

**UNIVERSIDAD TÉCNICA DE MACHALA**

**FACULTAD DE INGENIERÍA CIVIL**

**ESCUELA DE INFORMÁTICA**

**TRABAJO DE TITULACION PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS**

**TEMA:**

**“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONIEL CIA.LTDA”**

**AUTOR:**

SRTA. ANDREA ANABELL LOAIZA GONZAGA

**MACHALA - EL ORO - ECUADOR**

**2014**

**TÍTULO:**

**“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONIEL CIA.LTDA”**

**FORMULACIÓN DEL PROBLEMA.**

La compañía de Construcciones e Instalaciones Eléctricas CONIEL CIA.LTDA lleva la gestión de las actividades de campo diarias realizadas por los grupos de trabajo de manera manual, es decir tiene una persona encargada de llenar a mano una ficha de servicio, donde se lleva el control del trabajo que realizan.

**PROBLEMA CENTRAL**

La utilización inadecuada de las nuevas tecnologías de información, genera cierto retraso en los procesos productivos dentro de las empresas u organizaciones.

Actualmente la compañía no cuenta con aplicaciones que automaticen los procesos relacionados con la gestión de la información o que ayuden a agilizar los mismos.

**PROBLEMAS PARTICULARES**

* La compañía no cuenta con aplicaciones o sistemas que ayuden a agilizar los procesos.
* La empresa maneja procesos de forma manual.

**JUSTIFICACIÓN**

A través del tiempo ha existido competencia y en ocasiones cierto grado de rivalidad entre empresas ya que se tienden a captar el mayor número de clientes posible para incrementar la demanda de sus productos o servicios, por lo que se ven obligadas a trazar estrategias de marketing y publicidad cada vez mejores y enfocadas a la mayor cantidad de medios.

Basándonos en este enfoque determinamos la importancia de situar a Coniel Cia. Ltda. a disposición de sus clientes de tal manera que logremos interacción con los mismos por un medio digital, es decir a través del desarrollo de un portal web.

**OBJETIVOS**

**OBJETIVO GENERAL**

* Desarrollar e Implementar un portal web dinámico para la empresa construcciones e instalaciones eléctricas CONIEL CIA.LTDA.

**OBJETIVOS ESPECIFICOS**

* Recopilar y Analizar la información con el fin de esquematizar el portal web basado en los requerimientos del usuario.
* Diseñar un portal web, que sirva como herramienta de apoyo para el crecimiento comercial de la empresa.
* Usar las TIC’s aprovechando al máximo los recursos con los que cuenta la institución, para de esta manera crear un vínculo de comunicación con los usuarios a través de la publicación de información del personal, actividades y organización de los servicios con los que cuenta la compañía.
* Ofrecer servicios en línea, de tal manera que se puedan atender necesidades generales con respecto energía eléctrica e informática y telecomunicaciones.
* Facilitar la retroalimentación a la compañía por medio de un módulo de reclamos y sugerencias.

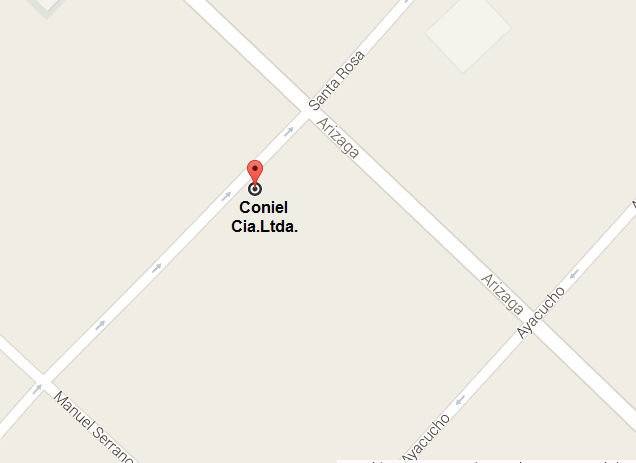
**MARCO REFERENCIAL**

**MARCO TEÓRICO**

**MARCO HISTÓRICO**

**Reseña Histórica de la CONIEL CIA.LTDA**

**UBICACIÓN**

****

**Dirección:** Machala, Santa Rosa e/ Arízaga y Gral. Manuel Serrano  
**Teléfono:** 2938-581

**ORGANIGRAMA ESTRUCTURAL**

**MARCO CONCEPTUAL**

**Historia de la Web**

La World Wide Web es la culminación del hipertexto y, como tal, sus orígenes se remontan a la propia historia del hipertexto. Quince años después de Tim Berners-Lee inventor de la WWW que Ted Nelson, en 1965 imaginara las máquinas literarias que permitirían a usuarios de todo el mundo publicar información en hipertexto, un joven ingeniero. Tim Berners-Lee, en sus ratos libres, investigaba un sistema de difusión descentralizada de información basado en el hipertexto y en el que cooperaban una red de centros de investigación.

Durante 1989 se realizaron varios experimentos para crear un sistema de comunicación entre los científicos de física nuclear de todo el mundo en el Centre Européen de Recherche Nucléaire (CERN) en Ginebra, donde se probaron varias técnicas sobre redes para conectarse a Internet, llegándose a estandarizar los protocolos de conexión TCP/IP. En marzo de ese mismo año, Tim Berners-Lee, investigador especializado en telecomunicaciones y colaborador del Laboratorio Europeo de Física de Partículas (CERN), propone al CERN utilizar un sistema de comunicación basado en el hipertexto para lograr ese flujo de información entre los científicos.

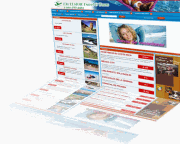
En noviembre de 1990 Tim Berners-Lee puso en marcha el sistema de hipertexto llamado Enquire que permitía almacenar piezas de información y conectarlas, y que se ejecutaba en un entorno multiusuario que permitía acceder a la información a varias personas a la vez. En marzo de 1991, Berners-Lee escribió el primer programa visualizador para un servidor y cliente, que se convirtió en el origen de la World Wide Web. Este sistema duró hasta 1993, cuando la International Standards Organization (ISO) estandarizó el lenguaje Hypertext Markup Language (HTML). Hasta entonces, los documentos se editaban mediante TeX y PostScript, pero estos lenguajes eran demasiado complicados teniendo en cuenta que debían ser leídos por todo tipo de ordenadores, desde simples terminales hasta estaciones de trabajo gráficas X-Windows. No existían los buscadores, ni se podían integrar imágenes en la pantalla puesto que no habían nacido las interfaces gráficas.

La idea que movió a Tim Berners-Lee, fue desarrollar un método eficiente y rápido para intercambiar datos entre la comunidad científica. Para ello, combinó dos tecnologías ya existentes (el hipertexto y el protocolo de comunicaciones de Internet), creando un nuevo modelo de acceso a la información intuitivo e igualitario: la Web que hace posible que cualquiera pueda utilizar Internet.

Actualmente, Tim Berners-Lee dirige el World Wide Web Consortium, la organización que coordina estándares y añade nuevas funcionalidades y desarrollos a la Web.

Desde mediados de 1991 hasta hoy, la Web no ha cesado de desarrollarse y evolucionar con nuevas capacidades que completan y mejoran este sistema digital, hipertextual y en red. Un ejemplo muy ilustrativo ha sido el crecimiento exponencial de la WWW. (Lapuente, 2013)

**¿QUÉ ES UN PORTAL WEB?**

****

 Un Portal es el punto de entrada a Internet donde se organizan y concentran los contenidos del propietario del mismo.

El objetivo principal del Portal es ayudar a los usuarios a encontrar lo que necesitan sin salir del mismo fidelizándoles e incentivándoles a utilizarlo de forma continuada.

Los 3 pilares fundamentales de un Portal para atraer la atención del usuario y son:

1. Información (Buscadores, directorios, noticias, catálogos y servicios)

1. Participación (E-mail, foros, chat)

1. Comodidad (brindar la mayor cantidad de información en un solo espacio y tenerlo todo a mano)

 De esa forma el Portal gana tráfico constante y alto y se autofinancia con publicidad.

**TIPOS DE PORTALES WEB**

Existen tres modalidades de portales:

1. **Portales horizontales:** También llamados portales masivos, se dirigen a una audiencia amplia, tratando de llegar a toda la gente con muchas cosas. Dentro de esta categoría están los mencionados en el punto anterior, y además podemos destacar estos otros: Terra, AOL, AltaVista, Lycos.

2. **Portales verticales:** Ofrecen contenido dentro de un tema específico como puede ser un portal de música (portalmusica.com), empleo (infoempleo, trabajar.com), de instituciones (Ayto. Rioseco, Diputación de Valladolid, Junta Castilla y León, Portal del menor,..), un portal de finanzas personales, arte o de deportes.

3. **Portales diagonales:** Son portales que utilizan como eje central contenidos propios de redes sociales como Flickr o YouTube, las cuales se pueden personalizar con contenidos y/o utilidades dirigidas a un público muy concreto. (Jones, 2008)

**PYTHON**

Python es un lenguaje de programación interpretado e interactivo de propósito general. Es, hasta cierto punto, comparable con otros lenguajes de programación de dominio específico que podemos encontrar dentro del ámbito de la Ingeniería como Matlab, Octave, R, SPSS o IDL.

Se trata también de un lenguaje de programación relativamente moderno y en constante, aunque moderada, renovación. Fue creado por Guido van Rossum en el año 1991 tomando prestadas muchas de las buenas ideas presentes en los lenguajes de programación que conocía. A diferencia de lo que viene siendo habitual, en vez de reinventar cada idea simplemente las incorporó de manera que tuvieran sentido.

Python es un lenguaje que todo el mundo debería conocer. Su sintaxis simple, clara y sencilla; el tipado dinámico, el gestor de memoria, la gran cantidad de librerías disponibles y la potencia del lenguaje, entre otros, hacen que desarrollar una aplicación en Python sea sencillo, muy rápido y, lo que es más importante, divertido.

**Características.-**

* Simple
* Sencillo de aprender
* Libre y fuente abierta
* Indentado
* Lenguaje de alto nivel
* Portable
* Interpretado
* Orientado a Objetos
* Ampliable (Permite combinar fragmentos con otros lenguajes de programación).
* Incrustable (Permite insertar código en otros lenguajes para dar facilidades de scripting).
* Librerias extendidas

**Ventajas:**

• Rápido de desarrollar.

• Sencillez y velocidad.

• Sus bibliotecas hacen gran parte del trabajo.

• Soporta varias bases de datos.

**Desventajas:**

• Los programas interpretados son más lentos que los compilados. (Duque, 2011)

**DJANGO**

Es un framework de desarrollo Web que ahorra tiempo y hace que el desarrollo Web sea divertido. Utilizando Django puedes crear y mantener aplicaciones Web de alta calidad con un mínimo esfuerzo.

Django nació naturalmente de aplicaciones de la vida real escritas por un equipo de desarrolladores Web en Lawrence, Kansas. Nació en el otoño boreal de 2003, cuando los programadores Web del diario Lawrence Journal-World, Adrian Holovaty y Simon Willison, comenzaron a usar Python para crear sus aplicaciones. El equipo de The World Online, responsable de la producción y mantenimiento de varios sitios locales de noticias, prosperaban en un entorno de desarrollo dictado por las fechas límite del periodismo. Para los sitios – incluidos LJWorld.com, Lawrence.com y KUsports.com – los periodistas (y los directivos) exigían que se agregaran nuevas características y que aplicaciones enteras se crearan a una velocidad vertiginosa, a menudo con sólo días u horas de preaviso. Es así que Adrian y Simon desarrollaron por necesidad un framework de desarrollo Web que les ahorrara tiempo – era la única forma en que podían crear aplicaciones mantenibles en tan poco tiempo – .

En el verano boreal de 2005, luego de haber desarrollado este framework hasta el punto en que estaba haciendo funcionar la mayoría de los sitios World Online, el equipo de World Online, que ahora incluía a Jacob Kaplan-Moss, decidió liberar el framework como software de código abierto. Lo liberaron en julio de 2005 y lo llamaron Django, por el guitarrista de jazz Django Reinhardt.

A pesar de que Django ahora es un proyecto de código abierto con colaboradores por todo el mundo, los desarrolladores originales de World Online todavía aportan una guía centralizada para el crecimiento del framework, y World Online colabora con otros aspectos importantes tales como tiempo de trabajo, materiales de marketing, y hosting/ancho de banda para el Web site del framework.

Esta historia es relevante porque ayuda a explicar dos cuestiones clave. La primera es el “punto dulce” de Django. Debido a que Django nació en un entorno de noticias, ofrece varias características que son particularmente apropiadas para sitios de “contenido” – sitios como eBay, craigslist.org y washingtonpost.com que ofrecen información basada en bases de datos –. (De todas formas, no dejes que eso te quite las ganas – a pesar de que Django es particularmente bueno para desarrollar esa clase de sitios, eso no significa que no sea una herramienta efectiva para crear cualquier tipo de sitio Web dinámico –. Existe una diferencia entre ser particularmente efectivo para algo y no ser efectivo para otras cosas).

La segunda cuestión a resaltar es cómo los orígenes de Django le han dado forma a la cultura de su comunidad de código abierto. Debido a que Django fue extraído de código de la vida real, en lugar de ser un ejercicio académico o un producto comercial, está especialmente enfocado en resolver problemas de desarrollo Web con los que los desarrolladores de Django se han encontrado – y con los que continúan encontrándose –. Como resultado de eso, Django es activamente mejorado casi diariamente. Los desarrolladores del framework tienen un alto grado de interés en asegurarse de que Django les ahorre tiempo a los desarrolladores, produzca aplicaciones que son fáciles de mantener y rindan bajo mucha carga. (Kaplan-Moss., 2006)

**POSTGRES**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

#### **Características Generales**

* Es una base de datos 100% [ACID](http://es.wikipedia.org/wiki/ACID)
* Integridad referencial
* Tablespaces
* Nested transactions (savepoints)
* Replicación asincrónica/sincrónica / Streaming replication - Hot Standby
* Two-phase commit
* PITR - point in time recovery
* Copias de seguridad en caliente (Online/hot backups)
* Unicode
* Juegos de caracteres internacionales
* Regionalización por columna
* Multi-Version Concurrency Control (MVCC)
* Multiples métodos de autentificación
* Acceso encriptado via SSL
* Actualización in-situ integrada (pg\_upgrade)
* SE-postgres
* Completa documentación
* Licencia BSD
* Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit. (Martínez, 2009)

**DISEÑO METODOLÓGICO PRELIMINAR.**

**METODOLOGIA UWE**

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas.

**Las fases o etapas a utilizar son:**

**1) Captura, análisis y especificación de requisitos:** En simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

     Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipado de la interfaz de usuario.

**2) Diseño del sistema:** Se basa en la especificación de requisitos producido por el análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

**3) Codificación del software:** Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior.

**4) Pruebas:** Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.

**5) La Instalación o Fase de Implementación:** es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados, y, eventualmente, configurados; todo ello con el propósito de ser ya utilizados por el usuario final.

     Esto  incluye la implementación de la arquitectura, de la estructura del hiperespacio, del modelo de usuario, de la interfaz de usuario, de los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones.

**6) El Mantenimiento:** es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control.

**NOMBRES DE LAS PERSONAS QUE PARTICIPAN EN EL PROCESO.**

Las personas que participan en este proceso de desarrollo, implementación y manejo del portal web son:

* **CONIEL CIA.LTDA**
* **Sr. Julio Loaiza** Gerente General
* **Analista, Diseñadora, Desarrolladora del portal web:** Srta. Andrea Anabell Loaiza Gonzaga.

**Estudiante de la Escuela de Informática**

**RECURSOS DISPONIBLES (MATERIALES INSTITUCIONALES Y FINANCIEROS).**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Recursos Humanos** | | | | | |
| **Nro.** | **Personal** | **Tiempo** | **Costo H/T** | | **Total** |
| 1 | Tesista | 4 meses | $200c/u | | $800.00 |
| **Subtotal de A** | | | | | **$800.00** |
| 1. **Recursos Tecnológicos** | | | | | |
| **B.1. Hardware** | | | | | |
| **Cant.** | **Descripción** | | **Costo unitario** | **Total** | |
| 1 | Computadora | | $600.00 | $600.00 | |
| **Subtotal de B1** | | | | **$600.00** | |
| **B.2. Software** | | | | | |
| **Cant.** | **Descripción** | | **Costo unitario** | **Total** | |
| 1 | Microsoft Word | | $0.00 | $0.00 | |
| 1 | Django | | $0.00 | $0.00 | |
| 1 | Python | | $0.00 | $0.00 | |
| 1 | Postgres | | $0.00 | $0.00 | |
| 1 | PyCharm | | $0.00 | $0.00 | |
| 1 | Google Chrome | | $0.00 | $0.00 | |
| 1 | MagicDraw | | $749.00 | $749.00 | |
| **Subtotal de B2** | | | | **$749.00** | |
| **Subtotal de B1+B2** | | | | **$1349.00** | |
| 1. **Otros** | | | | | |
| **Cant.** | **Descripción** | | **Costo unitario** | **Total** | |
| 1 | Materiales de oficina | | $90.00 | $90.00 | |
| 1 | Internet | | $100.00 | $100.00 | |
| 1 | Servicio de Dominio | | $200.00 | $200.00 | |
| 1 | Servicios de alojamiento en host | | $65.00 | $65.00 | |
| 1 | Movilización | | $85.00 | $85.00 | |
| 1 | Capacitación | | $300.00 | $300.00 | |
| **Subtotal de C** | | | | **$840,00** | |
| 1. **Imprevistos 5% de A+B+C** | | | | **$149,45** | |
| **COSTO TOTAL DEL PROYECTO** | | | | **$3138,45** | |

**BIBLIOGRAFÍA**

Duque, R. G. (2011). *Python para todos.* Madrid: Autoedición.

Jones, P. (19 de 12 de 2008). *http://utilidadytiposdeportales.wordpress.com/*. Obtenido de http://utilidadytiposdeportales.wordpress.com/

Kaplan-Moss., A. H. (08 de 10 de 2006). *http://django-book.mkaufmann.com.ar/*. Obtenido de http://django-book.mkaufmann.com.ar/chapter01.html#la-historia-de-django

Lapuente, D. M. (08 de 12 de 2013). *http://www.hipertexto.info/*. Obtenido de http://www.hipertexto.info/documentos/h\_www.htm