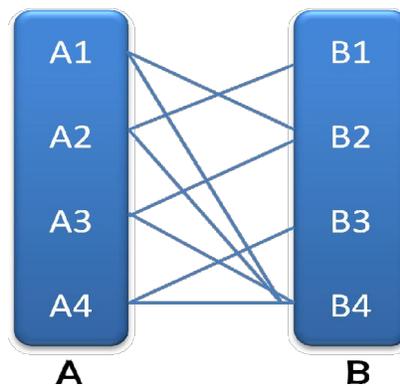
❖ **Muchas A Una**

Una entidad en A esta vinculada únicamente con una entidad en B, pero una entidad en B está relacionada con cualquier número de entidades en A.



❖ **Muchas A Muchas**

Una entidad en A esta asociada con cualquier número de entidades en B, y una entidad B está vinculada con cualquier número de entidades en A.



La cardinalidad de mapeo apropiada para un conjunto de relaciones determinado, dependerá del mundo real que el conjunto esta modelado.

2.5 Diagrama Entidad-Relación SAM

Es una representación gráfica de la estructura lógica general de una base de datos y se representa mediante los siguientes símbolos:

- ❖ **RECTÁNGULOS**
Representan conjuntos de entidades o clase

- ❖ **ELIPSES**
Representan atributos

- ❖ **ROMBOS**
Representan conjuntos de relaciones o asociaciones

- ❖ **LÍNEAS**
Se utilizan para la conexión de los atributos s los conjuntos de entidades y los conjuntos de entidades a los conjuntos de relaciones.

A cada componente o símbolo se le asocia su nombre a los conjuntos de entidades y los conjuntos de entidades a los conjuntos de relaciones.

Para el sistema SAM el modelo entidad-relación es el siguiente:

proyecto	idproyecto
	tipo_de_proyecto
	nombre_del_proyecto
	fecha_de_creacion
	fecha_ult_mod
	status
	ubicacion
	tamaño

usuarios	usr_usuario departamento perfm_cve usr_pass usr_email
archivos	Idproyecto nombre_archivo ubicacion tamaño
perfil_maestro	perfm_cve perfd_cve perfd_orden prog_cve

SAM cuenta con las entidades de: proyectos, archivos, usuarios y perfil_maestro

La relación entre cada una de las entidades es la siguiente:

- * La entidad **archivos** tiene relación de muchas a una con la entidad proyectos por medio de idproyecto de tipo entero.
- * La entidad **proyectos** tiene relación de muchas a una con Usuarios por medio de perfm_cve tipo entero.
- * La entidad **usuarios** tiene relación de uno a uno con perfil_maestro por medio de perfm_cve de tipo entero.
- * La entidad **perfil_maestro** tiene una relación de uno a uno con perfil_detalle por medio de prog_cve de tipo entero.

DIAGRAMA ENTIDAD-RELACIÓN

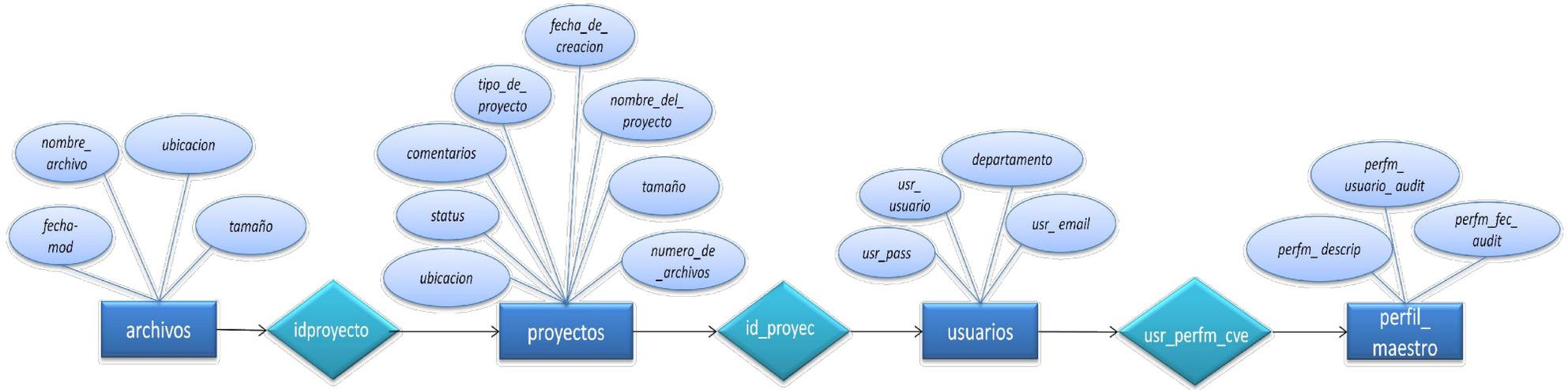


Figura 5.- Diagrama Entidad-Relación

2.5.1 Descripción de Relaciones

La relación entre la entidad usuarios y perfil_maestros es de uno a uno, esto es porque un usuario solo puede tener un perfil. Al mismo tiempo se relaciona con la entidad proyecto, en la que la relación es uno a muchos, porque, un usuario puede tener uno o más proyectos, a la vez la entidad proyecto se relaciona con la entidad archivo, en donde la relación es uno a muchos, ya que un proyecto contiene muchos archivos. Regresando a la entidad perfil_maestro está también se relaciona con la entidad perfil_detalle, en donde la relación es uno a uno, esto porque en la entidad perfil_detalle se describe al perfil_maestro, y solo puede corresponder un perfil_maestro a un perfil_detalle. La entidad perfil_detalle se relaciona con la entidad programa, esta relación es de uno a muchos porque un perfil puede contener varios programas, que un usuario con ese perfil puede ver y tiene permisos.

2.6 Entidades SAM

Para el sistema SAM todas las entidades y atributos que conforman el diagrama de entidades/asociaciones diseñado para el sistema.

❖ Entidad Proyectos

ATRIBUTO	DESCRIPCIÓN
idproyecto	Contiene un número consecutivo que asigna el programa por default de todo proyecto que ingresa.
tipo_de_proyecto	Nos indica el tipo de proyecto generado ya sea Comercial de televisión o película.
nombre_del_proyecto	Contiene el nombre del proyecto, tal como se encuentra almacenado físicamente.

fecha_de_creacion	Contiene la fecha en que fue creado el proyecto.
status	Contiene el estado en el que se encuentra el proyecto, puede ser en inicio, intermedio o finalizado.
ubicación	Contiene la ubicación exacta del lugar donde se encuentre físicamente.
tamaño	Contiene el tamaño exacto del Proyecto físicamente.
comentarios	Contiene información descriptiva acerca del proyecto.
numero_de_archivos	Contiene el número de archivos que contiene el proyecto.
id_proyec_usr	Contiene un identificador para asociarlo con un usuario.

Tabla 2. -Descripción de Entidad Proyectos

❖ Entidad Usuarios

ATRIBUTO	DESCRIPCIÓN
usr_usuario	Contiene el nombre de los diferentes usuarios del sistema SAM.
departamento	Contiene la fecha en que fue creado el proyecto.
perfm_cve	Tipo de permisos y/o privilegios a los que va a tener acceso el usuario dentro del sistema SAM.
usr_pass	Clave de acceso al sistema SAM.
usr_email	Contiene el correo electrónico del usuario SAM.
id_proyecto	Contiene un identificador para los proyectos.

Tabla 3. -Descripción de Entidad Usuarios

❖ **Entidad Perfil_maestro**

ATRIBUTO	DESCRIPCIÓN
perfm_cve	Tipo de permisos y/o privilegios a los que va a tener acceso el usuario dentro del sistema SAM.
perfm_descrip	Contiene una descripción del perfil

Tabla 4. -Descripción de Entidad Perfil_maestro

❖ **Entidad Archivos**

ATRIBUTO	DESCRIPCIÓN
idproyecto	Contiene un número del proyecto al que pertenece.
nombre_archivo	Contiene un el nombre del archivo, tal como se encuentra almacenado físicamente.
ubicacion	Contiene la ubicación física del archivo.
tamaño	Contiene el tamaño del archivo.
fecha_mod	Contiene la fecha de la última modificación

Tabla 5. -Descripción de Entidad Archivos

2.7 Llaves

Dentro del modelo de base de datos, existe una tarea muy importante que consiste en especificar cómo se van a distinguir las entidades y las relaciones. Para realizar esta distinción, se asigna una superllave a cada conjunto de entidades.

❖ **Superllave**

La superllave es un conjunto de uno o más atributos que juntos, permiten identificar en forma única a una entidad dentro del conjunto de entidades. Si K es una superllave, entonces también lo será cualquier superconjunto de K.

❖ **Llave candidato**

Son superllaves para las cuales ningún subconjunto propio es una superllave

❖ **Llave primaria**

Este término se utiliza para referirse a la llave candidato que elija el diseñador de la base de datos, como la forma principal de identificar a las entidades dentro de un conjunto de estas. Para SAM en la relación está determinada por llaves primarias.

2.8 Recursos de Almacenamiento para la Administración de Datos

Al día de hoy la organización de los principales recursos de almacenamiento se muestra en la **Tabla 6**.

Servidor	Tamaño	Recursos Compartidos	Protocolos
ServidorSAM1	700 GB, 1 TB,	/brooklin, /filmale, /filmale2	ftp, samba, nfs,
ServidorSAM2	Variable 24 TB, 1.4	/images1, /pfs1 /illusion	scpnfs, samba,
ServidorSAM3	TB 3.4 TB 2.4 TB 1 TB 3.7 TB 300 GB	/home/project /tránsito /S /ftpla	scpnfs, scp, samba nfs, scp, samba nfs, scp samba samba ftp

Tabla 6.- Principales recursos de almacenamiento

El Data Wrangler debe considerar la forma de tránsito más eficiente, es decir, aquella que consume la menor cantidad de recursos y de tiempo, además tomar en cuenta el no afectar el trabajo de otros departamentos.

Es de vital importancia que después de cada proceso de copiado y traspaso se verifique la existencia y consistencia de los datos para evitar así la pérdida de información, sin olvidar que para realizar cualquier operación cuyo destino sea el recurso “/pfs1” ésta deberá realizarse con los comandos provistos por Filmlight⁸, es decir, fl-cp, fl-mv, fl-ls, fl-rm y fl-code⁹.

⁸ Filmlight es el fabricante de los recursos Baselight, Northlight y Truelight.

⁹ Para una mayor referencia consultar el documento *Datawrangling for dummies*, Miguel Villasana 2006.

Los departamentos encargados de la administración del espacio en los recursos compartidos se muestra en la **Tabla 7**, de esta forma es responsabilidad de cada departamento llevar a cabo una depuración constante del espacio para evitar la saturación de los servidores, en consecuencia cualquier información contenida en los servidores y no reportada a los departamentos encargados de su administración podrá ser borrada sin previo aviso.

Departamento Responsable	Recursos Compartidos
Data Wrangler	/filmale, /filmale2, /images1, /pfs1,/tránsito, /S
Illusion	/illusion
Animación	/home/project
Ingeniería	/ftpla, /brooklin

Tabla 7.- Departamentos responsables de la administración del espacio en los servidores.

2.8.1 Herramientas para la Administración del Acervo Digital

La productividad y el desarrollo en cada etapa del proceso de producción dependen en su mayoría de la adecuada administración de los recursos humanos. Son los recursos humanos el principal motor para el crecimiento de Ollin Studio, por ello es necesaria una adecuada selección de los mismos, pero sobretodo es importante dotarlos de una capacitación continua, de la información necesaria para un mejor desarrollo de sus habilidades y de las herramientas necesarias para facilitar su trabajo.

En el caso concreto de la administración de acervo digital una herramienta que sería de gran utilidad es el desarrollo del software que facilite y haga transparente a las personas el manejo del acervo digital.

Para esto debe tomarse en cuenta las necesidades reales de cada persona involucrada en el flujo de trabajo, además la cantidad de recursos técnicos con los que se cuenta.

Entre las principales características que debe cumplir cualquier software dedicado a la administración de acervo digital se encuentran:

- (1) Desarrollo Multiplataforma.
- (2) Facilidad de Uso y Aprendizaje.
- (3) Altamente Configurable.
- (4) Fácilmente Escalable.
- (5) Ampliamente Documentado.

2.9 Propuesta General de Solución

El sistema será un software diseñado para facilitar la administración de acervo digital. Para su desarrollo se tomó en cuenta el flujo de trabajo como los recursos de almacenamiento disponibles.

Su objetivo general es brindar una solución integral para el manejo del acervo digital que garantice su integridad y una rápida disponibilidad en todo momento.

Como objetivos particulares podemos destacar:

- (1) Formar una base sólida que permita un flujo de trabajo constante.
- (2) Brindar información y estadísticas oportunas sobre el aprovechamiento de los recursos tanto humanos como técnicos que permitan la toma de decisiones adecuadas para lograr una mayor eficiencia en cada uno de los procesos.



(3) Ser un medio de comunicación entre las distintas etapas del proceso de producción con el fin de informar el estado que guarda cada elemento del acervo digital.

(4) Formar un vínculo entre Ollin Studio y los clientes de forma que ellos puedan estar enterados del avance de sus proyectos.

(5) Guardar un historial de los proyectos realizados en el pasado con la finalidad de que puedan ser reutilizados en proyectos futuros.

FASE III
DISEÑO
Y
CONSTRUCCIÓN

Capítulo III

DISEÑO Y CONSTRUCCIÓN

El diseño es la solución, es decir, una traducción de los requerimientos en formas que los satisfagan. Tomando en cuenta especificaciones detalladas del sistema, describiendo sus características como las salidas, entradas, procesos, archivos y/o bases de datos, controles, todos de manera que se cubran los requerimientos del proyecto en el análisis.

Plan para delinear la forma y el detalle de una solución. Es el proceso que determina las mejores características del sistema final, establece las fronteras superiores e inferiores en el comportamiento ó en la calidad de la implantación debe tener, así mismo determina el costo final que tendrá y su duración en construcción.

3.1 Estructura de datos

Todos los datos de un sistema consisten de elementos-datos. Los elementos-datos, se agrupan para conformar una estructura de datos.

El nivel de datos esencial es el elemento-dato (o simplemente, llamado o conocido por: campo, dato, parte elemental, atributo). No hay unidad más pequeña que tenga significado para el usuario del sistema, por ejemplo: `usr_usuario`, `departamento`, `perfm_cve`, `usr_pass`, `usr_email`, son elementos-dato en el flujo de datos (en la parte de almacenamiento de la información), de un sistema.

Entonces se puede decir, que una estructura de datos en un conjunto de datos que están relacionados entre sí y que describen en forma genérica las características estructurales de información de un componente del sistema.

Lo anterior se esquematiza de la siguiente forma:

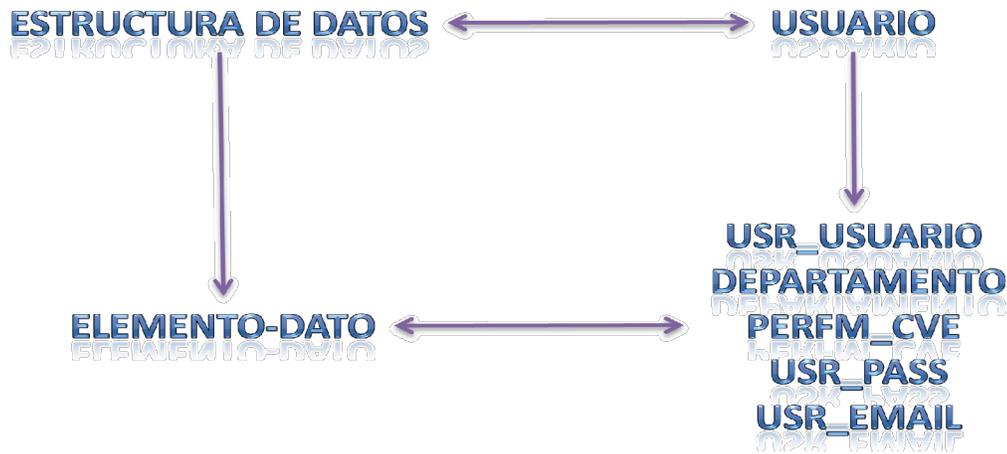


Figura 6.- Esquema de la relación entre un elemento-dato y una estructura de datos.

Se consideraron los siguientes objetivos:

1. Que sea más eficiente. Es decir un mejor uso de los recursos de Hardware, personal, económicos, etc.
2. Sea más fácil de mantener
3. Sea modificable. Es decir, la facilidad para adaptar al sistema a un cambio en el medio ambiente.
4. Sea flexible. Es decir, la facilidad para adaptar el sistema a un cambio en el medio ambiente.
5. Sea general. Quiere decir, la capacidad del sistema para el manejo u operación de un tema o concepto en una forma amplia
6. Sea útil. De aplicación inmediata y que apoye alguna o algunas funciones o procesos importante o necesarios de la empresa o áreas involucradas, para que así tengan un mejor desempeño.

3.2 Diseño Preliminar

En este apartado vamos a delinear y detallar la solución y el proceso que determina las mejores características del sistema final, establece las fronteras superiores e inferiores en el comportamiento y en la calidad que la implantación debe tener, así mismo determina el costo final que tendrá y su duración en construcción.

Los objetivos que contemplamos son:

- ❖ Eficiencia, donde encontramos un mejor uso de los recursos de Hardware, personal, económicos, etc.
- ❖ Fácil de mantener.
- ❖ Modificable, facilidad para adaptar al sistema a un cambio en el medio ambiente.
- ❖ Flexible, facilidad que deba tener el sistema para ejecutar variaciones en su proceso con poca o ninguna modificación.
- ❖ General, La capacidad del sistema para el manejo u operación de un tema o concepto de una forma amplia.
- ❖ Útil, de aplicación inmediata y que apoye alguna o varias funciones o procesos importantes o necesarios de la empresa o área(s) involucradas para que tengan un mejor desempeño.

Se tomó a consideración lo siguiente:

- ❖ **Asignación de nombre de los archivos**

Para distinguir un archivo de otro, se asignan nombre significativos que se utilizan para tener una referencia de cada elemento a través de proceso total de desarrollo de sistemas. es posible que se les asignen: las iniciales de la frase que los identifica o sus acrónimos o una etiqueta clasificatoria.

❖ **Descripción de los archivos**

La descripción de los archivos establece brevemente lo que representa el archivo en el sistema. La nomenclatura para el nombre de los archivos debe estar estructurada de forma que la persona que lo lea que no conoce nada o conoce muy poco, pueda identificar con facilidad en contenido y status del archivo.

❖ **Requerimientos de personal**

Para calcular los requerimientos de personal se requiere evaluar el tiempo de programación de software la estimación depende de tres elementos principales:

1. Nivel de experiencia del programador.
2. Nivel de complejidad del programa.
3. Nivel de comprensión del programador respecto al programa.

❖ **Análisis económico (Costos y beneficios del sistema)**

El costo es uno de los aspectos que determina si un sistema es aceptado o no, para el estudio económico de este sistema se identificó y estimó apropiadamente los costos.

Los costos varían según el tipo, el beneficio también varía de acuerdo con su tipo y ambos se clasifican según las ventajas que proporcionen al área.

El beneficio costo se genera en cuanto al manejo de información, pérdida de información, identificación de los procesos, evitar horas extras en los trabajadores.

3.3 Diseño de Salida

Es muy común que para los usuarios, la característica más importante de un sistema de información, es la salida que produce.

Si la salida no es de calidad, pueden pensar que todo el sistema no es necesario, bueno o adecuado y por tanto evite su utilización o incluso ocasiones errores (sabotaje) generando que el sistema falle.

3.3.1 Diseño lógico de salidas

El concepto salida, se aplica a cualquier tipo de información producida por un sistema informático o computacional, ya sea: impresa, desplegada, verbal, multimedia o a la red.

Cuando los desarrolladores, diseñan la salida, seleccionan métodos y técnicas para representar la información y crean documentos, informes u otros formatos que contienen en sí, la información que será producida por el sistema.

3.4 Diseño de entradas

El diseño de la entrada es la liga que une en primera instancia, al sistema de información con el mundo de los usuarios.

Algunos aspectos del diseño de entradas, variarán dependiendo de sí: el sistema estará orientado a operar por lotes (batch) o en línea (textual) o por una interfaz gráfica de usuario.

Objetivos de la entrada

El diseño de la entrada consiste en desarrollar especificaciones y procedimientos para:

1. La obtención y preparación de los datos, el cual es el conjunto de logística de obtención para poner los datos de la operación en una forma utilizable para su procesamiento.
2. La introducción de datos, es decir; capturar o introducir la información o los datos en la PC, en la forma más óptima, para su posterior procesamiento.

Entonces, la entrada de datos se puede efectuar por la lectura por la computadora de un documento impreso o perforado, o se puede realizar cuando los usuarios tecleen en forma interactiva los datos directamente al sistema o incluso se digitalice, o rastree un documento, para una posterior edición o análisis.

Considerando lo anterior, es importante tomar en cuenta los siguientes objetivos, como guía para el diseño de las entradas:

❖ **Control de cantidad de entradas**

Se debe hacer por las siguientes razones:

1. La preparación de los datos y su introducción a la computadora dependen, en general del personal. Esto repercute en el costo de mano de obra.
2. El proceso de entradas de datos es, en general, muy lento.

❖ **Evitar la demora**

Cuando el proceso se demora debido a la preparación o entrada de los datos, se generan en él, cuellos de botella. Evitar esto, deberá ser un objetivo fundamental del analista.

❖ **Evitar errores en la introducción de los datos**

La tasa de errores depende de la cantidad de datos, entre menos datos, menos errores. Entonces los errores al introducir los datos, deberán rechazarse por medio de técnicas o procesos automáticos de validación de la entrada. Esto es uno de los puntos fundamentales del concepto de integridad de la información. Aquí es importante recalcar la segunda regla de la Informática: **“Si entra basura, sale basura”**.

❖ **Evitar los pasos extra**

En una operación de miles de entradas, por día o por proceso, el efectuar una operación de más o de menos, puede hacer la diferencia entre un sistema eficiente o no, rentable o no, etc.

❖ **Ergonomía (Mantener sencillo el proceso)**

Los sistemas mejor diseñados se adaptan a los trabajadores que los utilizarán (y no al contrario), creándoles las menores molestias.

❖ **Permitir controles de acceso**

Las entradas al sistema deberán contar con métodos de control de accesos: Passwords o contraseñas que permitan que solo las personas adecuadas ingresen al sistema en su momento y lugar.

❖ **Crear procesos que permitan la seguridad**

Con guías para la captación de datos se debe comenzar captando solo aquellos que deben ser entradas. Existen dos tipos de datos que deben ser principalmente de entrada, cuando se procesan las transacciones.

1. Datos de Identificación. El elemento dato que identifica el objeto que se está procesando (los datos que identifican un registro se conocen como llave, clave o key).
2. Datos variables. Son aquellos datos de conceptos, que cambian para

cada transacción que se maneja o decisión que se lleva a cabo.

3.5 Creación de Base de Datos

Los sistemas de base de datos se diseñan para manejar grandes volúmenes de información normalizada, la información puede almacenarse permanentemente en forma centralizada o distribuida, puede ser explotada por diversos programas de aplicación, la finalidad de crear una base de datos es reducir la redundancia y evitar la inconsistencia; a los datos se les puede aplicar restricciones de seguridad, puede conservarse mejor la integridad de los mismos.

La normalización exige en efecto que algunos atributos o dominios aparezcan en más de una relación con el fin de identificar y efectuar las asociaciones entre ellas. Esta duplicidad no representa necesariamente un aumento de las necesidades de espacio de almacenamiento, ya que la normalización atañe a las estructuras lógicas (es decir, la vista de los datos propia del usuario, y no a la forma como los datos son almacenados físicamente).

El sistema manejador de la base de datos tendrá que evitar esta redundancia física. La normalización no comprende aspectos de la organización física de la base de datos, únicamente trata con el ámbito del diseño lógico. Las propiedades de la normalización dependen exclusivamente de la semántica de la base de datos y de las dependencias funcionales entre sus dominios, tal como los mira el diseñador de la base de datos. La normalización tiene una importancia primordial en la administración y diseño de la base de datos.

Posibles problemas al no normalizar

Si los atributos repetidos no se separan de acuerdo a la forma mencionada, al diseñar originalmente las relaciones, se corre el peligro, de que la

evolución de las bases de datos, exija esta separación más adelante.

Si se presenta la necesidad de independizar los atributos repetidos con posterioridad a la elaboración de los programas que los utilizan, estos programas tendrán que ser modificados y sometidos a nuevas pruebas.

Este costo de mantenimiento se eliminará con la normalización y la adecuada independencia de los datos en el software.

Se puede concluir entonces que la normalización mejora la independencia lógica de los datos, es decir, el aislamiento de un programa de aplicación respecto a cambios en las estructuras de los datos.

3.5.1 Estructura de modelo relacional para SAM

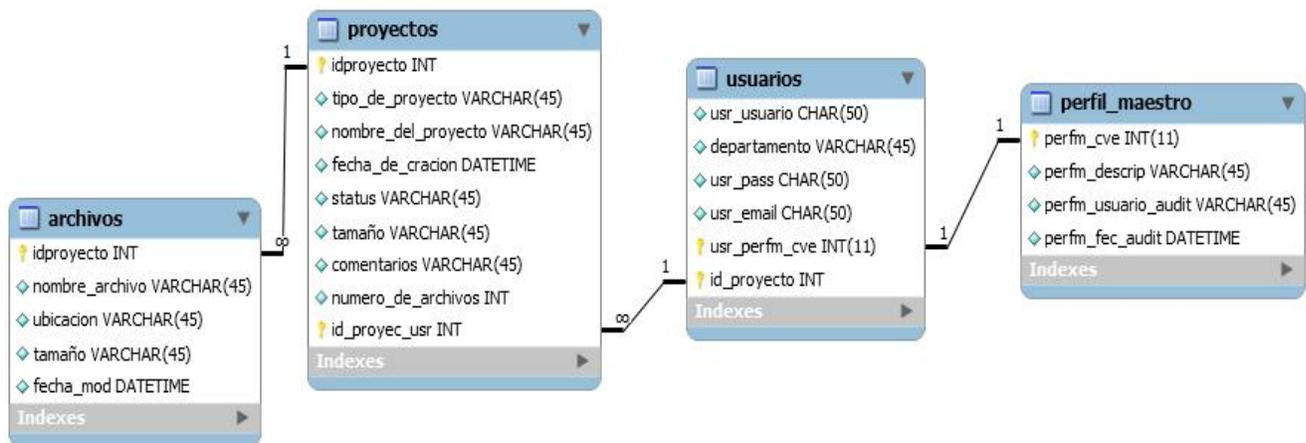


Figura 7.- Modelos relacional de SAM

Para poder diseñar nuestra base de datos debemos conocer los datos que vamos a guardar en ella. Primero debemos guardar datos de los proyectos que se van a trabajar, con sus características y en general datos que los describan. A su vez tendremos la tabla archivos con información específica de los mismos, esta tabla nos servirá para complementar la información contenida en proyectos esto referente a números de archivos que contenga cada proyecto.

También tenemos la tabla de “usuarios” que contiene los datos personales de los usuarios que van a ser responsables del sistema.

Para esto debemos crear una tabla que contenga estos datos, la llamaremos “usuarios”. En la tabla “perfil_maestro” vamos a guardar los detalles de los permisos que tenga cada usuario de SAM.

3.5.2 Arquitectura General de SAM

La arquitectura está basada en el modelo cliente-servidor, donde todas las transacciones serán procesadas y administradas por el servidor central, mientras una aplicación cliente corriendo en cada estación de trabajo se encarga de brindar una interfaz de fácil uso.

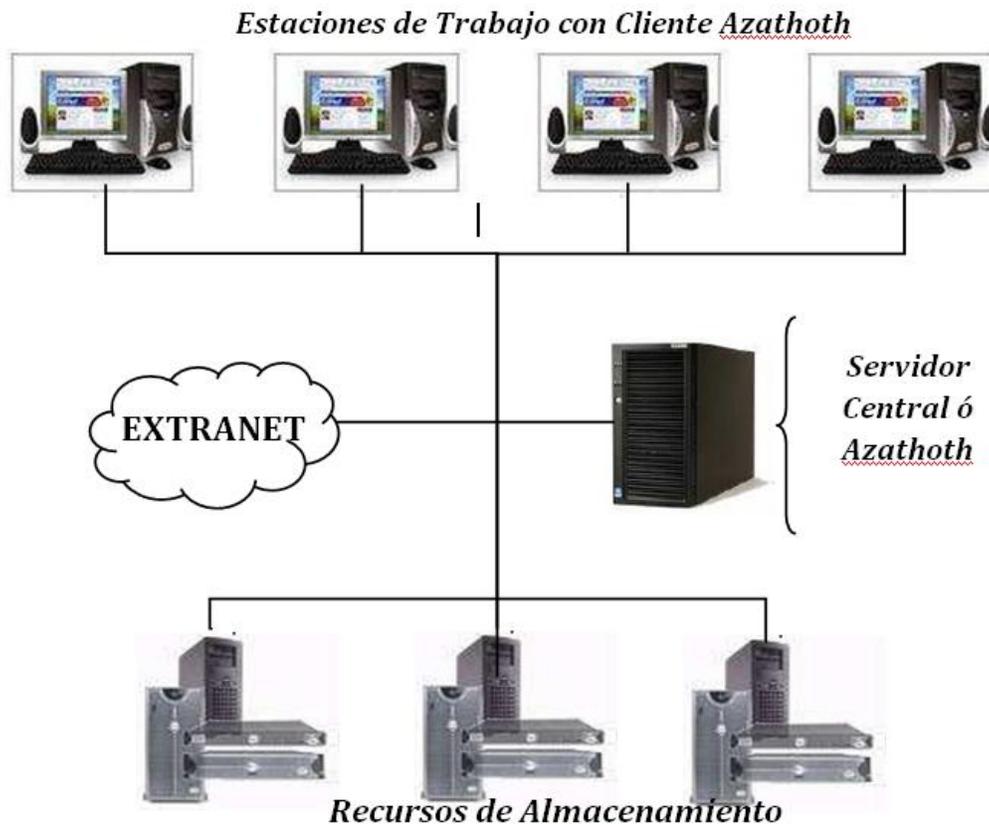


Figura 8.- Arquitectura General de SAM

3.5.3 Servidor Central

El servidor central es el encargado de registrar y verificar todas las acciones realizadas en el acervo digital, además de actualizar el metadata¹⁰ después de cada cambio. Es importante aclarar que el acervo digital será almacenado en los recursos destinados para este propósito por lo que en el servidor solamente se almacenarán los registros de los cambios realizados en el acervo, los archivos de procesamiento por lotes -scripts- y demás información necesaria que permita recuperar la edición final del material al obtener los archivos de acervo originales. De forma general nos referimos a él como servidor Central cuando en realidad es un conjunto de varios servicios agrupados en un solo servidor.

3.5.4 Servicio de Administración de Metadata (SAM)

Este servicio es el encargado de almacenar, actualizar e informar sobre el estado del metadata de cada uno de los recursos de acervo. Cada vez que se lleve a cabo una actualización en la edición de algún elemento del acervo, el SAM guardará la información sobre esta edición de forma que pueda ser reconstruida, por ejemplo: Al crearse un nuevo recurso de acervo a través del Northlight se almacenarán el EDL's, los valores iniciales del escáner como son: el formato, el crop, la orientación, la calibración, la densidad, el lugar de almacenamiento, etc. O bien al momento en que algún elemento del acervo sea editado en la sala de compuestos el SAM almacenará los archivos de procesamiento por lotes utilizados en su edición.

El SAM también será el encargado de suministrar la información sobre el metadata requerida por los otros servicios del Servidor SAM. Lo anterior se lleva a cabo mediante un servidor de bases de datos MySQL siguiendo el modelo relacional para así garantizar la consistencia y consolidación de los datos.

¹⁰ Se conoce como Metadata a los datos sobre los datos y se frecuentemente se utilizan para su clasificación.

3.5.5 Clientes SAM

Los clientes SAM trabajan sobre cada una de las estaciones de trabajo, su principal función es servir como centro de control para el Servidor SAM además de ser el medio por donde la información enviada por el Servidor es mostrada a los usuarios.

También es el responsable de brindar herramientas que faciliten la labor de los usuarios y no impliquen la necesidad de consultar información almacenada en el Servidor SAM o en los servidores dedicados al Almacenamiento, por ejemplo, mostrar la documentación de SAM.

Junto con los clientes SAM serán instalados algunos scripts que se integrarán a las distintas aplicaciones -Maya, Shake, etc.-y servirán para registrar los cambios hechos en la edición de cada elemento del acervo digital, con el fin de mantener actualizada la información en el servidor.

Es necesario que el desarrollo de los Clientes SAM sea multi-plataforma por lo que se utilizará la versión Open Source de la librería Qt.¹¹Lo anterior tiene como consecuencia que SAM se distribuya también bajo ésta misma licencia. Si se requiere que SAM sea un software comercial es necesario adquirir la versión Comercial de Qt.

Qt garantiza la total integración con las plataformas Linux, MacOS y Windows y utiliza el framework natural de cada una de las plataformas logrando así una gran estabilidad y aprovechamiento de los recursos en cada estación de trabajo.

¹¹ <http://www.trolltech.com/products/qt>

3.6 Desarrollo

El desarrollo de SAM será modular. Las distintas herramientas y servicios se liberarán paulatinamente y se someterán a la evaluación por parte de los usuarios. Todas las peticiones de los usuarios serán atendidas y los errores reportados serán corregidos hasta llegar a una versión estable de las mismas. Una vez obtenidas las versiones estables de todas herramientas y servicios necesarios para el funcionamiento de un módulo éste será liberado para su evaluación. El tiempo estimado para la primera versión totalmente funcional de SAM es de aproximadamente 24 meses.

3.6.1 Módulos de Software

La división por módulos se realizó con base en el flujo de trabajo y los servicios prestados por el servidor SAM, de tal manera que, al término del desarrollo de un módulo, los usuarios involucrados dentro de cierto conjunto de procesos contiguos en el flujo de trabajo se encuentren en la posibilidad del el uso de los servicios y herramientas destinadas para ese propósito.

Esto tiene como consecuencia que difícilmente la funcionalidad de algún servicio del servidor será total, sino hasta que la totalidad de los módulos relacionados con él sean desarrollados. Sin embargo, un desarrollo modular basado sobre procesos contiguos en el flujo de trabajo, brinda la oportunidad de un aprovechamiento inmediato de los servicios y herramientas, además el hecho de proximidad en los procesos permite incorporar, aunque sea localmente, una mayor eficiencia en la transferencia de información y por consecuencia en el flujo de trabajo.

Éste diseño de desarrollo también tiene la ventaja de permitir la evaluación continua del software y así permitir la corrección de errores de manera inmediata, garantizando que el resto del desarrollo prosiga sobre una base sólida. Los módulos de SAM se muestran a continuación:

3.6.2 Desarrollo del Núcleo del Servidor.

En esta etapa se implementarán las herramientas que garanticen un mínimo funcionamiento de los diversos servicios de SAM, como son: Comunicación entre el servidor y los recursos de almacenamiento, gestión de usuarios, comunicación con el servidor de bases de datos, inicialización de proyectos y comunicación cliente-servidor.

3.6.3 Recepción de EDL's y Creación de la estructura de Carpetas.

En este módulo se implementaran las herramientas que permiten la verificación de los EDL's, se crearán las bases de datos encargadas de almacenarlos y se desarrollarán las aplicaciones necesarias para la creación de la estructura de carpetas indispensable para el almacenamiento del acervo. También en este proceso se desarrollarán las interfaces para la recepción de referencias y su respectiva actualización en el Metadata.

3.6.4 Herramientas para el Data Wrangler

En esta etapa de desarrollarán las herramientas que faciliten el trabajo del Data Wrangler, es decir, las interfaces que permitan el copiado, traspaso y eliminación de archivos en los medios de almacenamiento.

Para lograrlo, es necesario el desarrollo de los programas encargados de la administración del espacio en los servidores, búsqueda de la mejor ruta disponible para el copiado, verificación del correcto traspaso de información y gestión de errores.

En esta etapa también se desarrollará la base de datos para el registro de modificaciones en el acervo digital y la respectiva actualización en el Metadata.

3.6.5 Utilidades para el Northlight

En este módulo se incluyen las herramientas que permiten el eficiente uso del escáner. En él se desarrollarán las aplicaciones para la solicitud de scan, la recepción y conversión de EDL's, recepción de referencias y se elaborarán las bases de datos que guardarán información sobre la edición de los eventos con el fin de reutilizarla en caso de ser necesario algún reescan. También se actualizará el metadata y se guardarán los datos necesarios para la generación de estadísticas.

3.6.6 Administración de Vfx

En esta etapa se desarrollarán las herramientas que permitan un adecuado manejo de la información sobre los Vfx. Se diseñarán las bases de datos que alojarán la información necesaria sobre cada uno de los Vfx (Técnicas utilizadas, FrameIn, FrameOut, Comentarios, Estado, ScreenShot, etc.), se diseñarán las aplicaciones necesarias para la supervisión de los efectos y se registrará el avance en cada uno de ellos.

3.6.7 Utilidades para Ilusión

El objetivo de este módulo es brindar un seguimiento detallado sobre el avance en cada uno de los efectos de compuestos, con el fin de brindar la información necesaria para el adecuado desarrollo de los mismos, además de brindar un sistema de respaldo que permita su recuperación en caso de contingencia o bien si es necesaria su reutilización en proyectos posteriores. Esto se logrará mediante el desarrollo de archivos de procesamiento por lotes que se integren al entorno de trabajo e informen sobre el avance en cada Vfx. También se desarrollarán aplicaciones que permitan el intercambio de información con el administrador de Vfx, la recepción de referencias y la actualización del metadata.

3.6.8 Administración de Proyectos

Este módulo es el encargado de brindar al usuario una aplicación que permita la navegación a través de los proyectos, mostrando información general acerca de ellos y a la vez será aquella que permita el acceso a la información más detallada sobre cada uno, es decir, hará la veces de ventana principal permitiéndonos acceder a distintos menús para consultar la información anteriormente detallada. En esta etapa también se generarán las bases de datos necesarias para almacenar los registros con información general sobre los proyectos.

3.6.9 Generación de Estadísticas

En este módulo se desarrollarán los algoritmos necesarios para la obtención de diversas estadísticas que nos informen principalmente el tiempo consumido en cada departamento, la productividad de cada área, la cantidad de recursos de almacenamiento ocupados por proyecto, la cantidad de tiempo ocupada en el tráfico de información y todas aquellas que se crean necesarias para mejorar el flujo de trabajo. También se desarrollarán las aplicaciones necesarias para mostrar la información anterior.

3.6.10 Importación y Exportación a Extranet

Este módulo es el encargado de extraer del servidor SAM la información que se mostrará a los clientes y la organizará para su transferencia y visualización vía la extranet. De igual manera se desarrollarán los servicios necesarios para la recepción de referencias, EDL's y demás información relevante por parte del cliente vía la extranet.

FASE IV
IMPLEMENTACIÓN
Y
OPERACIÓN

Capítulo IV

IMPLEMENTACIÓN Y OPERACIÓN

4.1 Modelo de Interfaz

El Modelo de Interfaz establece un vínculo entre el Usuario y el Analista, mostrando gráficamente cada Caso planteado.

Consta principalmente de la definición de las interfaces principales que participarán en la ejecución de un Caso cuando el sistema exista.

Por interfaces nos referiremos a Pantallas, Reportes o llamadas a otros sistemas. Sin embargo aplica con mayor frecuencia a Pantallas.

Es claro que un sistema puede estar compuesto por un gran número de pantallas, reportes y llamadas a otros sistemas, sin embargo, el modelo estará compuesto sólo por las interfaces más significativas.

Cabe mencionar que el formato y la presentación de la información (color de pantallas, distribución de información, reportes) son presentados de acuerdo a los requerimientos de los usuarios.

4.2 Interfaz de Usuario

Para ingresar al sistema se debe de abrir un explorador de Internet, y escribir en la barra de direcciones lo siguiente: www.ollin.com.mx/SAM

Para que nos aparezca la pantalla principal como se muestra a continuación:



Figura 9.- Interfaz Principal

Validación de Usuarios

Primero debemos asegurarnos de que sólo el personal autorizado (administradores, operadores o personas con autorización), tenga acceso y permiso para la modificación y manipulación del sistema.

Es aquí donde hacemos uso de la contraseña (password) que asignamos para acceder al sistema SAM.

Para lograr esta validación creamos una página PHP que lo haga. Esta página solamente pedirá la introducción de una contraseña mediante un <<input de tipo “password” para impedir que los demás puedan verla.

Esta contraseña es enviada a una página PHP que se encargará de validarla.

Validar Contraseña

Para validar la contraseña necesitaremos hacer una consulta a la base de datos. Esto lo haremos con variables de sesión de PHP, tomamos al usuario y contraseña y verificamos que exista en la base de datos, si existe lo dejamos pasar al sistema con los permisos correspondientes, de lo contrario lo regresamos a la pantalla de inicio.

El código para hacer la validación de contraseña se muestra a continuación.

```
$usuario=$_POST['nombre'];
$pass=$_POST['leo'];
$rs = mysql_query("SELECT * FROM usuarios
WHERE usr_usuario='$usuario' and usr_pass='$pass'");
    if (mysql_num_rows($rs)!=0){
        // usuario y contraseña válidos
        // defino una sesion y guardo datos
        $_SESSION['usuario']=$usuario;
        header ("Location:./ menu.PHP");
    }else {
        // si no existe le mando otra vez a la portada
        header("Location:./ index.PHP");
    }
}
```

Es importante mencionar que se crea una sesión en PHP, como medida de seguridad para evitar, que accedan al sistema vía URL, sin identificarse.

Una vez identificado, la siguiente pantalla nos muestra como se despliega la información de los proyectos en caso de ser empleado.

4.3 Menú Principal



Figura10.- Menú

Esta pantalla es la que nos va a permitir ver los datos del proyecto, que seleccionemos, dependiendo del servidor.

Para lograr esto utilizamos varias funciones desarrolladas por nosotros, junto con algunas clases para facilitar la construcción de las interfaces. Algunas de las funciones que utilizamos se muestran a continuación.

Se creó una clase para todas las funciones ftp, para facilitar el empleo de las mismas.

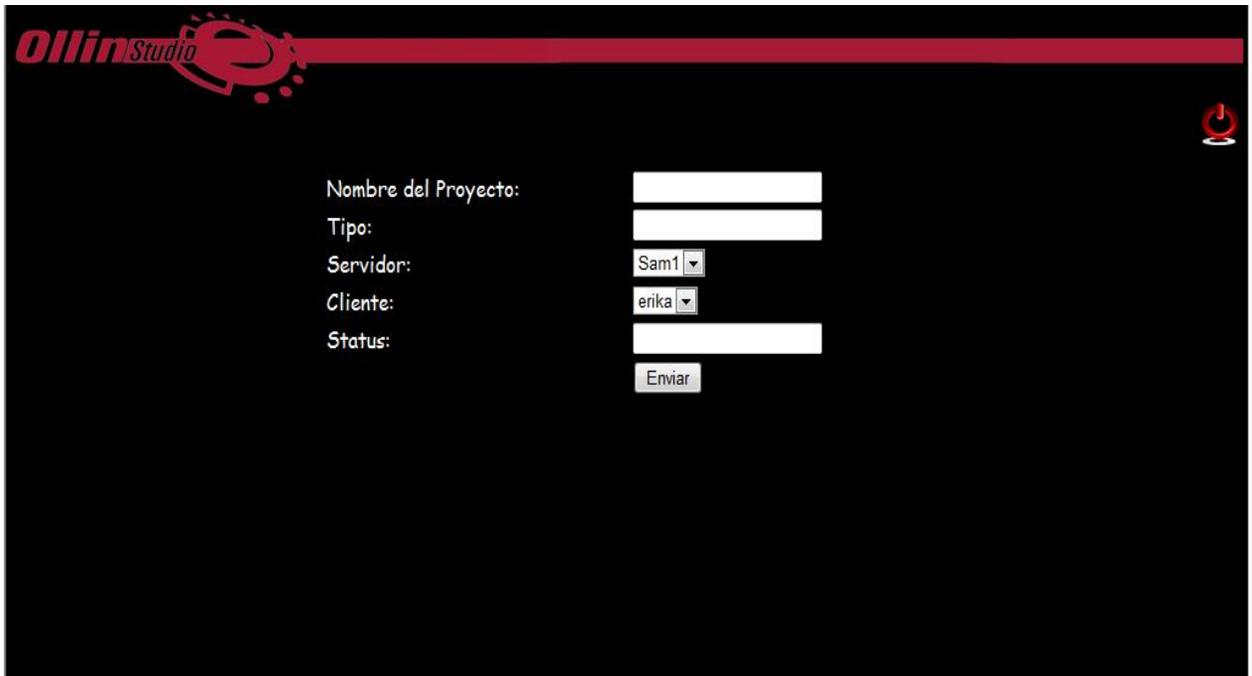
```
private function conectarFTP()
{
    $sid=false;
    $ftpLogging=false;

    $sid=ftp_connect($this->servidor);
    $ftpLogging=ftp_login($sid, $this->usuario, $this->clave);

    return $sid;
}
```

Esta función lo que hace es conectarse al servidor ftp, y el valor que regresa es un identificador de conexión, para poder ejecutar funciones de PHP en el servidor ftp.

4.4 Captura y Edición de Proyectos y Archivos



The screenshot shows the OllinStudio web interface. At the top left is the 'OllinStudio' logo. The main content area is black with white text and form elements. The form includes the following fields and controls:

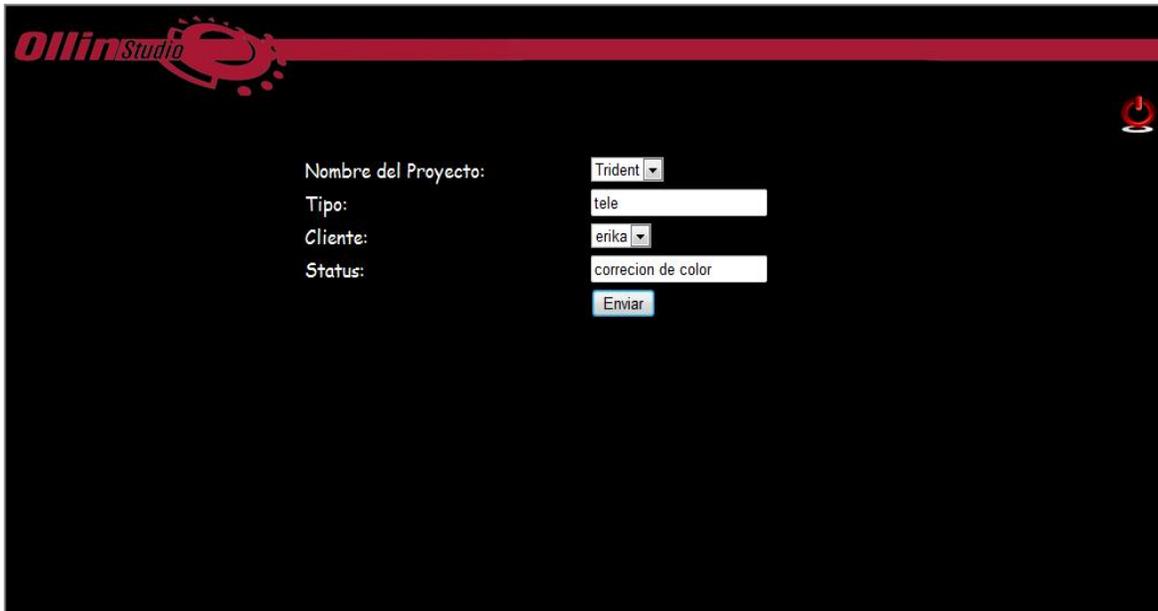
- Nombre del Proyecto:
- Tipo:
- Servidor:
- Cliente:
- Status:
- Enviar:

Figura 11.- Alta de proyecto

Esta es la pantalla para dar de alta un nuevo proyecto, es importante mencionar que el cliente debe estar dado de alta para poder ingresar el proyecto.

Esto se debe por que se consulta la base de usuarios para desplegar los clientes, y se va a guardar los datos en la tabla de proyectos.

Editar Proyecto



Nombre del Proyecto: Trident

Tipo: tele

Cliente: erika

Status: correccion de color

Enviar

Figura 12.- Edición de proyecto

Esta pantalla es para editar un proyecto. La cual vamos a estar utilizando para cambiar el status del proyecto.

De esta pantalla se crea la siguiente tabla de consulta:

ATRIBUTO	Tabla de origen
Nombre_del_proyecto	Proyectos
Tipo	Proyectos
Cliente	Usuarios
Status	Proyectos

Tabla 8.- Consulta estatus del proyecto

Algunas funciones ocupadas en esta pantalla son:

```
<?PHP
require 'conexión.PHP';
$res = mysql_query("select * from proyectos where idproyecto='$id' ");
?>
<select size="1" name="usuarios">
<?
while ($row = mysql_fetch_array($res))
{
?>
  <option value="<?PHP echo $row['nombre_del_proyecto']?>"><?PHP
echo $row['nombre_del_proyecto']?></ option>
<?
}
?>
</ select>
```

Aquí nos despliega los proyectos según el servidor que se eligió, donde podremos seleccionar el proyecto, y este a su vez nos muestra los datos de manera general que describen a ese proyecto, como nombre de proyecto, nombre del cliente al que pertenece, fecha de creación y el estado en el que se encuentra.

Eliminar Proyectos

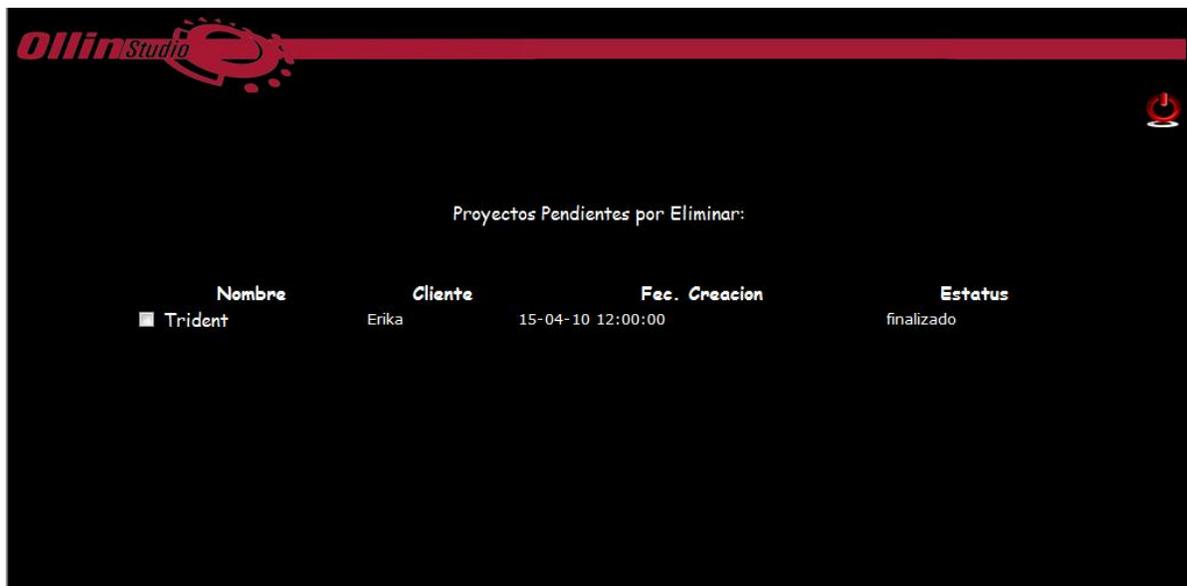


Figura 13.- Eliminar Proyectos

Esta pantalla nos despliega los proyectos que están pendientes por eliminar, solo nos va mostrar aquellos proyectos que en el status estén como finalizado, ya que aquí es donde vamos a impedir que se borren proyectos o archivos que no se puedan eliminar.

```
<?PHP
require 'conexión.PHP';
$res = mysql_query("select * from proyectos where status ='finalizado' ");
?>
<select size="1" name="usuarios">
<?
while ($row = mysql_fetch_array($res))
{
?>
  <option value="<?PHP echo $row['nombre_del_proyecto']?>"><?PHP echo
  $row['nombre_del_proyecto']?></ option>
<?
}
?>
</ select>
```

Subir Archivo

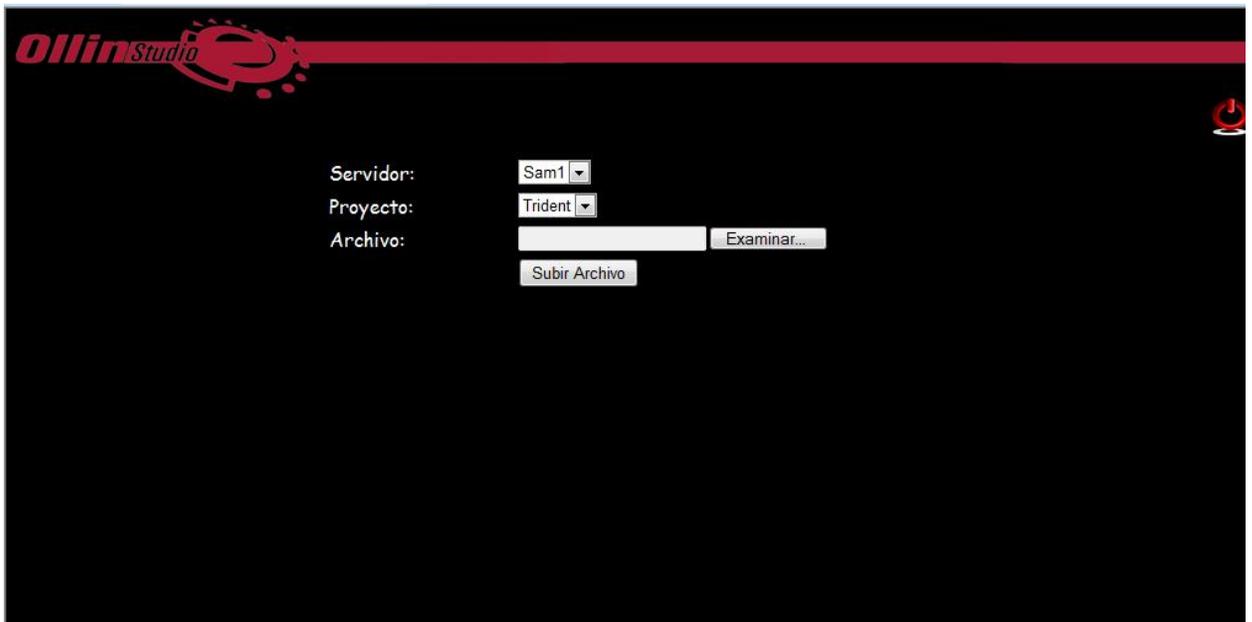


Figura 14.- Subir Archivo(s)

Esta pantalla es para agregar archivos que pertenezcan a un proyecto,

dándonos la opción de escoger el servidor al que queremos establecer como ubicación física y asignarlo a algún proyecto existente.

```
<?PHP echo $server;
require 'conexión.PHP';
$res = mysql_query("select * from proyectos");
?>
<select size="1" name="proyecto">
<?
while ($row = mysql_fetch_array($res))
{
?>
  <option value="<?PHP echo $row['nombre_del_proyecto']?>"><?PHP echo
$row['nombre_del_proyecto']?></ option>
<?
}
?>
```

Con el código antes descrito nos permite hacer una consulta a la base de datos de proyectos, que nos despliega como resultado los proyectos que se encuentran en la base de datos en estado finalizado ya que solo en este estado se podrán eliminar los archivos.

4.5 Administración de Usuarios empleados y clientes

A continuación mostraremos la administración de usuarios, para este caso en el sistema nos va aparecer una pantalla que nos va mostrar dos opciones, la administración de empleados y la administración de clientes.

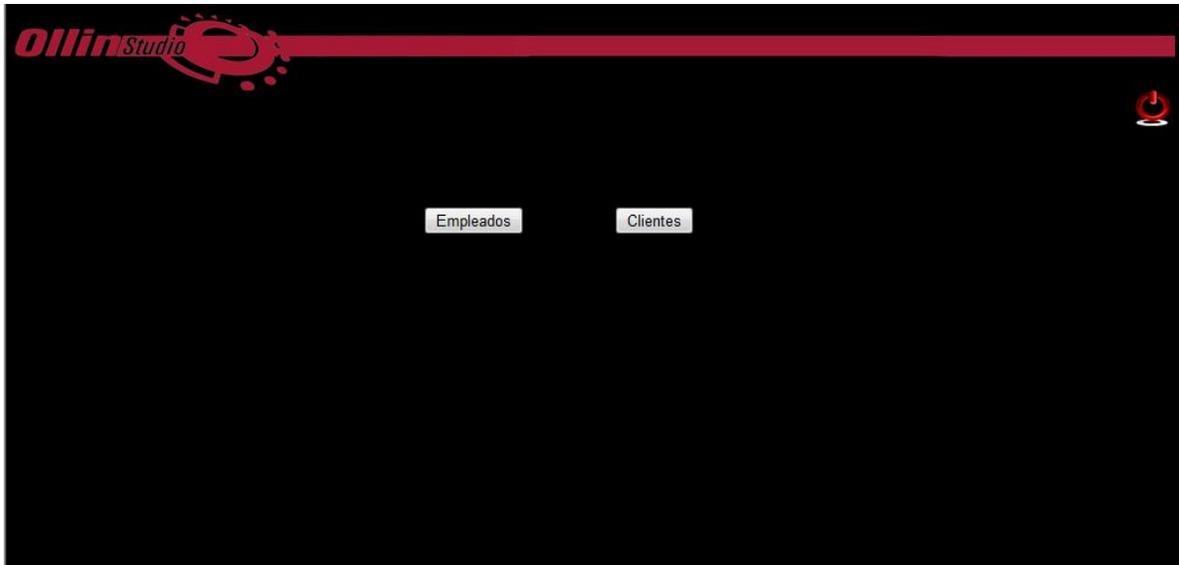


Figura 15.- Separación de usuarios

Para empleados.

Despliegue de datos de los usuarios

En esta pantalla nos va a desplegar la información de los empleados según este seleccionado en la opción. Contamos con la opción de eliminar usuario, modificar y agregar un usuario nuevo. Para la opción eliminar no nos despliega otra pantalla, elimina el usuario que esta seleccionado.

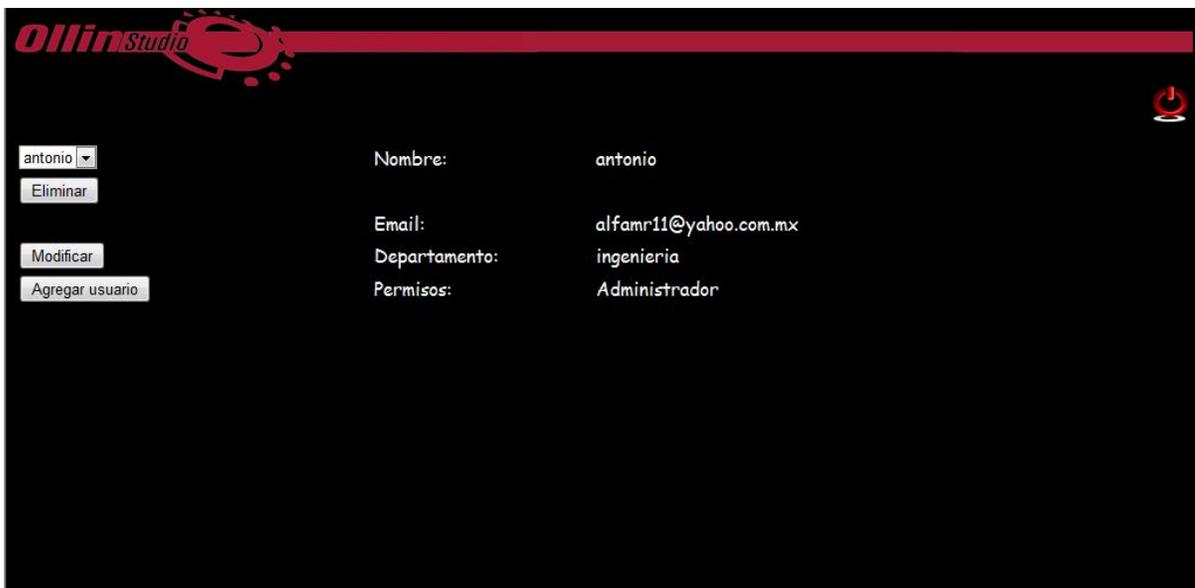


Figura 16.- Eliminar Usuario

En esta pantalla se conecta a la base de datos, para desplegar los usuarios existentes, desplegándolos en una ventana de opción, lográndolo con el siguiente código.

```
$res = mysql_query("select * from usuarios");  
?>  
<select size="1" name="usuarios">  
<?  
while ($row = mysql_fetch_array($res))  
{  
?>  
    <option value="<?PHP echo $row['usr_usuario']?>"><?PHP echo  
    $row['usr_usuario']?></ option>  
<?  
}  
?>  
</ select>
```

Modificar

En la pantalla de modificar usuario, lo que nos va a desplegar son los datos del usuario a modificar, en el caso de la contraseña no nos la va a desplegar, pero si el campo esta vacio dará por hecho que se mantiene la misma.

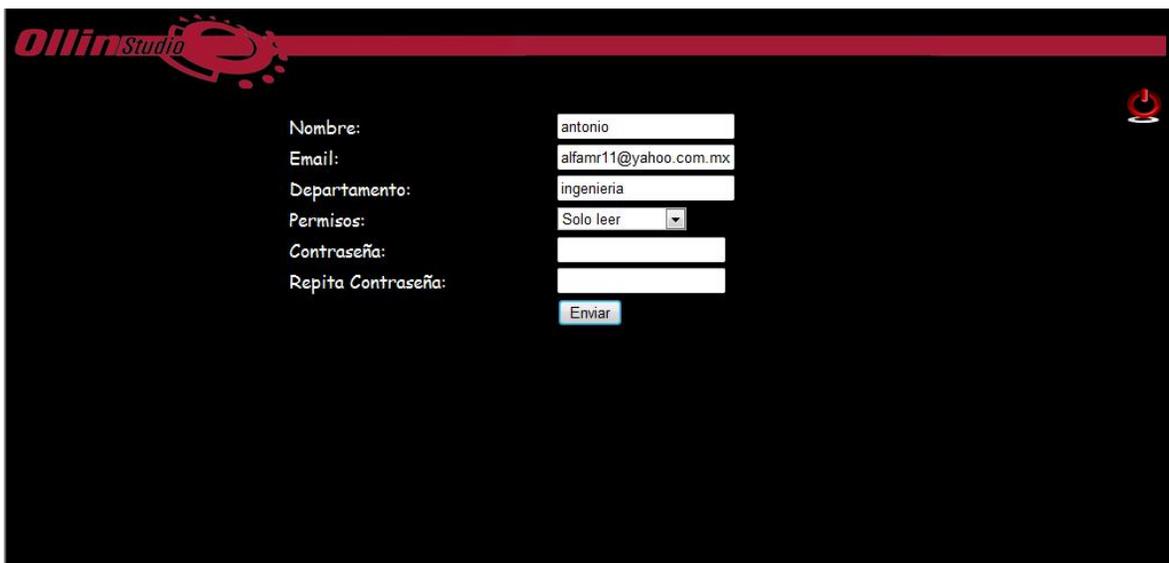


Figura 17.- Modificar usuario

```
require 'conexión.PHP';
```

```
$res = mysql_query("select * from usuarios where usr_usuario='$usuario' ");
```

```
$row = mysql_fetch_array($res);
```

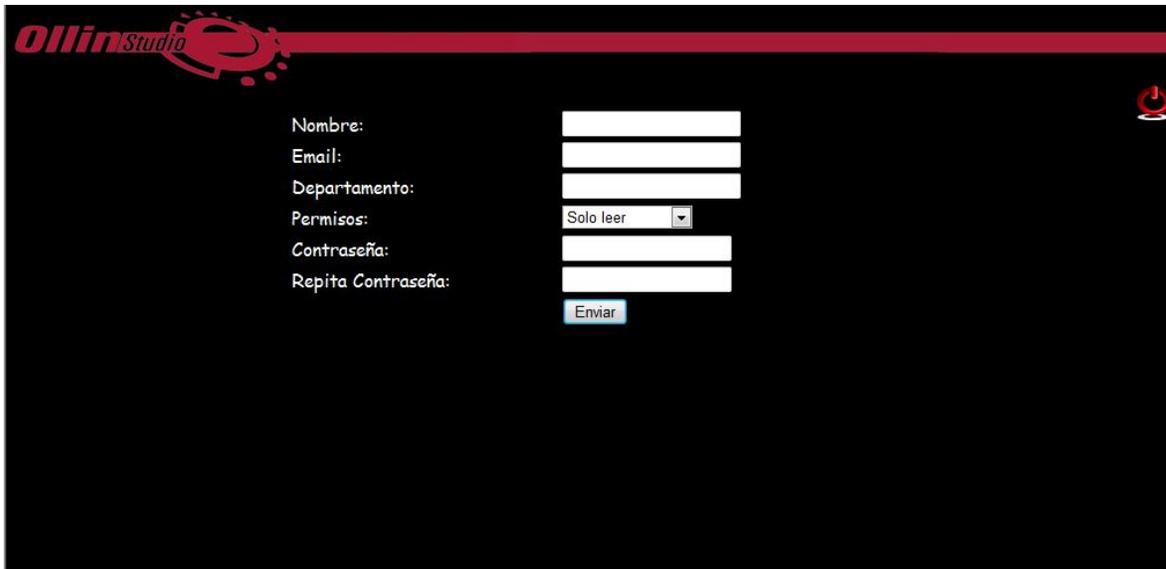
Con esta pantalla se genera la siguiente tabla de consulta.

ATRIBUTO	Tabla de origen
usr_usuario	Usuarios
usr_email	Usuarios
Departamento	Usuarios
Perfm_descrip	Perfil_maestro
Usr_pass	Usuarios

Tabla 9.- Consulta para modificar usuario

Agregar Usuario

En esta pantalla nos va pedir datos necesarios para dar de alta un usuario, la cual vamos a insertarlo en la tabla de usuarios.



Ollin Studio

Nombre:

Email:

Departamento:

Permisos: Solo leer

Contraseña:

Repita Contraseña:

Enviar

Figura 18.- Agregar usuario

Administración de clientes.

En esta pantalla es muy parecida a la pantalla que nos muestran los datos de los empleados, con la diferencia de que aquí solo nos va a mostrar a los clientes, y por lo cual algunos datos cambian.



Figura 19.- Administrar clientes

Al igual que en los empleados, para los clientes tenemos las mismas opciones eliminar, modificar y agregar un nuevo usuario.

Modificar cliente.

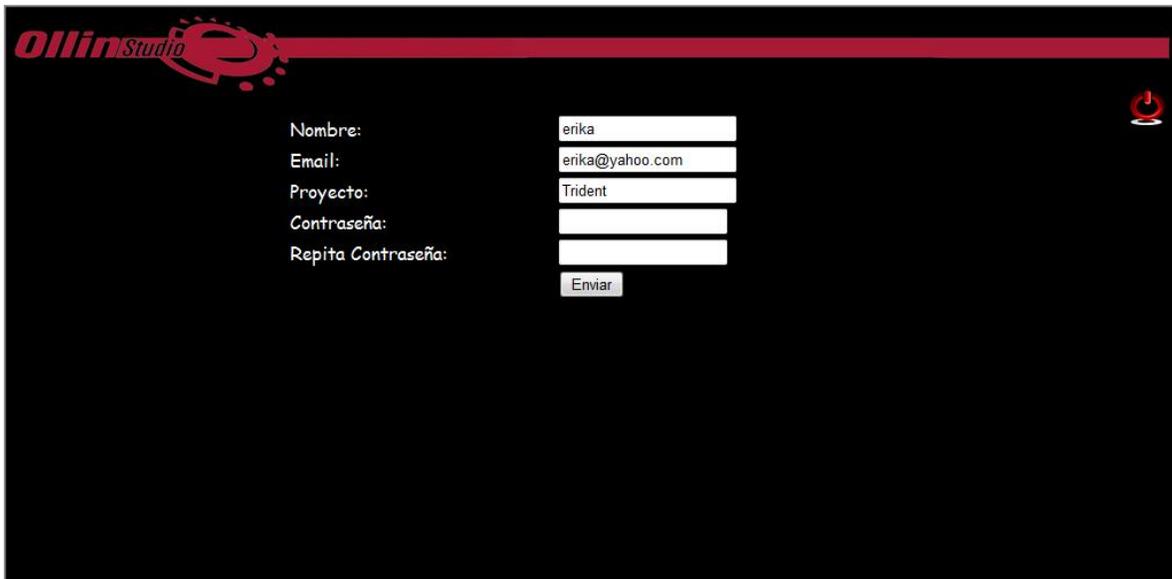


Figura 20.- Modificar cliente

En el caso de clientes no se le modifican los permisos porque siempre van a ser los mismos, al igual que para los empleados y el campo de contraseña esta en blanco se va mantener la contraseña que tenían.

Agregar cliente.

Esta pantalla nos pide datos para poder dar de alta un cliente, pero en el caso de los clientes se omiten los permisos, porque un cliente solo va tener los permisos de solo ver. Le vamos asignar una contraseña nosotros para que posteriormente si gusta el cliente cambiarla.

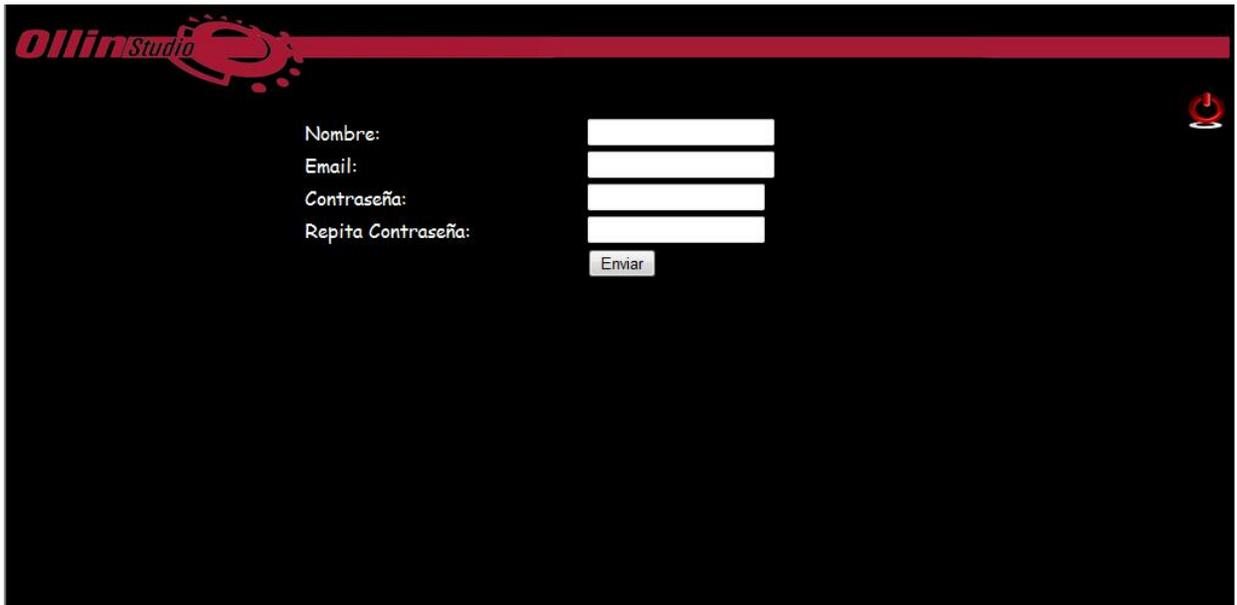


Figura 21.- Agregar cliente

En el caso de que al dar de alta un cliente y no se deje vacío, el sistema insertara como contraseña el nombre del cliente.

Menú clientes.

En el caso de que sea un cliente quién entre al sistema todo cambia, solo va poder ver su proyecto y cambiar su contraseña.

En esta pantalla le va mostrar algunos datos de su proyecto, va tener la opción de cambiar su contraseña y ver los avances de su proyecto.



Figura 22.- Menú clientes

Consulta ver avances.

En esta pantalla lo que le va desplegar son los videos que Ollin Studio le genera de avances, para que el cliente pueda verlo, por lo que el cliente podrá, verlo pero además descargarlo si así lo requiere.

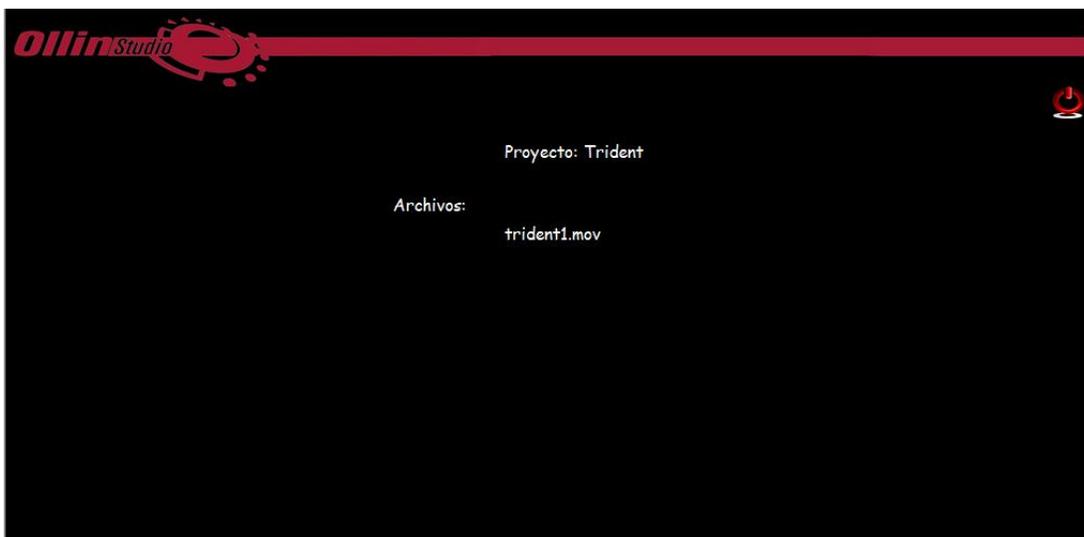
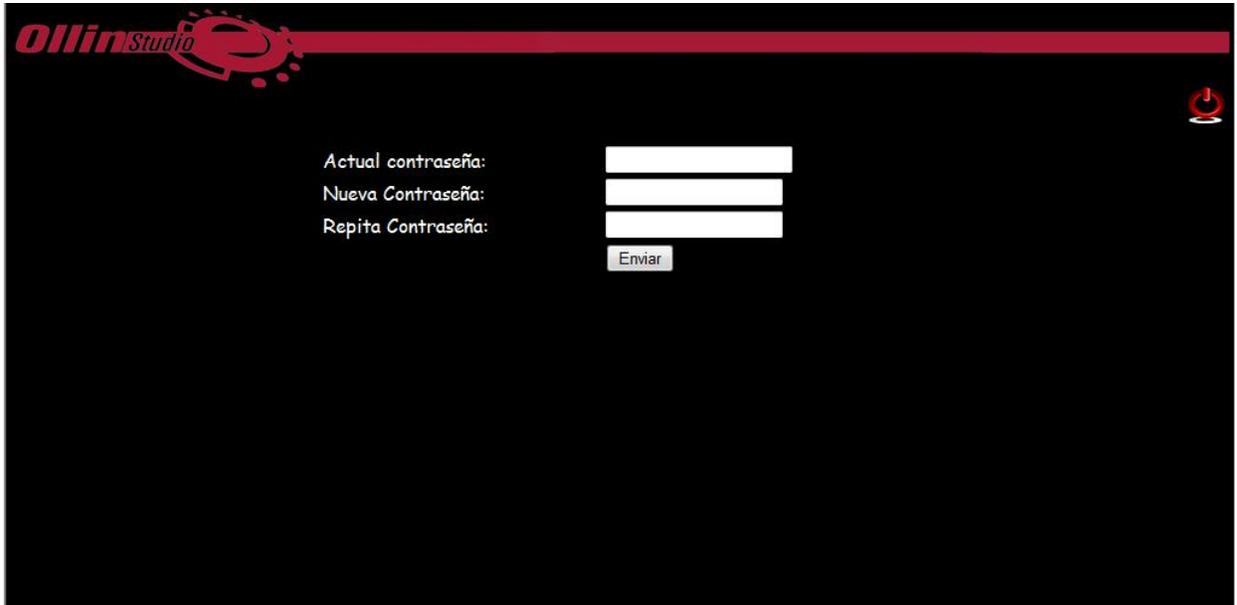


Figura 23.- Consulta de estatus del proyecto

Cambiar contraseña.

Esta pantalla le da la oportunidad de cambiar su contraseña al cliente, por si no le gusta la que nosotros le dimos.



Actual contraseña:

Nueva Contraseña:

Repita Contraseña:

Figura 24.- Cambiar contraseña

CONCLUSIONES

Con SAM en operación se mejoro el manejo del acervo digital, además de contribuir a lograr una mayor productividad en los procesos de Ollin Studio, facilitando además la forma de trabajo.

En la **Tabla 10** se muestran los problemas frecuentes con los que se enfrentaba la empresa antes de SAM y los beneficios que obtuvo en costo y tiempo después de SAM.

Los montos de esta tabla fueron calculados de acuerdo a los sueldos que se manejan actualmente en la empresa Ollin Studio.

TABLA DE CONCLUSIONES

Problemas	Costo antes del sistema SAM	Costo después del sistema SAM
Evitar la duplicidad de información.	\$ 5,000.00 (3 horas x semana)	\$0.00
<i>Al evitar duplicidad de información se evita pagar horas extras al empleado que es encargado de depurar y liberar espacio en servidores de almacenamiento, el sueldo base de este empleado es de \$25,000.00. Se hizo una estimación de horas extras que un empleado llega a laborar.</i>		
Pérdida de información al eliminarla y tener que volver a generar la información, provocando que no entregue el material a tiempo.	60% del costo total del contrato.	\$ 0.00
<i>Al generar nuevamente la información del proyecto eliminado, implica pagar horas extra a varios trabajadores de acuerdo al proceso y status en que se encuentra el proyecto en cuestión. Cabe mencionar que los contratos se facturan arriba de \$ 100,000.00.</i>		

Reducir el tiempo de producción.	100% del total del tiempo de producción.	Ganancia en costo 60% y en tiempo de un 50%.
<i>Con SAM se obtuvo una eficaz administración y se logró reducir hasta un 60% en costo y un 50% de tiempo del total invertido en el proyecto, con esto la empresa presenta mejoras y ganancias.</i>		
Tener un control de almacenamiento de información de acuerdo a la capacidad de servidores e Identificar por medio del nombre el status del archivo, ubicación y detalles del mismo.	\$ 5,000.00	\$ 0.00
<i>Con SAM se controla el almacenamiento en servidores ahorrando tiempo y dinero, facilitando además el manejo de acervo digital, de acuerdo al status en el que se encuentre el proyecto los empleados pueden eliminar el proyecto una vez concluido y a su vez liberar capacidad de almacenamiento en servidores. El identificar los archivos con facilidad nos permite reducir hasta en un 50% el tiempo utilizado en la ubicación de los proyectos y/o archivos facilitando la forma de trabajo.</i>		
Tener vínculo entre Ollin Studio y los clientes de forma que ellos puedan estar enterados del avance de sus proyectos en línea y tiempo real.	\$ 0.00	Mejora imagen de la empresa
<i>Al tener un vínculo Ollin Studio con sus clientes, es útil porque mejora la imagen de la empresa al tener contacto directo con sus clientes y que los clientes puedan revisar avances y/o status de proyectos en línea, esto puede ser una ventaja respecto a otras empresas, en la cual los clientes al contratar pueden presentar preferencia por Ollin Studio, también crea un vínculo directo entre la empresa y sus clientes.</i>		

Tabla 10.- Conclusiones (Comparativo antes y después del sistema SAM)

BIBLIOGRAFÍA

- [1] Análisis y diseño de sistemas
3era Edición
Kendall & Kendall
Pearson Education

- [2] Fundamentos de bases de datos
Silberschatz Korth Sudarshan
Editorial Mc Graw Hill
Cuarta edición

- [3] Sistemas de Bases de Datos
“Diseño, implementación y administración”
Peter Rob/Carlos Coronel
Thomson

- [4] Manuales de Maestría de Ciencias en Ingeniería de Sistemas
Dr. Leopoldo A. Galindo

- [5] Análisis y Diseño Orientado a Objetos
Aplicado a Sistemas de Información
Luis Fernando Castro Careaga
Derechos Reservados 1999 por Luis Fernando Castro Careaga.

- [6] Introducción a PHP 4 Noviembre 2000
Vicente Aguilar ¡vaguilar@linuxfreak.com

- [7] Manual de PHP
Stig Sther Bakken, Alexander Aulbach, Egon Schmid, Jim Winstead,
Lars Torben Wilson, Rasmus Lerdorf, Andrei Zmievski, Jouni Ahto.
Editado por Rafael Martínez 16-11-2001
Copyright 1997, 1998, 1999, 2000, 2001 por el Grupo de
documentación de PHP.

- [8] MySQL Álvaro Marín Illera split77@terra.es
Instalación, configuración y manejo de MySQL

- [9] MySQL Reference Manual for Version 4.0.3-beta
Copyright c 1997-2002 MySQL AB

ANEXO

Herramientas

El software utilizado para el desarrollo del sistema es:

Sistema Operativo Fedora Linux. Los sistemas operativos basados en UNIX poseen herramientas muy útiles para realizar sesiones remotas, como es el telnet y el SSH (Secure Shell), que nos permite acceder a nuestro sitio web desde cualquier sitio. Esto facilita enormemente la tarea, pues las modificaciones, ya sea al código, la base de datos u otros servicios, los podremos realizar estemos o no físicamente frente a la máquina.

- **PHP (ProcessadorHipertex)**

El lenguaje PHP es un lenguaje de programación de estilo clásico, es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones.

PHP se ejecuta en el servidor, eso nos permite acceder a los recursos que tenga el servidor como una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

- **MySQL**

Se eligió MySQL como manejador de bases de datos porque es parte de los recursos con los que cuenta la empresa además de ser compatible en entornos UNIX, ofreciendo una mejor integración con el lenguaje PHP.

- **APACHE SERVER**

Se pensó en Apache el servidor de páginas ya que es uno de los pocos servicios que no necesitan de mayores modificaciones después de su instalación. Facilita la publicación de las páginas además fue un requerimiento de la empresa.

INSTALACIÓN EN LINUX

- **PHP**

Para instalar PHP en nuestro servidor únicamente debemos instalar sus respectivos rpm's:

```
rpm -ivh PHP-4.0.6-7.i386.rpm
```

```
rpm -ivh PHP-mysql-4.0.6-7.i386.rpm
```

- **MySQL**

Para instalar MySQL en nuestro servidor únicamente debemos instalar sus respectivos rpm's:

```
rpm -ivh mysql-3.23.41-1.i386.rpm  
rpm -ivh mysql-devel-3.23.41-1.i386.rpm  
rpm -ivh mysql-navigator-1.3.8-62.i386.rpm  
rpm -ivh mysql-server-3.23.41-1.i386.rpm  
rpm -ivh mysqlclient9-3.23.22-6.i386.rpm
```

- **Apache**

Para instalar el servidor Apache en nuestro servidor únicamente debemos instalar un solo rpm:

```
rpm -ivh apache-1.3.20-16.i386.rpm
```

- **Pruebas PHP**

La empresa cuenta con servidores con Linux y una vez instalado Apache, PHP y MYSQL, lo primero que se hizo fue comprobar que PHP funciona bien.

Formato de Entrevista

Entrevista para la determinación de necesidades en la administración de archivos digitales en la empresa Ollin Studio

Datos del entrevistado

Nombre: _____

Cargo: _____

Cuestionario

- ¿Cuál es la función que desempeña?
- ¿Existe un sistema para el desarrollo de su función, cual es y cómo opera?
- ¿Qué áreas y cuantas personas están a su cargo?
- ¿Cuáles son los objetivos de su área?
- ¿Cuáles son los principales problemas (no solo actuales sino pasados) que usted ha enfrentado para cumplir con sus objetivos?
- ¿Qué es lo que le ha impedido a usted resolver estos problemas?
- ¿Qué necesita para resolverlos?
- ¿Considera usted que la solución de los problemas que menciono es determinante para el pleno logro de la empresa?
- ¿Cuáles cree usted que son las actividades críticas para un mejor desempeño de su área?
- ¿Cuáles son las principales funciones que su área realiza?
- ¿Cuáles de sus funciones considera usted que presenta debilidades en su ejecución y porque?
- ¿Sus funciones abarcan a los centros de producción?
- ¿En qué magnitud y en qué forma?
- ¿Qué cambios espera en su área con la implementación de SAM?
- ¿Cuál cree que sea el impacto en su área y en Ollin Studio?
- ¿Cuál es la información que usted necesita recibir de su área y como requiere que aparezca?
- ¿Qué herramientas considera usted que podrían optimizar su función?
- ¿Cuál es el proceso que se lleva actualmente?
- ¿Quiénes participan en todo el proceso?
- ¿Cuál es el flujo de trabajo que se lleva en la empresa?
- ¿Cuáles son los requerimientos que necesita para estructurar la información?
- ¿Cómo desea visualizar la información del sistema?
- ¿Cuál es la información que requiere que aparezca al realizar consultas?