



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Trabajo de grado previa la obtención del título de
INGENIERO EN SISTEMAS COMPUTACIONALES

TEMA:

“Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.”

AUTOR:

Sergio Dionisio Benítez Tixicuro

DIRECTOR:

Ing. Iván García, Mgs.

Ibarra – Ecuador

2011

UNIVERSIDAD TECNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

1. IDENTIFICACIÓN DE LA OBRA

La Universidad Técnica del Norte dentro del proyecto Repositorio Digital Institucional, determina la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la Universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	100242285-3		
APELLIDOS Y NOMBRES:	BENÍTEZ TIXICURO SERGIO DIONISIO		
DIRECCIÓN:	FLORES 12-53 Y AV. JAIME RIVADENEIRA		
EMAIL:	cci_ibarra@hotmail.com		
TELÉFONO FIJO:	2612311	TELÉFONO MOVIL:	099957488

DATOS DE LA OBRA	
TÍTULO:	"Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico."
AUTOR:	BENÍTEZ TIXICURO SERGIO DIONISIO
FECHA:	22-NOVIEMBRE-2011
SÓLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA:	INGENIERO EN SISTEMAS COMPUTACIONALES
ASESOR / DIRECTOR:	ING. IVÁN GARCÍA, MGS.

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, BENÍTEZ TIXICURO SERGIO DIONISIO, con cédula de identidad Nro. 100242285-3, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago la entrega del ejemplar respectivo en formato digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión; en concordancia con la Ley de Educación Superior Artículo 143.

Yo, BENÍTEZ TIXICURO SERGIO DIONISIO, con cédula de identidad Nro. 100242285-3, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor de la obra de trabajo de grado denominado: "Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.", que ha sido desarrollado para optar por el título de: Ingeniero en Sistemas Computacionales, en la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Técnica del Norte.

.....

Sergio Dionisio Benítez Tixicuro

CI:100242285-3

Ibarra, a los 24 días del mes de noviembre del 2011

CERTIFICACIÓN

El Señor egresado Sergio Dionisio Benítez Tixicuro, ha trabajado en el desarrollo del proyecto de tesis **“Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico”**, previo a la obtención del Título de Ingeniero en Sistemas Computacionales, realizando con interés profesional y responsabilidad, lo cual certifico en honor a la verdad.

Ing. Iván García, Mgs.

DIRECTOR DE TESIS

DEDICATORIA

A mis hijos, Sergio Jr., e Isaac David, por ser la base fundamental de inspiración para la culminación de esta etapa en mi vida.

A mi madre María del Carmen, por darme el ejemplo de que nada es imposible si se lo deseas y se tiene fe en Dios, por ser mi guía en el camino del bien, saber enseñarme la importancia de culminar todos los proyectos trazados en la vida, éste es el fruto de su esfuerzo y dedicación.

A mi esposa amada Verito, por los momentos difíciles que tuvo que sobrellevar para la culminación de este proyecto, infundiéndome en mi serenidad y fuerza en los momentos de desaliento.

A mis profesores, quienes me enseñaron que no todo es fácil, pero tampoco imposible, sólo hay que dedicarse e investigar; y ser honrado en cada acto de la vida para culminar los proyectos.

Sergio Benítez Tixicuro

AGRADECIMIENTO

Al Ing. Iván García, Mgs., por darme una guía concreta, por ser el facilitador de sus conocimientos que aunados a mi trabajo han llevado a término los objetivos trazados en la elaboración del proyecto.

A mi madre, que me infundió ética, amor y perseverancia en los estudios, los cuales me llevaron a ser mejor persona y mejor profesional.

A mis profesores de la magna Universidad Técnica del Norte, de la cual me llevo los conocimientos y vivencias que servirán de guía durante mi vida profesional.

A los funcionarios de la Dirección Provincial Ambiental de Imbabura por haberme brindado las facilidades necesarias para la culminación del presente trabajo...

Sergio Benítez Tixicuro

TABLA DE CONTENIDO

DEDICATORIA.....	I
AGRADECIMIENTO.....	II
RESUMEN	IX
SUMMARY	X
INTRODUCCIÓN	- 11 -
Descripción:	- 12 -
Objetivos	- 14 -
Objetivo General:.....	- 14 -
Objetivos Específicos:.....	- 14 -
Alcance	- 15 -
Herramientas de desarrollo	- 16 -
Impactos	- 16 -
Capítulo I	- 18 -
1.1. Visión.	- 19 -
1.1.1. Propósito.	- 19 -
1.1.2. Alcance.	- 20 -
1.1.3. Posicionamiento.	- 20 -
1.1.4. Descripción de participantes en el proyecto y usuarios.	- 22 -
1.1.5. Perfil de los participantes y usuarios.	- 23 -
1.1.6. Descripción global de producto.....	- 25 -
1.1.7. Costo y precios.....	- 27 -
1.1.8. Característica global del producto.....	- 28 -
1.1.9. Seguridad.....	- 28 -
1.1.10. Otros requisitos del producto.....	- 28 -
1.2. Plan de desarrollo de software.....	- 29 -
1.2.1. Propósito.	- 29 -
1.2.2. Alcance.	- 29 -
1.2.3. Vista general del proyecto.....	- 30 -
1.2.4. Puntos clave.....	- 31 -
1.2.5. Entregables del proyecto.	- 32 -
1.2.6. Organización del proyecto.....	- 34 -
1.2.7. Plan del Proyecto.	- 35 -
1.3. Captura de requisitos.	- 41 -
1.3.1. Análisis del sistema actual.....	- 41 -
1.3.1.1. Casos de negocios.	- 43 -
1.3.1.2. Modelo de casos de uso.....	- 45 -
1.3.2. Requisitos nuevos de los Usuarios.....	- 50 -
1.3.2.1. Modelo de casos de uso.	- 51 -
1.3.3. Descripción del Sistema Propuesto.	- 53 -
1.3.3.1. Arquitectura del Software.	- 53 -

1.4. Especificación del Sistema.	- 54 -
1.4.1. Descripción del Sistema.	- 54 -
1.4.1.1. Especificación de Casos de Uso.	- 54 -
1.4.2. Requisitos de datos.	- 60 -
1.4.2.1. Modelo Entidad-Relación.	- 60 -
1.4.3. Requisitos de Hardware y Software.	- 61 -
1.4.4. Requisitos de Telecomunicaciones.	- 63 -
Capítulo II	- 64 -
2.1. Descripción detallada del sistema.	- 65 -
2.1.1. Programas, módulos reutilizables y Objetos.	- 65 -
2.1.1.1. Prototipo de Interfaz de usuario.	- 67 -
2.1.2. Modelo Relacional.	- 77 -
2.1.2.1. Ficheros y Base de datos.	- 77 -
2.1.2.2. Diccionario de datos.	- 78 -
2.1.2.3. Carga de sistema y tiempos de respuesta.	- 78 -
2.1.2.4. Interfaces, tanto humanas como de máquinas.	- 79 -
2.2. Descripción de los controles del sistema propuesto.	- 79 -
2.2.1. Modelo de Implementación.	- 79 -
2.2.2. Estándares de Programación.	- 83 -
2.2.3. Técnicas de Implementación.	- 84 -
2.2.4. Casos de prueba.	- 85 -
Capítulo III	- 86 -
3.1. Diseño final del sistema.	- 87 -
3.2. Diagramas definitivos del sistema.	- 89 -
3.3. Descripción detallada de la lógica del sistema.	- 97 -
3.4. Descripción de entradas y salidas.	- 99 -
3.4.1. Ficheros.	- 100 -
3.4.2. Pantallas.	- 104 -
3.4.3. Listados.	- 109 -
3.5. Cadenas de Ejecución.	- 109 -
3.5.1. Scripts.	- 109 -
3.6. Resultados de Pruebas de cada módulo.	- 112 -
3.7. Resultados de pruebas del Sistema.	- 112 -
3.8. Resultados de pruebas de Integración.	- 112 -
3.9. Manual de operadores del sistema.	- 113 -
3.10. Manual de usuario del sistema.	- 113 -
Capítulo IV	- 114 -
4.1. Pruebas.	- 115 -
4.1.1. Plan de pruebas del sistema.	- 115 -
4.1.2. Informe de los resultados de las pruebas.	- 116 -

4.1.3.	Acciones a tomar para corregir las desviaciones.....	- 117 -
4.2.	Instalación.....	- 118 -
4.2.1.	Planes de contingencia.....	- 118 -
4.2.2.	Plan de revisión post – instalación.....	- 118 -
4.2.3.	Informe de la instalación.....	- 119 -
4.2.4.	Carta de aceptación del sistema.....	- 119 -
4.3.	Mantenimiento.	- 119 -
4.3.1.	Listado de fallos detectados.	- 119 -
4.3.2.	Listado de mejoras solicitadas por los usuarios.	- 120 -
4.3.4.	Actas de trabajo y aceptación.	- 121 -
CONCLUSIONES Y RECOMENDACIONES		- 122 -
Conclusiones:		- 123 -
Recomendaciones:		- 124 -
Glosario de Términos:.....		- 125 -
Referencias:		- 127 -
Libros:.....		- 128 -
ANEXOS		- 129 -
ANEXO NUM 1: Diccionario de Datos		- 130 -
ANEXO NUM 2: Lista de Riesgos		- 137 -
ANEXO NUM 3: Ficheros Creados para realización del sistema.		- 138 -
ANEXO NUM 4: Listado de Programas.		- 142 -
ANEXO NUM 5: Manual del Operador del Sistema Web - Control Asistencia-MAE.-		169 -
ANEXO NUM 6: Manual de Usuario del Sistema Web - Control Asistencia-MAE ...-		191 -
ANEXO NUM 7: Construcción del Cable.....		- 197 -
ANEXO NUM 8: Actas de Trabajo		- 199 -
ANEXO NUM 9: Documentación de la Etapa de Pruebas.		- 201 -
ANEXO NUM 10: BENCHMARK (Comparación Sistemas similares) y Cálculo de costo (Cocomo).....		219

ÍNDICE DE FIGURAS

INTRODUCCIÓN

Figura 1: Módulos del proyecto	15
--------------------------------------	----

CAPÍTULO I

Figura 1.1. Descripción del Proceso Unificado.....	37
Figura 1.2. Arquitectura Actual.	41
Figura 1.3. Diagrama casos de uso global del negocio.....	43
Figura 1.4. Diagrama Casos de Uso del Sistema Actual	45
Figura 1.5. Pantalla parámetros y asignación del sistema actual.	46
Figura 1.6. Pantalla Carga datos del sistema actual.....	47
Figura 1.7. Pantalla cálculo de Resultados del sistema actual.....	48
Figura 1.8. Pantalla Reporte de Resultados del sistema actual generado en cristal report.....	49
Figura 1.9. Casos de uso de requerimientos nuevos	51
Figura 1.10. Arquitectura del Sistema Propuesto	53
Figura 1.11. Modelo de casos de uso Seguridad y Accesos	54
Figura 1.12. Diagrama de Casos de Uso de Sistema Propuesto global.....	55
Figura 1.13. Diagrama de Casos de Uso Administrar parámetros.	56
Figura 1.14. Diagrama de Casos de Uso Crear Empleados y Asignar PMV.	57
Figura 1.15. Diagrama de Casos de Uso Cargar y calcular los datos del reloj.....	58
Figura 1.16. Diagrama de Casos de Uso Consultar y reportar E/S Emp.	59
Figura 1.17. Diagrama Entidad Relación de la base de datos requerida.	60
Figura 1.18. Imagen Reloj Biométrico de la Institución	61
Figura 1.19. Cable y conectores para construcción de cable para Comunicación Reloj Biométrico a Computador.....	61
Figura 1.20. Infraestructura de telecomunicación.....	63

CAPÍTULO II

Figura 2.1. Diagrama de clases del sistema.	66
Figura 2.2. Prototipo de Formulario de Seguridad.	67
Figura 2.3. Prototipo de Formulario de Usuario.	68
Figura 2.4. Prototipo de Formulario de parámetros (Turno).....	69
Figura 2.5. Prototipo de Formulario de parámetros (Departamentos).	70
Figura 2.6. Prototipo de Formulario de parámetros (Secciones).	70
Figura 2.7. Prototipo de Formulario de parámetros (Cargos).	71
Figura 2.8. Prototipo de Formulario de Asignación (Empleados).	72
Figura 2.9. Prototipo de Formulario de Asignación (Permisos).	73
Figura 2.10. Prototipo de Formulario de Asignación (Movilizaciones).....	74
Figura 2.11. Prototipo de Formulario de Asignación (Vacaciones).....	74
Figura 2.12. Prototipo de Formulario de Carga de datos.	75
Figura 2.13. Prototipo de Formulario de Consulta.	76
Figura 2.14. Diagrama Entidad Relación.	77
Figura 2.15. Diagrama de secuencia para Consultar Entradas y Salidas de empleado.	80
Figura 2.16. Diagrama de secuencia para la administración del sistema.	81

Figura 2.17. Diagrama de componentes del sistema.....	82
Figura 2.18. Diagrama de despliegue del sistema.	82
Figura 2.19. Diagrama de secuencia para la administración del sistema (fuente).....	85
Figura 2.20. Gráfico de casos de prueba propuesto para el sistema.....	85

CAPÍTULO III

Figura 3.1. Pantalla del diseño final del sistema etapa Seguridad.	87
Figura 3.2. Pantalla del diseño final del sistema etapa Administración.	87
Figura 3.3. Pantalla del diseño final del sistema etapa Consulta.....	88
Figura 3.4. Diagrama de actividades de parámetros modificación.	89
Figura 3.5. Diagrama de actividades de parámetros modificación.	90
Figura 3.6. Diagrama de actividades de parámetros nuevo.	90
Figura 3.7. Diagrama de actividades de parámetros eliminar.	91
Figura 3.8. Diagrama de actividades de asignación modificar.....	92
Figura 3.9. Diagrama de actividades de asignación nuevo.	93
Figura 3.10. Diagrama de actividades de asignación eliminar.....	93
Figura 3.11. Diagrama de actividades Cargar datos al sistema.	94
Figura 3.12. Diagrama de actividades Calcular datos al sistema.	95
Figura 3.13. Diagrama de actividades Reporte de datos.	96
Figura 3.14. Descripción E/S Pantalla Ingreso.....	104
Figura 3.15. Descripción E/S Pantalla Parámetros Departamento Nuevo.....	104
Figura 3.16. Descripción E/S Pantalla Parámetros Departamento Listado.....	104
Figura 3.17. Descripción E/S Pantalla Asignación Empleado Nuevo.	105
Figura 3.18. Descripción E/S Pantalla Asignación Empleado Listado.	105
Figura 3.19. Descripción E/S Pantalla Cargar Datos al sistema.....	106
Figura 3.20. Descripción E/S Pantalla Calcular Atrasos, Horas extras, Horas Trabajadas.	106
Figura 3.21. Descripción E/S Pantalla Consulta y reporte de E/S empleados.....	107
Figura 3.22. Descripción E/S Pantalla Resultado consulta y reporte de E/S empleados.....	107
Figura 3.23. Descripción E/S Pantalla Resultado consulta y reporte de E/S empleados en pdf.....	108
Figura 3.24. Descripción de Pantalla de Permisos, Movilizaciones y vacaciones de empleados en pdf.....	108

ÍNDICE DE TABLAS

INTRODUCCIÓN

Tabla 1. Herramientas de desarrollo a usar y justificación.....	16
---	----

CAPÍTULO I

Visión

Tabla 1.1. Definición del Problema	21
Tabla 1.2. Posicionamiento del Producto	21
Tabla 1.3. Resumen de participantes.....	22
Tabla 1. 4. Resumen de usuarios.	22
Tabla 1. 5. Perfil director de proyecto.	23
Tabla 1.6. Perfil Administrador.	23
Tabla 1.7. Perfil usuario.....	24
Tabla 1.8. Necesidades de los interesados y usuarios.	24
Tabla 1.9. Resumen de características.....	26
Tabla 1.10. Costo y Precios	27

Plan desarrollo de software

Tabla 1.11. Roles y Responsabilidades.....	35
Tabla 1.12. Plan de Fases	35
Tabla 1.13. Plan de Fases-Hitos.....	36
Tabla 1.14. Roles y Responsabilidades – Fase de inicio.....	38
Tabla 1.15. Roles y Responsabilidades – Fase de elaboración	39

CAPÍTULO II

Tabla 2.1. Herramientas a utilizar en el desarrollo	65
Tabla 2.2. Carga y tiempos de respuesta del sistema.....	78

CAPÍTULO III

Tabla 3.1. Representación del mapa del sitio.	88
--	----

CAPÍTULO IV

Tabla 4.1. Plan de pruebas del sistema.....	115
Tabla 4.2. Informe de resultados de las pruebas.....	116
Tabla 4.3. Acciones a tomar para corregir desviaciones.....	117
Tabla 4.4. Listado de fallos detectados.....	119
Tabla 4.5. Listado de mejoras solicitadas por los usuarios.	120
Tabla 4.6. Traza detallada de los cambios realizados.	121

RESUMEN

La confianza que se obtiene en los resultados que generan los relojes biométricos como forma de control y optimización de la asistencia del personal, han despertado el interés de combinar los relojes biométricos y los sistemas web, tomando en cuenta que los sistemas existen pero no a medida, además se han desarrollado con herramientas propietarias y es hora de migrarlas a herramientas libres y metodologías nuevas que son necesarias para este tipo de proyectos; además de brindar una gran ayuda por medio de la visualización externa de la información a través de Internet, pudiendo interactuar empleados y autoridades en forma transparente.

Sergio Benítez.

SUMMARY

The confidence that is obtained in the results that generate the biometric clocks as form of control and optimization of the assistance of the personnel, they have woken the interest up of combining the biometric clocks and the web systems, bearing in mind that the systems exist but not made-to-measure, in addition they have developed with proprietary tools and it is time to migrate to free tools and new methodologies that are necessary for this type of projects; beside offering a great help by means of the external visualization of the information across Internet, being able to interact employees and authorities in transparent form.

Sergio Benítez

INTRODUCCIÓN

Antecedentes, Objetivos, Alcance y Herramientas.

Breve descripción.

Descripción:

La Dirección del Ambiente de Imbabura, es una Institución dedicada a la preservación de la naturaleza, la cual mediante comunicado emitido desde el Ministerio del Ambiente a sus direcciones provinciales, ordena un control de asistencia a todos sus empleados a nivel nacional con medios biométricos.

Anteriormente, el control de la asistencia se lo hacía manualmente por medio de firmas en documentos impresos, lo cual generaba muchas falencias al momento de pagar y realizar un control eficiente del empleado.

Es así, que hace 12 meses la Dirección del Ambiente de Imbabura, hizo la adquisición de un reloj biométrico de huella digital y el Sistema para el control de la asistencia de su personal.

Antecedentes:

El Sistema Implementado fue desarrollado en la ciudad de Quito, al momento existen **falencias** en el sistema debido a los ajustes que se realiza en cada empresa, por lo cual las versiones van variando, pero debido a la necesidad de vender el aparato biométrico, se entrega una versión para generar reportes que está en desarrollo y no se ha depurado bien todavía.

Por lo cual surgen varios problemas como son:

Hardware: Al momento de descargar los datos al sistema, se lo realiza con la salida USB, esto hace que no siempre descargue bien los datos.

Sistema: Al momento de cargar los datos al sistema se debe convertir el archivo generado por el reloj, esto genera molestia a la persona encargada.

Hay muchas opciones para ingresar parámetros de configuración en el sistema que no están muy claras en su función y otras que no son necesarias para la institución.

Para ingresar los permisos, no se tiene un formato que utilice la Dirección del Ambiente, esto hace que como usuarios se ajusten a la plantilla que viene en el programa, pero que no están de acuerdo a la necesidad de la institución.

Otro problema existente, es que, al generar los reportes no todos funcionan, algunos se generan con errores que pueden deberse a la configuración, la falta de parámetros o por alguna línea de código en el sistema.

Además la contadora de la institución manifiesta: que necesita que el sistema genere un reporte específico de los permisos, atrasos y la hora de entrada y salida del almuerzo para el cálculo de los pagos de cada empleado.

El Director provincial manifiesta: la falta de un sitio web, para poder observar y controlar a sus empleados mediante reportes según le convenga desde cualquier parte en donde se encuentre, y que, el inconveniente es que siempre es necesario que la Secretaria imprima dichos reportes y que no se optimiza la entrega de información ni la observación detallada de cada uno de sus empleados.

Los empleados manifiestan: que necesitan informarse sobre sus reportes de asistencia para poder observar los atrasos, permisos, etc., y así poder realizar algún reclamo, si así lo creyere conveniente, pero, lo más importante es que se lo realice a través del internet para consultar su asistencia desde cualquier lugar en donde se encuentren.

Además de esto, los reportes no se ajustan a los requerimientos del Ministerio, ya que, como es obvio son reportes que están predeterminados por el Sistema actual y que no pueden ser cambiados según la necesidad de la Institución.

La realización de este proyecto se enfoca en generar reportes que solucionen las insuficiencias del sistema actual con respecto al control de la asistencia del personal, administrado y visualizado vía Web, esto solucionará en forma eficiente la situación actual de asistencia del personal.

Objetivos

Objetivo General:

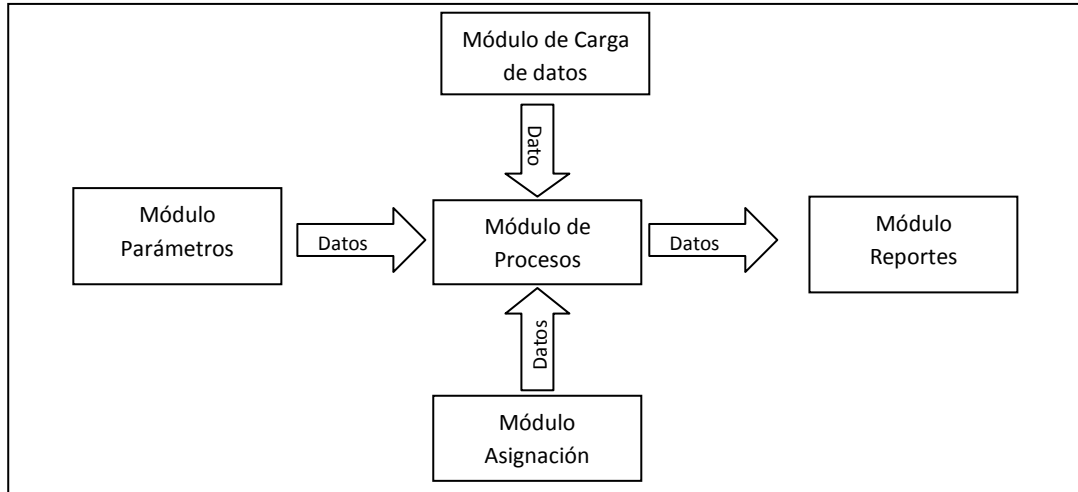
Optimizar el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante Sistema Web, asistido con un colector de datos biométrico.

Objetivos Específicos:

- Determinar los requerimientos y necesidades de la Institución con respecto al control de la asistencia de personal.
- Diseñar la arquitectura de los procesos y funciones del Sistema Web para control de asistencia de personal.
- Desarrollar e implementar el Sistema Web para control de asistencia de personal, utilizando herramientas libres.
- Realizar pruebas e Implantar el Sistema Web para control de asistencia de personal en la Dirección Ambiental de Imbabura.

Alcance

En este proyecto se va a desarrollar 5 Módulos:



Fuente: Propia

Figura 1: Módulos del proyecto

- **Módulo Parámetros.**
Crear, modificar, eliminar Usuarios.
Crear, modificar, eliminar parámetros del Turno.
Crear, modificar, eliminar Cargos.
Crear, modificar, eliminar Departamentos.
Crear, modificar, eliminar Secciones.
- **Módulo Asignación.**
Crear, modificar, eliminar Empleados.
Asignar Permisos a Empleados.
Asignar Vacaciones a Empleados.
Asignar Movilizaciones a Empleados.
- **Módulo de Carga de datos.**
Trasmisión de Datos mediante archivo de texto generado por reloj Biométrico.
- **Módulo de Procesos.**
Re cálculo de Horas E/S.
Re cálculo de Atrasos.
Re cálculo de Horas Extras.
Re cálculo de Horas Trabajadas.
- **Módulo Reportes.**
Reporte Personal (Vacación, Permisos, Movilización).
Reporte (Entradas y Salidas, Horas Extras, Atrasos, Horas Trabajadas).

Herramientas de desarrollo

La implementación del sistema Web de la institución se llevó a cabo con las herramientas:

	Herramientas	¿Por qué?
<u>Base de Datos:</u>	MySQL	Escalabilidad y flexibilidad, Alto rendimiento, Alta disponibilidad. Robusto soporte transaccional. Fortalezas en Web. Fuerte protección de datos
<u>Plataforma</u>	PHP	La Comunidad PHP, rendimiento, portabilidad, etc.
<u>IDE</u>	NetBeans 6.9	Entorno de desarrollo modular, compatible total con PHP.
<u>Servidor Aplicación</u>	Apache	Es robusto y compatible con PHP y NetBeans.
<u>Framework:</u>	<u>Symfony</u>	Symfony es maduro, estable, profesional y está muy bien documentado. Compatible con NetBeans 6.9.

Fuente: Propia

Tabla 1. Herramientas de desarrollo a usar y justificación

Impactos

- **Impacto social y tecnológico;**
 - Este trabajo, aporta grandes beneficios a nuestro entorno social, al realizar una contribución al estado con la parte más importante que es el control de la asistencia del personal de las instituciones públicas, que puede ser estandarizado a nivel nacional.
 - Además incentiva a la utilización de herramientas libres, como base de confianza para el desarrollo de más sistemas desarrollados en el

país y que puedan innovar tecnológicamente los procesos productivos manuales que se realizan hasta hoy.

- **Impacto educativo:**

- En la parte educativa sirve como base para incentivar a la utilización de software libre y que contribuya a seguir implementando programas desarrollados por estudiantes y profesionales de las instituciones educativas, para el fortalecimiento institucional con mejoras en las áreas de investigación e innovación tecnológica.

- **Impacto ecológico y económico :**

- El Ministerio del Ambiente, como un incentivo a la ecología, al utilizar éste sistema, disminuye la utilización de papel en las impresiones y promueve la implementación de más sistemas que utilicen menos recursos impresos, optimice los tiempos de consulta y controle de mejor manera a sus empleados.

Capítulo I

Inicio:

**Visión, Plan desarrollo de Software, Recolección y Análisis de Requerimientos de
la Institución.**

Introducción.

Entender que se va a construir.

1.1. Visión.

1.1.1. Propósito.

Para realizar este proyecto es necesario recolectar, analizar y describir las necesidades de la Institución y sus procesos con respecto a la asistencia del personal de la Dirección Provincial del Ambiente de Imbabura. Esta parte del documento va a describir los procesos requeridos por los participantes en el proyecto y los usuarios finales.

Las necesidades que a continuación se incluye han sido extraídas de las diferentes reuniones que se han celebrado con las autoridades de la Institución. En primer lugar está el Ing. Segundo Fuentes Director Provincial Ambiental de Imbabura, la Lcda. Mónica Maza Secretaria de la Institución; como se observa en las Actas de trabajo Núm. 1 del Anexo Núm. 8.

El presente proyecto tiene como objetivo registrar la asistencia del personal de dicha institución en una base de datos y realizar los cálculos con éstos, para realizar los respectivos reportes, todo mediante un sistema web. La recolección de datos cuenta con un reloj biométrico, donde registra fechas, hora de entrada y salida, con ésta información se puede lograr solucionar las necesidades.

Las necesidades específicas a satisfacer son:

- Debe integrarse los datos contenidos en los relojes biométricos de Ibarra y Cotacachi en la base de datos del Sistema.
- La descarga de los datos se realizará por medio de cable con conectores rs232, desde las secretarías de cada cantón.
- La administración del Sistema se lo realizará vía Web.
- Es necesario dar seguridad media a estos datos (Usuarios).
- El sistema realizará los reportes según lo acordado con las autoridades de la institución.
- Los reportes se podrán exportar a archivos con extensión.pdf (Acrobat).
- Los reportes en pantalla deberán tener un gráfico estadístico para su visualización.
- Los empleados podrán consultar sus entradas y salidas.

1.1.2. Alcance.

Esta sección del documento se refiere al proyecto sistema web elaborado por el proponente Sergio Benítez de la Facultad de Ingeniería en Ciencias Aplicadas, aplicado a la Dirección Provincial Ambiental de Imbabura localizado en Ibarra y Cotacachi, teniendo como objeto registrar calcular y reportar la asistencia de personal de la Institución vía Web.

1.1.3. Posicionamiento.

- Oportunidad de Negocio.

El sistema web que se pretende desarrollar, permitirá a la Dirección Provincial Ambiental de Imbabura el control de su personal, para analizar y gestionar los recursos económicos y de personal, para contribuir a la eficiencia del entorno laboral que tiene la institución. Y se pueda expandir a nivel nacional el presente proyecto.

¿Dónde estamos?

En la Dirección Provincial Ambiental de Imbabura se adquirió el dispositivo biométrico con su respectivo software, el cual no ha cumplido con todas las necesidades y expectativas de la institución, es por esto, que sin menospreciar dicho sistema, se puede investigar y realizar un nuevo sistema cumpliendo con los requerimientos de los usuarios.

¿Hacia dónde Vamos?

Este proyecto pretende dar a conocer la fortaleza de las herramientas libres y sistemas web para guardar y procesar la información que coleccionan medios biométricos que están dando resultados eficientes en el ámbito productivo, institucional, empresarial, etc., para control de su recurso humano.

¿Hacia dónde queremos ir?

Este proyecto sirve como base en el desarrollo de aplicaciones biométricas y de seguridad para control de personal, toda vez que, integrando nuevas necesidades y características se podrá implementar y estandarizar a nivel nacional e internacional como un medio gestor de los mismos.

¿Qué hacer ahora?

Para empezar, se necesita recolectar las necesidades de la institución y crear una planificación, analizar las herramientas a utilizar, estimar el tiempo de desarrollo, y lo más importante, hacer que no se complique el usuario, sino el desarrollador para que la interfaz sea fácil de usar.

- **Sentencias que definen el problema**

DEFINICIÓN	Inadecuado Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, por carencia de implementación de un sistema desarrollado a medida y que integre oficinas geográficamente dispersas vía web.
AFECTADOS	Autoridades, Empleados.
IMPACTO	Desinterés de las autoridades por el control que realizan a su recurso humano por la falta de capacitación en el manejo del sistema. Los empleados desconocen cómo se están registrando sus entradas y salidas.
SOLUCIÓN	<p>Poner a disposición una interfaz amigable al usuario, que sea de fácil manejo y que sirva para realizar los reportes a medida de su conveniencia.</p> <p>Especificar los requerimientos en la interfaz y reportes para que las necesidades sean solucionadas específicamente a la medida de lo que necesitan consultar.</p> <p>Descargar información del reloj biométrico para que no se sature la memoria en un tiempo reducido.</p> <p>Sentar una base para que sirva como medio de consulta para el análisis y toma de decisiones para gestionar el recurso de personal.</p>

Fuente: Propia

Tabla 1.1. Definición del Problema

- **Sentencia que define la posición del producto.**

PARA	<ul style="list-style-type: none"> ✓ Dirección de la Institución. ✓ Departamento Administrativo Financiero. ✓ Empleados.
QUIENES	Autoridades, Empleados.
EL NOMBRE DEL PRODUCTO	Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.
QUE	Carga Datos del reloj biométrico (fecha, hora E/S), Calcula, Consulta y Reporta la asistencia del personal de la Dirección Provincial del Ambiental de Imbabura.
¿NO CÓMO?	El sistema implementado que sólo puede estar instalado en un computador y realiza reportes impresos, mediante la carga dificultosa de los datos del reloj biométrico.
NUESTRO PRODUCTO	Mejora la carga de datos al sistema, la Interfaz es muy amigable, ayuda a realizar consultas y reportes vía web. Utiliza herramientas libres sin costo.

Fuente: Propia

Tabla 1.2. Posicionamiento del Producto

1.1.4. Descripción de participantes en el proyecto y usuarios.

Para realizar de una forma efectiva la recolección de requerimientos y modelamiento del sistema es necesario reunir a los participantes y usuarios, especificar el perfil de éstos, y sus necesidades e inquietudes, a continuación se detallan:

- Resumen de participantes del proyecto

NOMBRE	DESCRIPCIÓN	RESPONSABILIDAD
ING. Segundo Fuentes.	Director Provincial del ambiente de Imbabura.	Representa los usuarios posibles del sistema. Aprueba requisitos y procesos del proyecto. Provee recursos y elementos para el desarrollo del proyecto.
Lcda. Mónica Maza.	Secretaria de la Institución.	Representa a todos los usuarios que usarán el sistema Web. Responsable de la gestión de permisos y parámetros correspondientes con el sistema.

Fuente: Propia

Tabla 1.3. Resumen de participantes.

- Resumen de usuarios

NOMBRE	DESCRIPCIÓN	USUARIO
USUARIO	Es aquel que tendrá acceso a realizar la consulta de su asistencia mediante su login y contraseña. Empleado de la Institución.	Realiza la consulta de su asistencia vía web.

Fuente: Propia

Tabla 1.4. Resumen de usuarios.

- Entorno de usuarios.

Los usuarios tendrán acceso mediante un browser¹, y tendrán que digitar en la barra de dirección www.asistenciamai.com. Con un login y contraseña entrarán al sistema para realizar la respectiva consulta de su asistencia. Tendrán sus respectivos manuales de instalación y de usuario.

Los usuarios podrán visualizar:

Nombre Apellidos, Fecha, Hora Ingreso, Hora salida Almuerzo, Hora Entrada Almuerzo, Hora Salida, Atrasos, Horas Extras, Horas totales trabajadas, y visualización de permisos movilizaciones o vacaciones.

¹**Browser:** es un programa que permite ver la información que contiene una página web

Gráfico Estadístico:

Tendrán Acceso a observar en forma visual los resultados de cada consulta que hagan, con su respectivo gráfico estadístico de los atrasos, horas extras y horas trabajadas en el rango de fecha en que se crea la consulta.

Exportar a pdf:

La consulta se podrá exportar a pdf para su respectivo análisis.

1.1.5. Perfil de los participantes y usuarios.

DIRECTOR DE PROYECTO.

REPRESENTANTE	Ing. Segundo Fuentes.
DESCRIPCIÓN	Director Provincial del Ambiente de Imbabura.
RESPONSABILIDADES	Representa los usuarios posibles del sistema. Aprueba requisitos y procesos del proyecto. Provee recursos y elementos para el desarrollo del proyecto.
CRITERIO DE ÉXITO	Direcciona los requerimientos del proyecto.
GRADO DE PARTICIPACIÓN	Aprueba los requerimientos y funcionalidad del sistema.
COMENTARIOS	Ninguno.

Fuente: Propia
Tabla 1.5. Perfil director de proyecto.

ADMINISTRADOR DEL SISTEMA WEB.

REPRESENTANTE	Lcda. Mónica Maza.
DESCRIPCIÓN	Administra los usuarios, parámetros, y carga de datos al sistema.
RESPONSABILIDADES	Representa a todos los usuarios que usarán el sistema Web. Responsable de la gestión de permisos y parámetros correspondientes con el sistema.
CRITERIO DE ÉXITO	Verificar la seguridad del proyecto para que cumpla todos los parámetros que necesite el sistema.
GRADO DE PARTICIPACIÓN	Activa.
COMENTARIOS	Ninguno.

Fuente: Propia
Tabla 1. 6: Perfil Administrador.

- Perfiles de usuario.

USUARIO.

REPRESENTANTE	Empleados de la institución.
DESCRIPCIÓN	Empleados de la institución, que se encuentran laborando y están registrados en la institución como empleados.

TIPO	Usuario permanente.
RESPONSABILIDADES	Consultar y corregir falencias al momento de timbrar en sus ingresos y salidas a la institución.
CRITERIO DE ÉXITO	Sugerir y difundir esta nueva forma de realizar el marketing institucional.
GRADO DE PARTICIPACIÓN	Activa y constante.
COMENTARIOS	Ninguno.

Fuente: Propia
Tabla 1.7. Perfil usuario.

NECESIDADES DE LOS INTERESADOS Y USUARIOS.

NECESIDADES	PRIORIDAD	INQUIETUDES	SOLUCIÓN ACTUAL	SOLUCIÓN PROPUESTA
Integrar los 2 relojes biométricos de los cantones Ibarra y Cotacachi.	ALTA	Se necesita que los datos que generan los 2 relojes biométricos estén en un almacén de datos centralizado.	Envío del archivo que genera el sistema actual de control de asistencia en Excel para su respectivo análisis.	Diseñar e implantar un Sistema Web donde se recoja datos de los 2 cantones vía web.
Recoger los datos directamente del reloj biométrico al computador.	ALTA	El encargado de obtener los datos del reloj necesita no realizar muchos procesos para sacar la información del reloj.	Utilizando una flash memory para sacar los datos y un programa para convertir en un formato que reconozca el sistema.	Realizar una conexión mediante cable utp conectado al puerto serial (rs232), que tiene el reloj y el computador. Y un programa que descargue la información y vacíe la memoria del reloj inmediatamente.
Poder gestionar usuarios, carga y consulta de la asistencia vía web	ALTA	Las autoridades y empleados necesitan tener una forma de averiguar la asistencia de personal de la Institución.	Sistema en un solo computador.	En el sistema web poder introducir los parámetros y datos, y poder reportar los mismos mediante un browser.

Fuente: Propia
Tabla 1.8. Necesidades de los interesados y usuarios.

ALTERNATIVAS Y COMPETENCIA.

Adquirir un sistema Web de un costo alto, desarrollado por una empresa especializada en este tipo de sistemas de control de asistencia.

Seguir con el Sistema actual y adquirir la actualización del sistema que está en etapa de desarrollo y con software propietario.

1.1.6. Descripción global de producto.

Perspectiva del proyecto.

El proyecto de desarrollo debe generar una facilidad en la interfaz para ingreso de parámetros específicos, además de realizar la carga de los datos del reloj biométrico a la base de datos de una forma sencilla sin muchos procesos y presentar los reportes de una forma concreta y gráfica que es lo que necesita la Institución.

- Mostrará una interfaz sencilla para ingreso de datos mediante herramientas que ayuden a esta labor.
- Mostrará la forma de gestionar la seguridad de los usuarios que puedan acceder a este sistema.
- Mostrará una forma de cargar los datos muy sencilla, mediante herramientas que ayuden a este proceso.
- Mostrará reportes específicos y de forma gráfica de la asistencia del personal de la institución.

Resumen de características.

A continuación se mostrará un listado con los beneficios que obtendrá el cliente a partir del producto.

BENEFICIOS DEL USUARIO	CARACTERÍSTICAS QUE LO APOYAN
Reducir el tiempo para cargar los datos del reloj biométrico al sistema.	Mediante un extractor de datos del reloj, directo que está configurado para descargar los datos al computador y vaciar el reloj para no saturarlo. Con un solo clic y luego cargar el archivo a la base de datos.
Reducir el tiempo en la creación, edición y eliminación de parámetros que son necesarios para el funcionamiento del sistema.	Toda la interfaz debe contar con una forma fácil de creación, edición y eliminación directa y no con muchos vínculos, esto es posible con herramientas que se utilizarán para hacer un fácil manejo de los datos.
Visualizar la asistencia del personal.	Se puede consultar de cualquier parte del mundo los 365 días del año y durante las 24 horas.
Visualizar en un gráfico de barras los atrasos y horas extras.	Mediante herramientas de generación de gráficos se debe lograr este objetivo.
Poder guardar o almacenar las consultas en pdf para su respectivo análisis.	Mediante herramientas de generación de formato pdf se debe lograr este objetivo.

Fuente: Propia

Tabla 1.9. Resumen de características.

1.1.7. Costo y precios.

TIPO	CANTIDAD	DETALLE	COST UNIT	COSTO ACTUAL	COSTO REAL
HARDWARE	1	COMPUTADOR P4 Dual Core/ 1 GB RAM/250DD	750	750	0
	1	IMPRESORA MULTIFUNCION A TINTA HP 2480	80	80	0
	1	RELOJ BIOMETRICO WALSO WU5	750	750	0
	2	CABLE UTP Y CONECTORES PUERTO RS232	10	20	20
	1	HOSTING	120	120	120
	1	NOMBRE DOMINIO: www.asistenciamai.com	15	15	15
SUBTOTAL			1725	1735	155
SOFTWARE	1	MYSQL	0	0	0
	1	APACHE XAMP	0	0	0
	1	PHP	0	0	0
	1	NET BEANS V. 6,9	0	0	0
	1	SSOO WINDOWS 2003	200	200	0
	1	FRAMEWORK SYMFONY V. 1,4	0	0	0
SUBTOTAL			200	200	0
VARIOS	1	INTERNET X 9 MESES	190	190	120
	1	PAPELERIA GLOBAL	200	200	200
	1	MOVILIZACION	300	300	300
	1	VIATICOS	120	120	120
SUBTOTAL			810	810	740
SUBTOTAL				1935	895
IMPREVISTOS	15%			290,25	134,25
TOTAL				2225,25	1029,25

Fuente: Propia
Tabla 1.10. Costo y Precios

Licenciamiento e instalación (LIBRE).

• FINANCIAMIENTO.	
Dirección Provincial del Ambiente de Imbabura.	0,00
• FINACIAMIENTO PROPIO.	1029,25
TOTAL	1029,25

Se estimó conveniente realizar el cálculo de costo de desarrollo del sistema en caso de comercializarlo. Ver anexo número 10.

1.1.8. Característica global del producto.

Facilidad de acceso y uso.

El Sistema web esta desarrollado con herramientas libres, el mismo que será accesible mediante internet, su navegación se realiza con las flechas del teclado o los movimientos del ratón, el navegante realiza sus consultas de manera fácil y amigable con el usuario

1.1.9. Seguridad.

Restricciones en la administración del sistema con respecto a la información, además sólo las autoridades y empleados podrán acceder a esta información.

Niveles de acceso a la aplicación:

1. Administrador: Acceso Total a administración de parámetros (Crear, Editar, Borrar), carga y cálculo de datos, generación de reportes.
2. Empleado: Acceso a Consulta de Entradas y salidas, atrasos, horas trabajadas, permisos, vacaciones y movilizaciones.
3. Administrador Cotacachi: Carga de datos del reloj biométrico de Cotacachi al Sistema.

1.1.10. Otros requisitos del producto.

Requisitos de documentación.

- Manual de operador.
- Manual usuario.

Requisitos de Calidad.

Es necesario establecer un estándar de calidad para que el producto informático cumpla con su objetivo, desde su estructura misma hasta la presentación de la documentación, para esto, se necesita cumplir con las normas que define la Metodología.

1.2. Plan de desarrollo de software.

El sistema Web será desarrollado usando la metodología RUP, la misma que permitirá utilizar sus normas para definir el proyecto, esto ayuda a organizar de mejor manera tanto para desarrollo como para documentación. Parte de la documentación es el presente documento.

1.2.1. Propósito.

Este proyecto tiene como propósito el analizar, planificar y delimitar el diseño, desarrollo e implementación de un software específico para solucionar las necesidades de la institución con un mínimo costo. Pudiendo controlar el tiempo de desarrollo.

Se debe tener en cuenta todos los factores y variables a utilizar para que el sistema tenga éxito y se culmine.

Los participantes del proyecto son:

Analista del proyecto, el mismo que define los tiempos, esquemas, recursos, herramientas entre otros, a utilizarse en el desarrollo del cronograma de actividades.

Miembros del equipo de desarrollo que ejecutan y desarrollan la planificación del cronograma. Además aportan con ideas y soluciones adicionales.

1.2.2. Alcance.

Mediante la metodología RUP se puede producir el plan de desarrollo de este proyecto, esto es para obtener una documentación clara y precisa. En el desarrollo se genera el documento *Visión* quien describe la estructura del producto. La recolección de necesidades se lo hizo con personas que emplean directamente el sistema, lo cual arrojó, una buena base para definir el documento Visión, el cual se podrá modificarlo conforme se vaya desarrollando, con la actualización a través de sus versiones para ajustar a las interacciones que se generen con el avance del proyecto.

1.2.3. Vista general del proyecto.

Luego de varias reuniones que se realizó con los participantes del proyecto y con ayuda de recomendaciones y exposición de las necesidades de la persona encargada de generar la información de la asistencia del personal, además de tomar en cuenta alguna de las características del actual sistema, y con la ayuda del documento Visión se logra dar un enfoque general del proyecto, el cual se detalla a continuación, cuyo objetivo es poder satisfacer las necesidades de la institución, la cual aportó de manera generosa con toda la información.

- **Módulo Parámetros.**

Crear, modificar, eliminar **Usuarios**, son los usuarios que van a ingresar al sistema para controlar la seguridad del mismo.

Crear, modificar, eliminar parámetros del **Turno** (horas y minutos de entrada y salida).

Crear, modificar, eliminar **Cargo**, son los cargos que tienen los empleados para poder identificarlos en su desempeño.

La institución cuenta con departamentos que a su vez se dividen en secciones y que contienen sus respectivos empleados, por lo que hay que crearlas para asignar al empleado.

Crear, modificar, eliminar **Departamentos**.

Crear, modificar, eliminar **Secciones**.

- **Módulo Asignación.**

Este módulo es necesario para Crear, modificar, eliminar Empleados. A su vez es necesario asignar permisos, movilizaciones y vacaciones si ha tenido el empleado.

- **Módulo de Carga de datos.**

La parte fundamental del sistema se encuentra en el módulo de carga de datos, sirve para la transmisión de Datos generado por reloj biométrico. Es importante explicar que el reloj biométrico descarga un fichero de tipo plano que contiene código, fecha, y hora de entrada o salida del empleado, con lo cual deberá manipularse al momento de cargar al sistema de esta forma.

- **Módulo de Procesos.**

En esta etapa se debe calcular los minutos de atrasos, horas y minutos extras, horas y minutos trabajados, con los datos ingresados a la base de datos mediante un archivo de texto que se lo realiza en el anterior módulo.

- **Módulo Reportes.**

La consulta se lo hace con el código del empleado un rango de fechas, en pantalla debe aparecer; los datos de ingresos y salidas del empleado, en horas y minutos los atrasos, extras, y horas trabajadas, además, y es lo más importante, si ha tenido permiso, movilización o vacación el empleado.

Adicionalmente debe generar un reporte de permisos, movilización o vacaciones que haya tenido el empleado, para poder analizarlo.

Es importante el gráfico estadístico y la generación de archivos pdf, para su posterior análisis.

1.2.4. Puntos clave.

De las entrevistas se generaron puntos claves a implementar:

Puntos Fundamentales:

- Seguridad en la presentación de la información.
- Adaptabilidad del usuario y la facilidad de uso.
- Facilidad para cargar los datos del reloj biométrico al sistema.
- Incorporación del sistema para los 2 cantones (Ibarra-Cotacachi).

1.2.5. Entregables del proyecto.

Los artefactos a desarrollarse se detallan a continuación, hay que tomar en cuenta que de acuerdo a la metodología RUP, los procesos pueden cambiar e incrementarse conforme se realiza el proyecto por lo cual todos los artefactos son objeto de modificaciones, pero con cada interacción e hitos se trata de estabilizarla.

- **Visión.**

En este documento se define la perspectiva de los usuarios con respecto al proyecto, especificando las necesidades y características del mismo.

- **Plan de desarrollo del software.**

Es el presente documento.

- **Glosario.**

Se establece una descripción de los términos que se utiliza en este proyecto.

- **Especificaciones de casos de uso.**

Se describe detalladamente los flujos de procesos, requisitos no-funcionales asociados, los actores externos que pueden incluirse en los procesos.

- **Prototipos de interfaces de usuario.**

En esta parte se da a conocer una idea gráfica de cómo estarán distribuido todos los procesos requeridos, los cuales se podrán modificar o actualizar conforme se desarrolle el proyecto, pero que tendrán que ser una base sólida de lo que quiere la institución. Los prototipos se realizarán en forma gráfica en programas que tengan la opción de realizar formularios y herramientas para la visualización de la interfaz requerida.

- **Modelo de análisis y diseño.**

En este modelo se representa los casos de uso en diagrama de clases y las relaciones que existen entre ellas.

- **Modelo de datos.**

Este modelo describe la representación lógica de los datos, de acuerdo con el enfoque para modelado relacional de datos.

- **Modelo de implementación.**

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar, al final de la fase de elaboración, posteriormente tiene bastante refinamiento).

- **Modelo de despliegue.**

Este modelo muestra el despliegue, la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

- **Casos de prueba.**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración.

- **Lista de riesgos.**

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ubicados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación. Ver Anexo Núm. 2.

- **Manual de instalación.**

Este documento incluye las instrucciones para realizar la instalación del producto.

- **Material de apoyo al usuario final.**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: guías del usuario, guías de operación, guías de mantenimiento.

- **Producto.**

Las carpetas con el contenido del sistema serán almacenadas en CD con la guía para su respectiva instalación. Todas las versiones serán incluidas para poder observar los avances.

1.2.6. Organización del proyecto.

- **Participantes en el proyecto.**

- **Jefe de proyecto o tutor.** Con una experiencia en metodologías de desarrollo, herramientas CASE y notaciones, en particular la notación² UML y el proceso de desarrollo RUP.
- **Programador.** Los conocimientos que poseen y con el que debe cumplir este perfil es de: conocimiento amplio en programación web con tecnología php, usando el Framework Symfony; conocimiento en UML; manejar base de datos de MySQL y configuración del servidor Apache.
- **Interfaces externas.**

Los empleados y autoridades que usen este sistema serán los que brinden recomendaciones y sugerencias.

²**Notación:** sistema de signos convencionales que se adoptan y utilizan para expresar determinados conceptos de una disciplina concreta

▪ **Roles y responsabilidades.**

Puesto	Responsabilidad
Jefe de Proyecto.	Es de suma importancia debido a que es generalmente quien asume la dirección del proyecto encargándose así de proveer del personal necesario, los recursos imprescindibles y de tomar las decisiones que ayuden a que el proyecto cumpla con los objetivos propuestos.
Programador.	Contribuir con sus conocimientos en el desarrollo del proyecto, creando versiones del proyecto para poder adquirir recomendaciones y mejoras hasta ejecutar el entregable final. Es responsable del desarrollo del proyecto, modelo de datos, seguridad y accesibilidad.

Fuente: Propia

Tabla 1.111. Roles y Responsabilidades

1.2.7. Plan del Proyecto.

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

- **Plan de las fases.**

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra la distribución de tiempos y el número de iteraciones de cada fase (para las fases de construcción y transición es sólo una aproximación muy preliminar).

Fase	Nro. Iteraciones	Duración
Fase de Inicio.	1	6 semanas.
Fase de Elaboración.	1	10 semanas.
Fase de Construcción.	5	24 semanas.
Fase de Transición.	1	4 semanas.

Fuente: Propia

Tabla 1.12. Plan de Fases

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

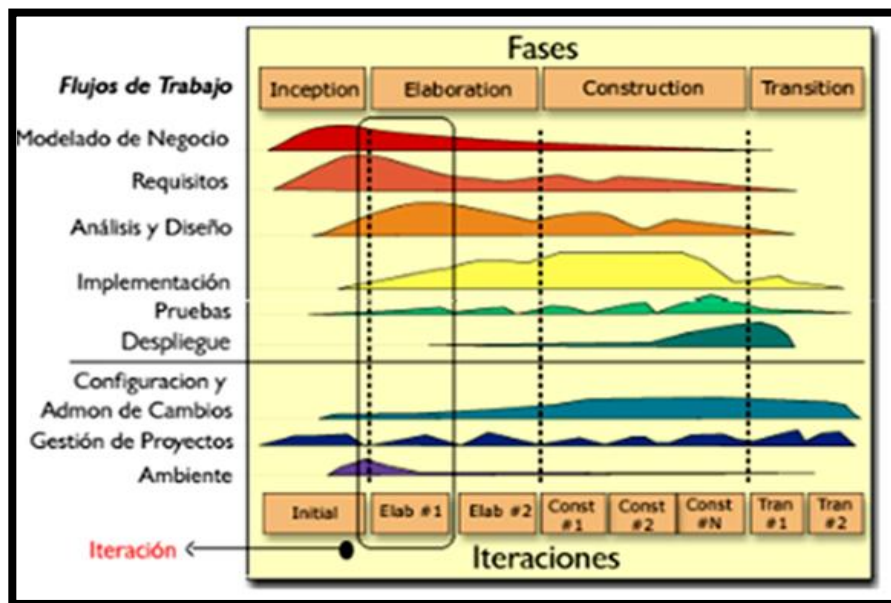
Descripción	Hito
Fase de especificación	<p>En esta fase desarrollará los requisitos del modelo desde la perspectiva del usuario, los cuales serán establecidos en el artefacto "Visión".</p> <p>Se tiene que claramente especificar los requisitos o funcionalidades específicas del sistema que se están solicitando.</p>
Fase de planeación	<p>En esta fase se analizarán los requisitos y se desarrollará un prototipo de arquitectura (incluyendo las partes más relevantes y/o críticas del sistema). Al final de esta fase, todos los casos de usos correspondientes a requisitos que serán implementados en la primera release de la fase de desarrollo deben estar analizados y diseñados.</p> <p>El cuándo dependerá del número de personas implicadas en el proyecto, así como de limitaciones impuestas por los receptores del proyecto, quienes serán los que fijen las condiciones temporales del proyecto. Es importante crear un modelo que deje plenamente satisfechos a sus usuarios en torno a lo que ellos esperan, además se deberá establecer los costos incurridos en el manejo del recurso.</p>
Fase de desarrollo	<p>Durante la fase de desarrollo se terminan de analizar y diseñar todos los casos de uso, refinando el modelo de análisis / diseño, el producto se construye en base a dos iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el usuario. El hito que marca el fin de esta fase es la versión de la release 3.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.</p>
Fase de transmisión de tecnología y puesta en marcha	<p>En esta fase se presentarán el release final para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase, incluye la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.</p>

Fuente: Propia

Tabla 1.13. Plan de Fases-Hitos

- **Calendario del Proyecto.**

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de inicio y elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido, marca el énfasis de cada disciplina (workflow³) en un momento determinado del desarrollo.



Fuente: Ejemplo Plan de desarrollo de software RUP

Figura 1.1. Descripción del Proceso Unificado

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado completo, suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

³ WORKFLOW: Flujos de Trabajo

Disciplinas / Artefactos generados o modificados durante la fase de inicio	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 2	Semana 6
Requisitos		
Glosario	Semana 2	Semana 5
Visión	Semana 2	Semana 6
Modelo de Casos de Uso	Semana 3	siguiente fase
Especificación de Casos de Uso	Semana 4	siguiente fase
Especificaciones Adicionales	Semana 4	siguiente fase
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 4	siguiente fase
Modelo de Datos	Semana 4	siguiente fase
Implementación		
Prototipos de Interfaces de Usuario	Semana 4	Semana 7
Modelo de Implementación	Semana 4	siguiente fase
Pruebas		
Casos de Pruebas Funcionales	Semana 6	Semana 7
Despliegue		
Modelo de Despliegue	Semana 6	Semana 8
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones	Semana 1	Semana 7
Ambiente	Durante todo el proyecto	

Fuente: Propia

Tabla 1.14. Roles y Responsabilidades – Fase de inicio

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 6	aprobado
Requisitos		
Glosario	Semana 6	aprobado
Visión	Semana 6	aprobado
Modelo de Casos de Uso	Semana 3	Semana 7
Especificación de Casos de Uso	Semana 4	Semana 7
Especificaciones Adicionales	Semana 4	Semana 7
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 7	Revisar en cada iteración
Modelo de Datos	Semana 8	Revisar en cada iteración
Implementación		
Prototipos de Interfaces de Usuario	Semana 10	Revisar en cada iteración
Modelo de Implementación	Semana 11	Revisar en cada iteración
Pruebas		
Casos de Pruebas Funcionales	Semana 12	Revisar en cada

		iteración
Despliegue		
Modelo de Despliegue	Semana 12	Revisar en cada iteración
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1 y planes de las Iteraciones	Semana 1	Revisar en cada iteración
Ambiente	Durante todo el proyecto	

Fuente: Propia

Tabla 1.15. Roles y Responsabilidades – Fase de elaboración

- **Seguimiento y control del proyecto.**

▪ **Gestión de requisitos.**

Los requisitos del sistema son especificados en el artefacto “Visión”. Cada requisito tendrá una serie de atributos tales como importancia, estado, iteración donde se implementa, etc.

▪ **Control de plazos.**

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto.

▪ **Control de calidad.**

Los defectos detectados en las revisiones y formalizados también en una solicitud de cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias. Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist⁴ incluidas en RUP.

⁴**Checklist:** lista de comprobaciones.

- **Gestión de riesgos.**

A partir de la fase de inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para mitigarlos o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

- **Gestión de configuración.**

Se realizará una gestión de configuración para llevar un registro de los artefactos generados y sus versiones.

1.3. Captura de requisitos.

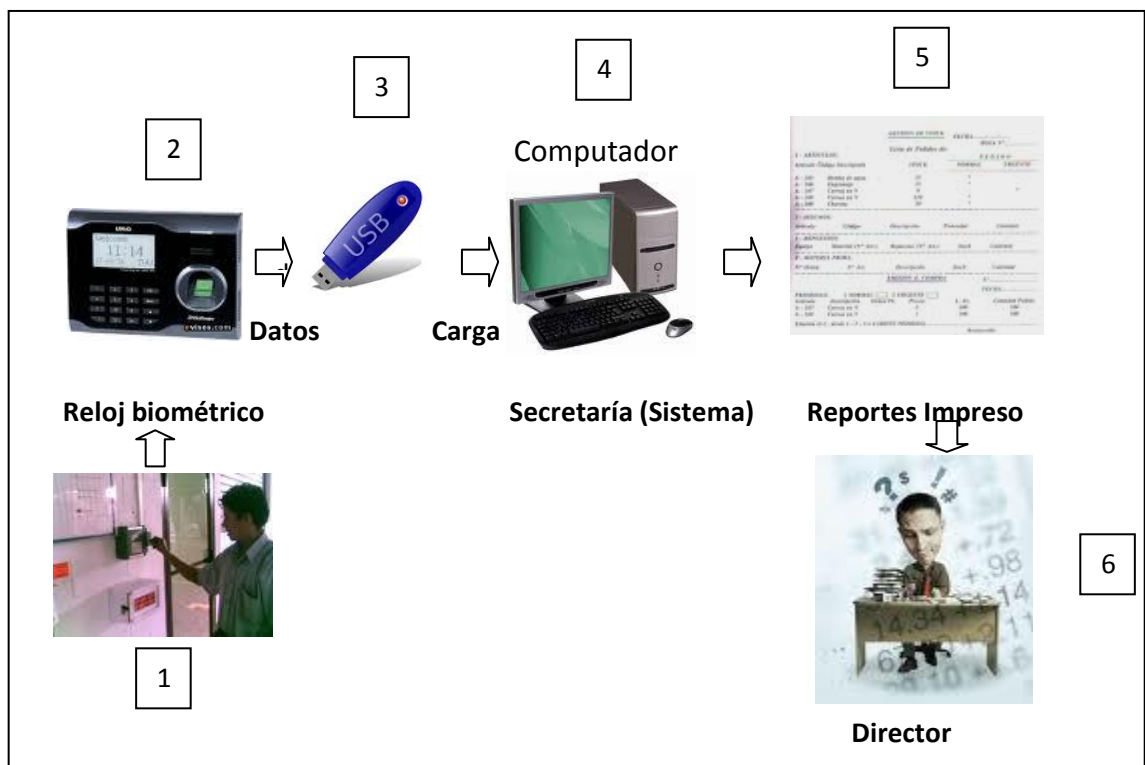
En esta sección, se va a describir en una forma gráfica como está formado el sistema actual y también los requerimientos nuevos de los usuarios.

Para esto se utilizará diagramas de caso de uso y gráficos que describan en forma clara y precisa el funcionamiento y estructura. Además define los recursos tecnológicos de hardware, software y telecomunicaciones para la realización del presente proyecto.

1.3.1. Análisis del sistema actual.

En esta etapa se trata de describir los actores y el funcionamiento del sistema actual en forma gráfica.

La Arquitectura gráfica es la siguiente:



Fuente: Propia.

Figura 1.2. Arquitectura Actual.

Descripción:

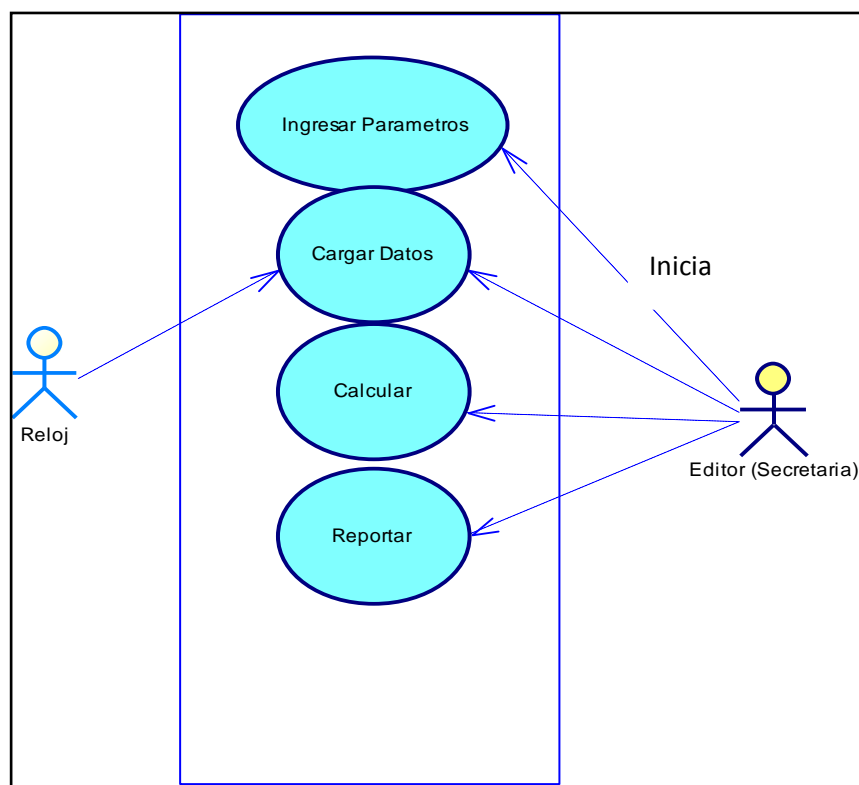
1. El primer paso no tiene ningún inconveniente al momento de recolectar los datos, la falencia es del empleado cuando no timbra correctamente o simplemente no timbra.
2. El reloj de Ibarra y Cotacachi puede almacenar hasta 60000 transacciones (número de timbradas) y hasta 5000 usuarios, posee una memoria que almacena esta información, pero es necesario vincular al reloj de Cotacachi al mismo sistema lo cual se soluciona siendo el sistema Web.
3. En este punto se halla una falencia. El reloj fue adquirido para que la información sea extraída por medio de una Flash Memory, aquí surge el inconveniente, que cuando hay daños en el puerto USB no se descargan los datos, por lo que, es necesario la conexión vía cable por el puerto RS 232 el cual posee el reloj y el computador; con esto se soluciona la salida de datos del reloj.
4. Otro inconveniente es la información generada por el reloj y almacenada en Flash Memory, posee todos los registros de timbradas desde su instalación, lo que hace que la carga al sistema sea muy lenta. Esto se lo soluciona mediante la generación de un programa que obtiene los datos y deja vaciando la memoria del reloj con lo cual solo se extrae los datos requeridos a la fecha de extracción.
5. En la generación de reportes, el sistema instalado actualmente no posee un formato diseñado para la toma de decisiones, ni ajustado a requerimientos de la institución; además sólo se puede consultar desde un computador e imprimir en el formato prediseñado por el sistema.

6. Si los reportes no generan resultados deseados, no existe un buen control por parte de las autoridades.

1.3.1.1. Casos de negocios.

Ahora vamos a explicar en forma gráfica los procesos de la institución y sus actores:

Identificación de Roles del Entorno del Negocio:



Fuente: Propia.

Figura1.3. Diagrama casos de uso global del negocio.

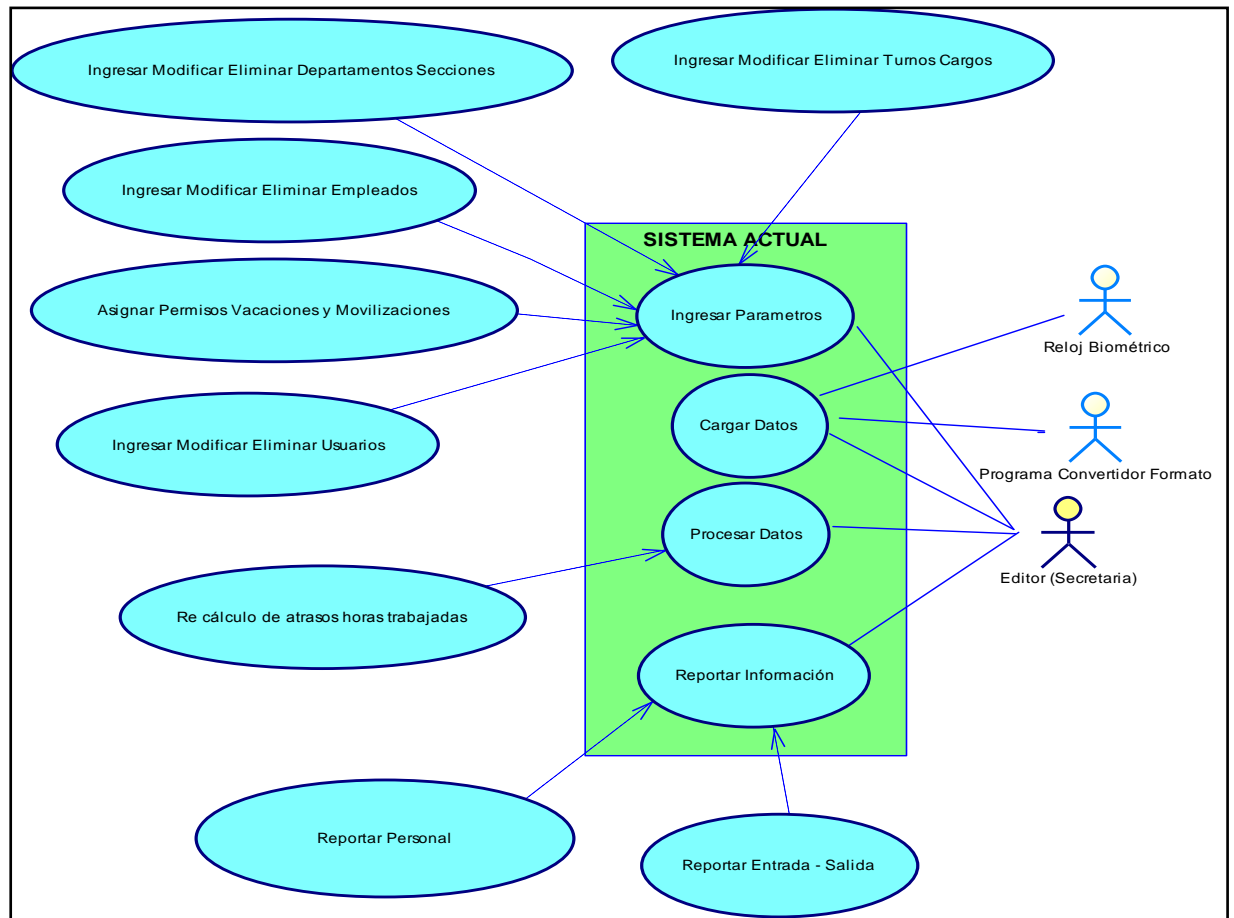
Descripción de los Casos de Uso global del Negocio:

1. La Secretaria, Ingresa los datos de parámetros como son: Departamentos, Secciones de la Institución, Turnos, Cargos y luego crea a los Empleados con su respectivo código generado en el reloj biométrico.

2. Luego, la misma persona Carga los datos generados por el reloj al sistema, mediante un archivo que descarga del reloj biométrico.
3. Con este archivo se procede a convertirlo para que el sistema pueda tener el formato para que pueda cargar los datos con un programa proporcionado por el vendedor del reloj.
4. Se carga los datos. Cargado los datos se realiza el re cálculo con los datos para generar los minutos de atraso y horas trabajadas dependiendo los cambios que se hayan realizados.
5. Por último se hace necesario realizar los reportes de Entradas y Salidas de la institución, dependiendo los empleados y fechas solicitadas por las autoridades, en forma impresa. Todo es manipulado por una persona encargada (Secretaria). Y no es accesible para ningún empleado.

1.3.1.2. Modelo de casos de uso.

Identificación de Casos de Uso del Sistema Actual.



Fuente: Propia.

Figura 1.4. Diagrama Casos de Uso del Sistema Actual

Descripción de los Casos de Uso del Sistema Actual:

a) Ingresar Parámetros:

1. Descripción Breve:

El caso de uso describe el proceso Ingresar Parámetros, para el acoplamiento de los datos del Reloj Biométrico con el Sistema de Control de Asistencia.

2. Flujo Básico de Eventos:

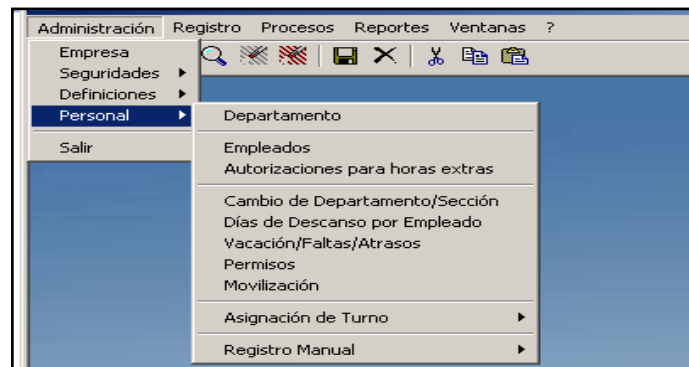
1. La secretaria Ingresa los parámetros de Turnos, Cargos de los Empleados.

2. Crea los Departamentos que existen en la Institución, con sus respectivas secciones.
3. Luego, puede crear a los empleados con el código que se grabó en reloj biométrico, y con los respectivos datos personales y parámetros ingresados anteriormente como son los de turno, cargo, departamento y sección al que pertenece.
4. Se puede asignar permisos, vacaciones y movilizaciones del personal.
5. Aquí se puede crear modificar o eliminar a nuevos usuarios.

Nota: Toda esta parte es de administración.

3. Precondiciones:

1. Es necesario tener **creado el código o pin del empleado en el reloj (1 a 5000)**, además de seguir la secuencia del 1 a 4 numeral anteriormente descritos en el flujo de eventos.



Fuente: Sistema Full Time.

Figura 1.5. Pantalla parámetros y asignación del sistema actual.

b) Carga Datos:

1. Descripción Breve:

El caso de uso describe el proceso Carga de datos generado del reloj biométrico al Sistema.

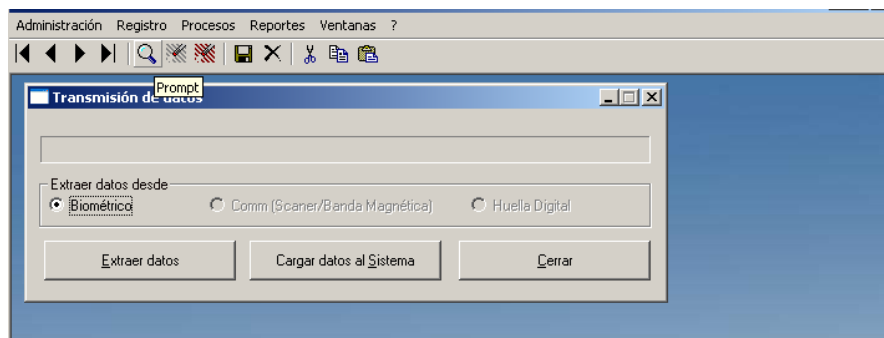
2. Flujo Básico de Eventos:

1. La secretaria debe descargar en un flash memory la información generada por el reloj biométrico.
2. Transformar el archivo generado por el reloj a un formato para que pueda ser cargado por el sistema.
3. Abre el sistema y escoge la opción de registro.
4. Luego abre con un clic en Carga de datos al sistema el cual le permite escoger el archivo que ya esta transformado para ser cargado.
5. Por último debe salir un mensaje de carga completa.

Nota: Toda esta parte es de carga de datos.

3. Precondiciones:

1. Es necesario tener **creado al empleado** en el sistema con el mismo código grabado en el reloj, de lo contrario se aborta la carga.



Fuente: Sistema Full Time.

Figura 1.6. Pantalla Carga datos del sistema actual.

c) Procesar Datos:

1. Descripción Breve:

El caso de uso describe el proceso de cálculo de atrasos horas trabajadas, con los datos cargados del reloj biométrico al Sistema.

2. Flujo Básico de Eventos:

1. Hacer clic en proceso.
2. Escoger la opción Re-cálculo de Horas.
3. Escoger el departamento y sección y el código del empleado (para uno solo) para todos en blanco.
4. Ingresar la fecha inicial y final del proceso de re-cálculo.

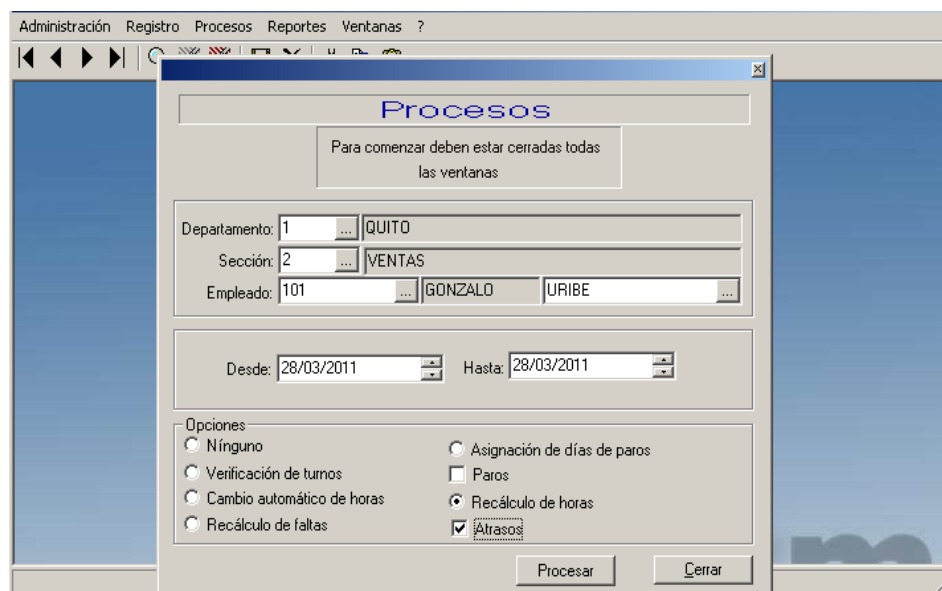
Nota: Entre más largo es el rango de la fecha se demora mucho más.

5. Por último procesar.

Nota: Toda esta parte es de Re-cálculo de horas.

3. Precondiciones:

1. Es necesario tener creado al empleado, cargada la información del reloj al sistema.



Fuente: Sistema Full Time.

Figura 1.7. Pantalla cálculo de Resultados del sistema actual.

d) Reportar Información:

1. Descripción Breve:

El caso de uso describe el proceso de reportes de resultados de los cálculos generados con la información proporcionada del reloj.

2. Flujo Básico de Eventos:

1. Hacer clic en reportes.
2. Escoger el reporte que se desee.
3. Ingresar parámetros de departamento y sección y el código del empleado (para uno solo) y fecha inicio y final a consultar.
4. Escoger si se desea imprimir, exportar (Excel, Word).
5. Por último Ok.

Nota: Toda esta parte es de Reportes.

3. Precondiciones:

1. Es necesario tener creado al empleado, cargada la información del reloj al sistema y realizado el re-cálculo.

Entradas y Salidas																		
1 of 1																		
29/03/2011																		
1 QUITO																		
2 VENTAS																		
URIBE GONZALO																		
Turno 3 ESPECIALI																		
Fecha	Fecha	Ent.	Sal.	Ent.	Sal.	Ent.	Sal.	Ent.	Sal.	Ent.	Sal.	Ent.	Sal.	Ent.	Sal.	I.H.	I.H.	
Entrada	Salida	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Trab.	Sistem a	Atra:	
08/04/2009	09/04/2009	17:00	07:00	00:00	00:00	00:00	00:00	00:00	00:00	00:00	00:00	00:00	00:00	00:00	00:00	12:00		
Turno 1 JORNADA UN																		
Fecha	Fecha	Ent.	Sal.														I.H.	I.H.
Entrada	Salida	Trab.	Trab.														Sistem a	Atra:
01/04/2009	01/04/2009	08:00	18:00														9:00	
Turno 2 NORMAL																		
Fecha	Fecha	Ent.	Salid	Entrad	Sal.												I.H.	I.H.
Entrada	Salida	Trab.	Alm.a	Trab.	Trab.												Sistem a	Atra:
02/04/2009	02/04/2009	07:00	14:00	15:00	18:00												8:30	
03/04/2009	03/04/2009	14:00	15:00	18:00	00:00												1:00	5:20
04/04/2009	04/04/2009	07:00	14:00	18:00	00:00												5:30	
MEDINA DAVID																		
Turno 2 NORMAL																		
Fecha	Fecha	Ent.	Salid	Entrad	Sal.												I.H.	I.H.
Entrada	Salida	Trab.	Alm.a	Trab.	Trab.												Sistem a	Atra:

Fuente: Sistema Full Time.

Figura 1.8. Pantalla Reporte de Resultados del sistema actual generado en cristal report.

1.3.2. Requisitos nuevos de los Usuarios.

Como se observa en el proceso del sistema actual hay muchas cosas que pueden suprimirse, hacer más fácil la carga y además ajustar los reportes a las necesidades de la Institución.

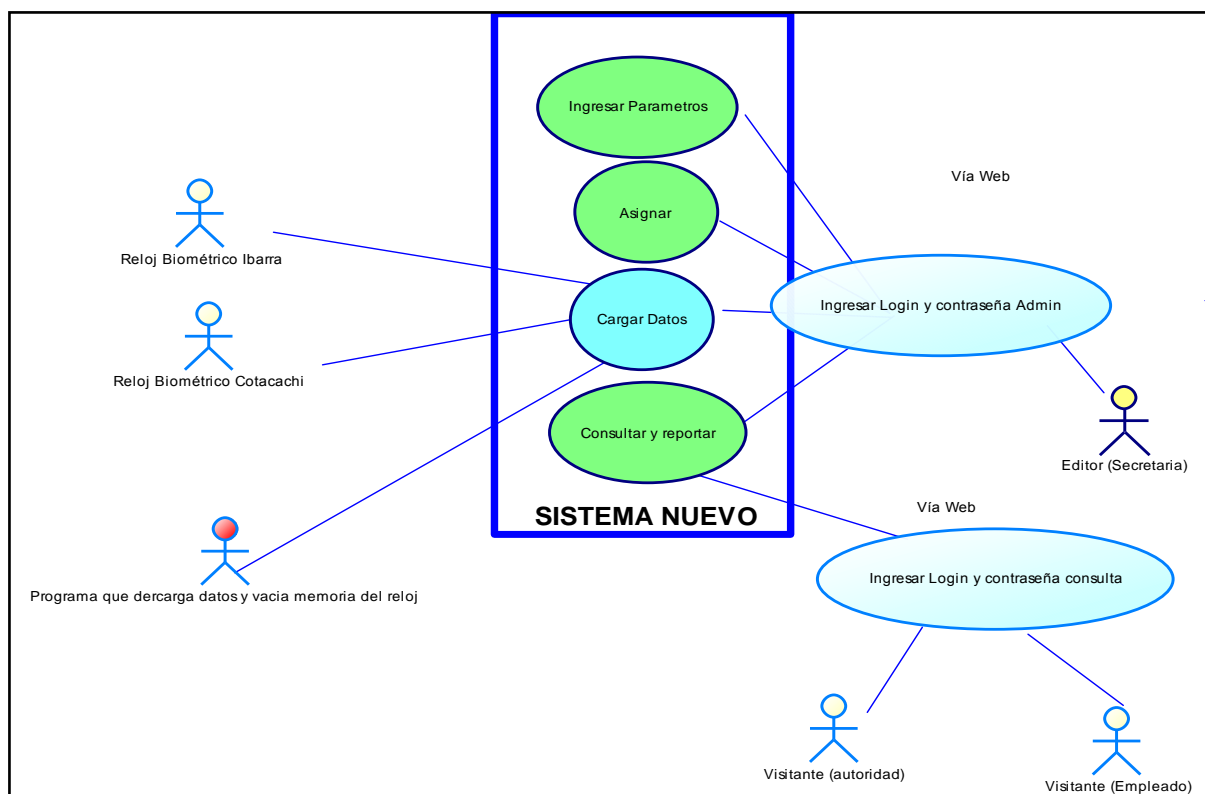
Se ha realizado varias reuniones con las autoridades de la Institución exponiendo sugerencias y recogiendo necesidades, los cuales están resumidos en las Actas de Trabajo Núm. 1 que se adjuntan en el Anexo Núm. 8.

En resumen consta lo siguiente:

- a) El sistema nuevo debe integrar los datos de los relojes biométricos de Ibarra y Cotacachi en una sola base de datos.
- b) La descarga de datos se realizará mediante cable por medio del puerto rs232 del reloj para salida y entrada de datos en el puerto del computador.
- c) La administración del sistema se lo realizará vía Web, es decir el sistema deberá estar subido en la red de Internet.
- d) La administración deberá estar sometido a seguridades con usuario y contraseña que se asignará al responsable.
- e) Los usuarios podrán acceder a consultas y reportes con un usuario y contraseña asignada a los empleados.
- f) El reporte deberá ser según lo acordado entre autoridades.
- g) Los reportes deberán poder exportarse a pdf.
- h) Por último la consulta debe tener un gráfico dinámico, según la consulta cambie su descripción.

En adelante se aclara en forma gráfica y textual la forma en que se va a solucionar.

1.3.2.1. Modelo de casos de uso.



Fuente: Propia.

Figura 1.9. Casos de uso de requerimientos nuevos

Descripción de los Casos de Uso de requerimientos nuevos:

a) Cargar Datos:

1. Descripción Breve:

El caso de uso describe el proceso de carga de datos con los nuevos requerimientos de los usuarios como son la integración de los dos relojes biométricos y el programa que descargue la información y resetee la memoria del reloj.

2. Flujo Básico de Eventos:

1. Ingresar al programa de descarga de datos con su respectiva contraseña de administrador.
2. Descargar datos al disco duro del servidor.
3. Ingresar en el sistema y escoger el botón carga.
4. Buscar en el disco duro el archivo para cargar datos al sistema.
5. Hacer clic en cargar.

3. Precondiciones:

1. Es necesario tener creado al empleado con el mismo código que se guardó en el reloj biométrico.

b) Ingresar al sistema:

1. Descripción Breve:

El caso de uso describe el proceso de ingreso por medio de un usuario y contraseña para consultas y administración del sistema vía web.

2. Flujo Básico de Eventos:

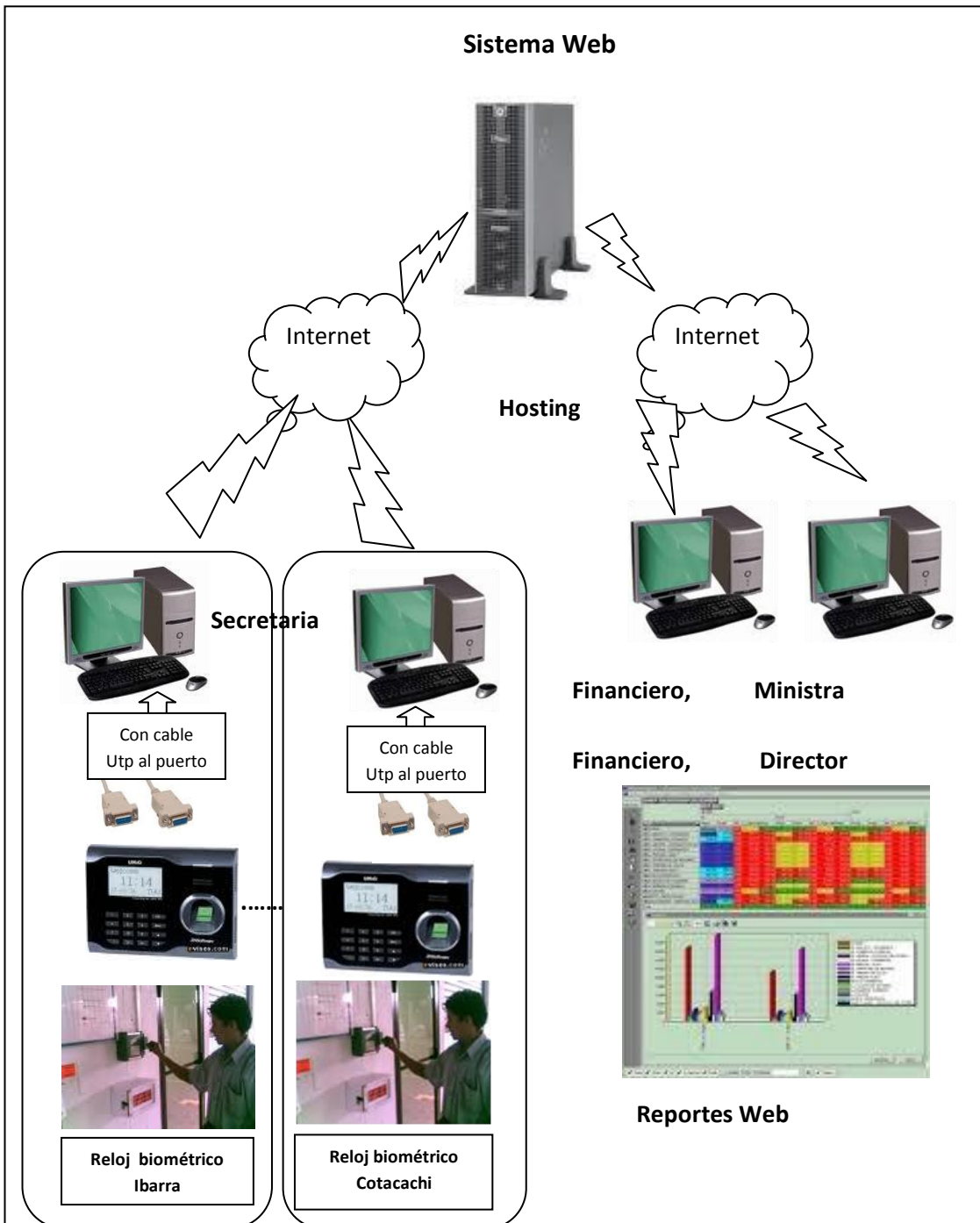
1. Ingresar al programa introduciendo el usuario y contraseña.
2. Si es administrador accederá a las opciones de ingreso de parámetros, carga y consulta.
3. Si es usuario accederá sólo a las opciones de consulta.

3. Precondiciones:

1. Es necesario tener creado al empleado, cargada la información del reloj al sistema.

1.3.3. Descripción del Sistema Propuesto.

1.3.3.1. Arquitectura del Software.



Fuente: Propia.

Figura 1.10. Arquitectura del Sistema Propuesto

Se puede ver la importancia de usar los medios biométricos como forma de control y optimización de la asistencia del personal (Fig. 1.9.).

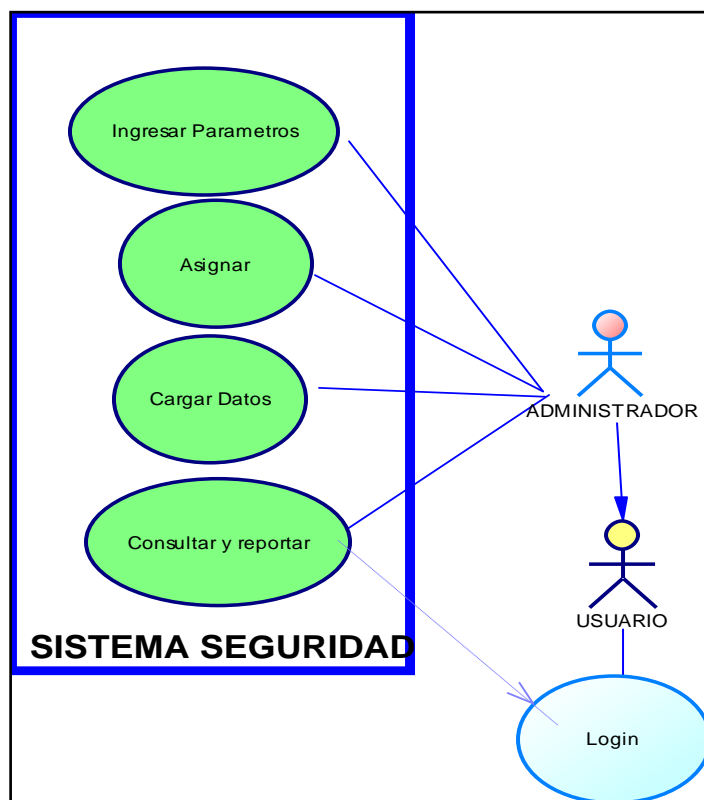
1.4. Especificación del Sistema.

1.4.1. Descripción del Sistema.

En esta parte del documento se describe los casos de uso identificados en los requerimientos y arquitectura recogida y descrita.

1.4.1.1. Especificación de Casos de Uso.

Modelo de casos de uso Seguridad y Accesos.

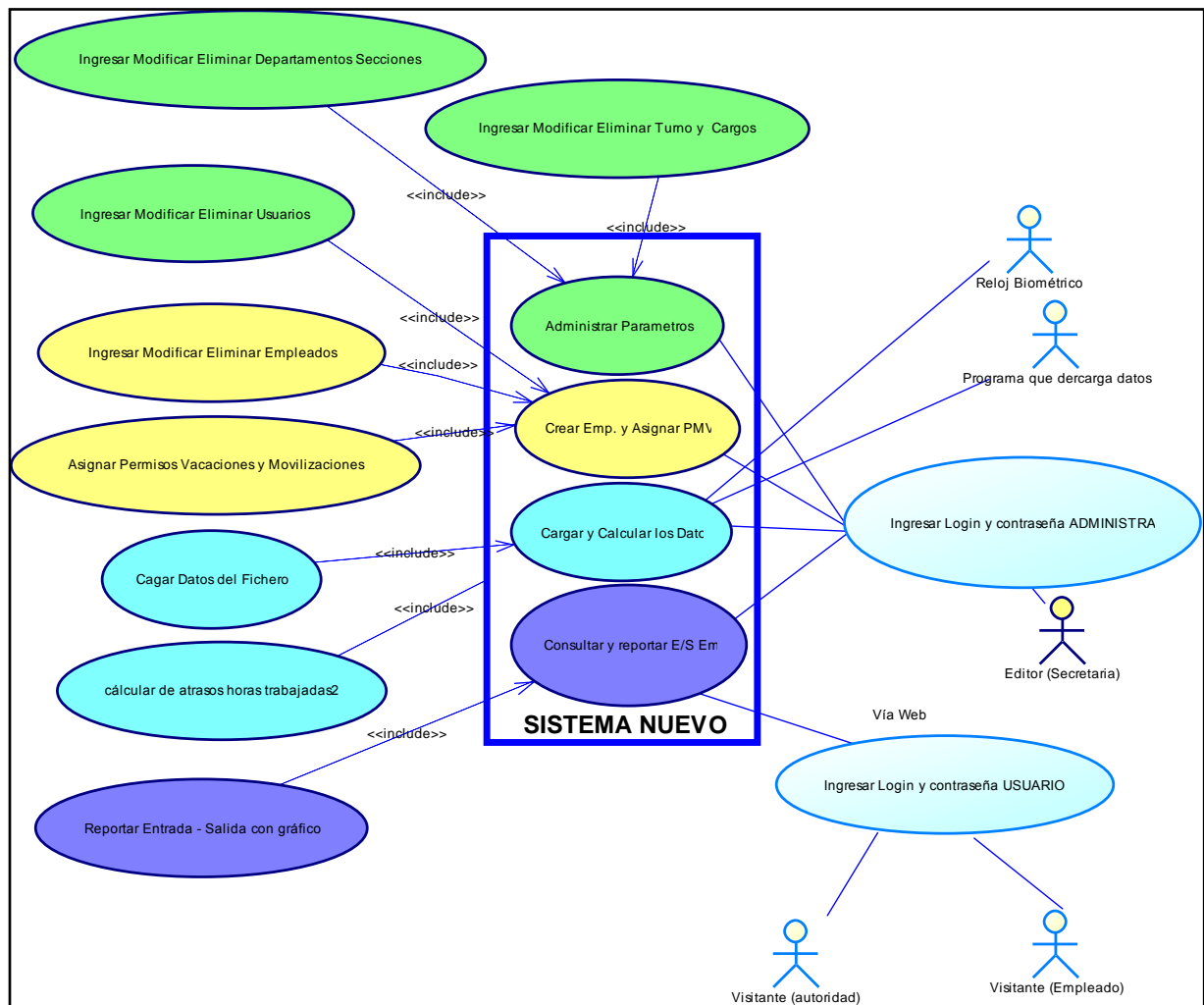


Fuente: Propia.

Figura 1.11. Modelo de casos de uso Seguridad y Accesos.

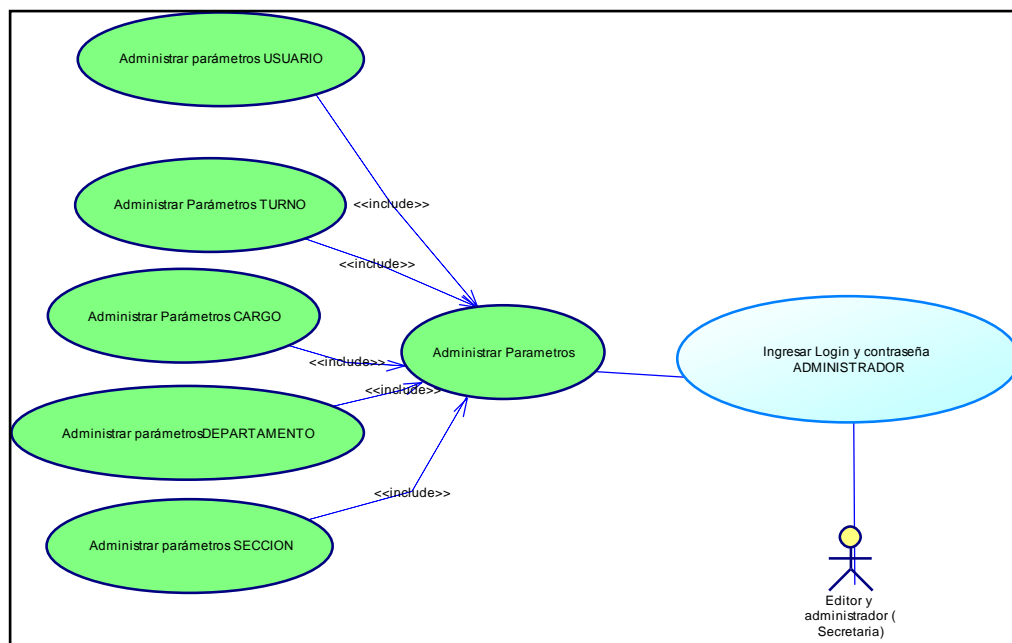
El caso de uso describe el proceso de ingreso por medio de un usuario y contraseña para consultas y administración del sistema vía web. Hay un tiempo de espera para que la sesión se cierre en cualquiera de los casos por seguridad, para restablecer deberá ingresar nuevamente.

Identificar Casos de Usos del Sistema Propuesto:



Fuente: Propia.

Figura 1.12. Diagrama de Casos de Uso de Sistema Propuesto global.



Fuente: Propia.

Figura 1.13. Diagrama de Casos de Uso Administrar parámetros.

Descripción de los Casos de Uso del Sistema Propuesto:

1. Ingresar Parámetros:

Descripción Breve:

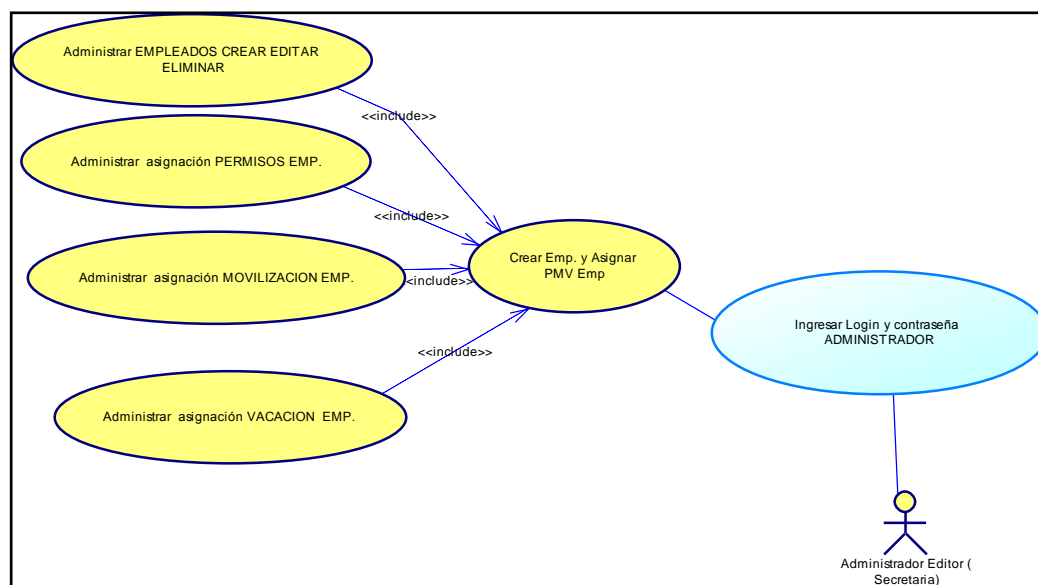
El caso de uso describe el proceso Ingresar Parámetros, para el acoplamiento de los datos del Reloj Biométrico con el Sistema de Control de Asistencia.

Flujo Básico de Eventos:

1. La secretaria Ingresa los parámetros de Turnos, Cargos de los Empleados.
2. Crea los Departamentos que existen en la Institución, con sus respectivas secciones.

Precondiciones:

Ninguna.



Fuente: Propia.

Figura 1.14. Diagrama de Casos de Uso Crear Empleados y Asignar PMV.

2. Asignar:

Descripción Breve:

El caso de uso describe el proceso Crear al empleado y Asignar parámetros, para cada empleado.

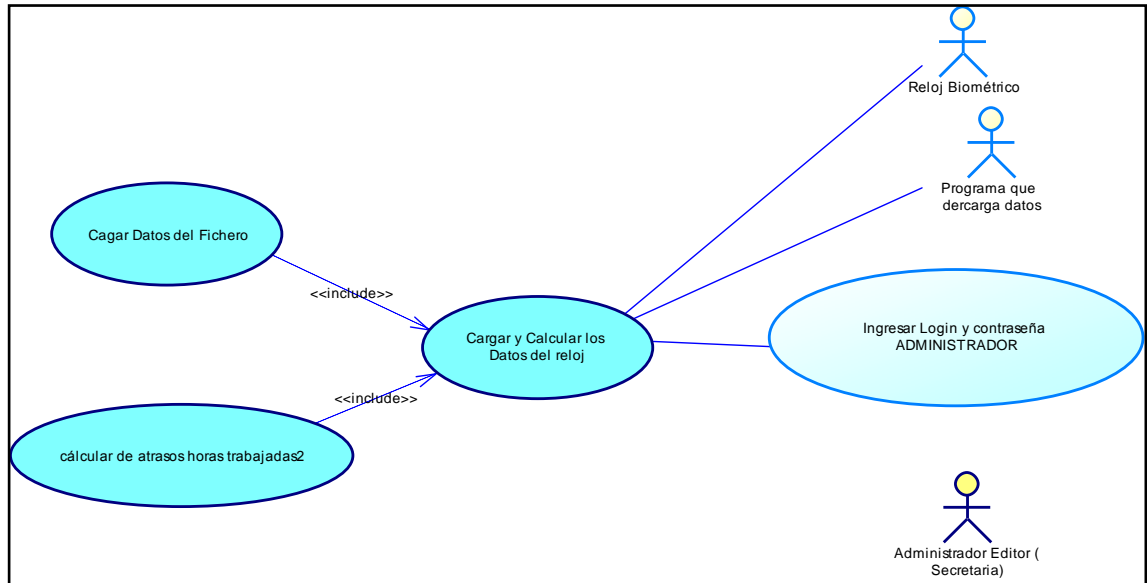
Además se asigna permisos, vacaciones y movilizaciones que tenga el empleado.

Flujo Básico de Eventos:

1. Ingresar a la opción asignar.
2. Crea un nuevo empleado, asignado los parámetros correspondientes.
3. Graba los cambios.
4. De ser el caso puede asignar permisos, vacaciones y movilizaciones al empleado.

Precondiciones:

1. Tener ingresado todos los parámetros de turno, cargo, departamentos y secciones.



Fuente: Propia.

Figura 1.15. Diagrama de Casos de Uso Cargar y calcular los datos del reloj.

3. Cargar Datos:

Descripción Breve:

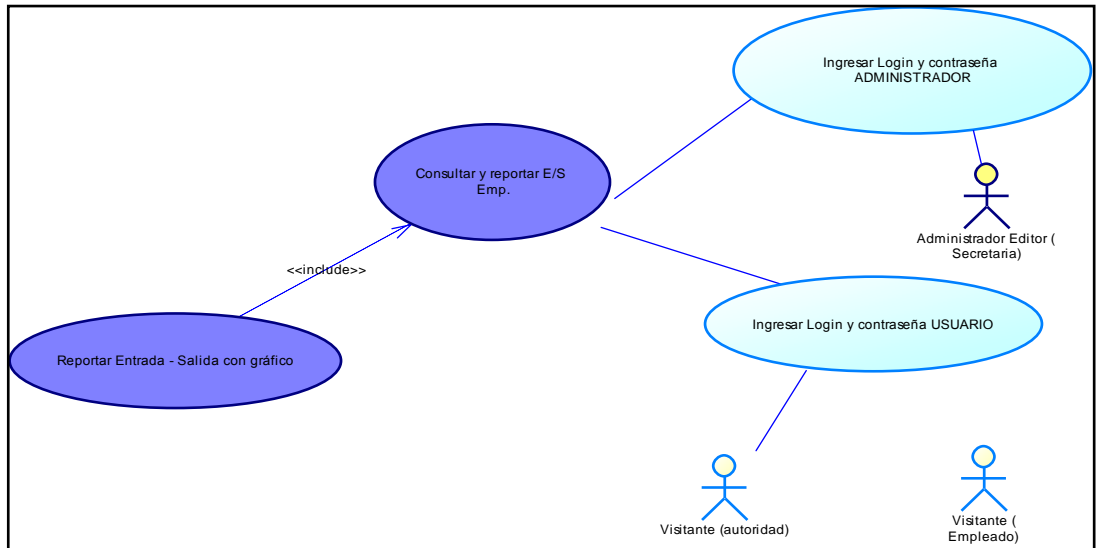
El caso de uso describe el proceso de carga de datos y cálculo de atrasos, horas extras y horas trabajadas de los datos generados por el reloj biométrico.

Flujo Básico de Eventos:

1. Ingresa a la opción cargar.
2. Buscar el archivo generado por el reloj que se encuentra en el disco duro del servidor.
3. Clic Aceptar (debe salir un mensaje de datos cargados correctamente).
4. Luego, clic en calcular para que procese la información

Precondiciones:

1. Tener ingresado todos los parámetros de turno, cargo, departamentos y secciones y creado los empleados con el mismo código del reloj.



Fuente: Propia.

Figura 1.16. Diagrama de Casos de Uso Consultar y reportar E/S Emp.

3. Consultar y reportar datos:

Descripción Breve:

El caso de uso describe el proceso de consulta y reporte de los resultados de los cálculos realizados con los datos generados por el reloj biométrico.

Flujo Básico de Eventos:

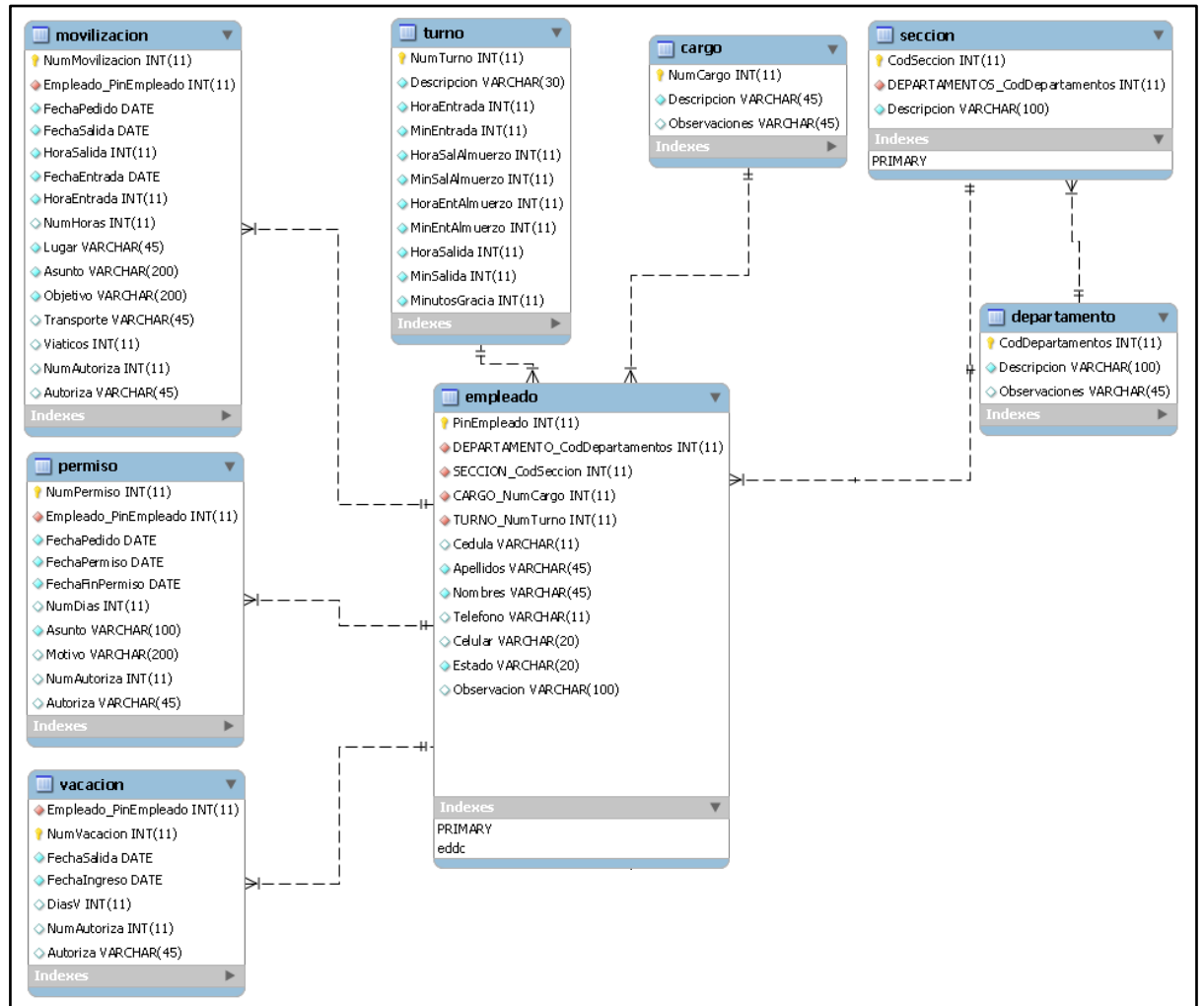
1. Escoger la opción búsqueda.
2. Ingresar la fecha inicial y final a consultar.
3. Ingresar el código similar en inicio y fin si es un solo empleado.
4. Ingresar el código final si es un rango de empleados.
5. Ingresar el número de registros a mostrar en pantalla (por defecto son 24 registros los que se ven).
6. Escoger la opción Buscar
7. Si es necesario generar .pdf clic en el botón asignado

Precondiciones:

1. Tener ingresado todos los parámetros de turno, cargo, departamentos y secciones, además creado los empleados con el mismo código del reloj y cargado los datos generados por el reloj.

1.4.2. Requisitos de datos.

1.4.2.1. Modelo Entidad-Relación.



Fuente: Propia.

Figura 1.17. Diagrama Entidad Relación de la base de datos requerida.

1.4.3. Requisitos de Hardware y Software.

Hardware:

Los requisitos para que funcione el sistema son:

1. Reloj Biométrico de Huella Digital/Clave: La institución tiene uno con las siguientes características:

- Marca: Waldso.
- Modelo: WU-5.
- Capacidad de usuarios: 5.000 usuarios
- Capacidad de Transacciones: 60.000.
- Comunicación: USB y RS232.



Fuente: Manual Instalación Waldso.

Figura 1.18. Imagen Reloj Biométrico de la Institución

2. Cable UTP 5 metros (X2) y 2 conectores seriales macho-hembra (X2).



Fuente:

https://www.paratupc.es/productos/small/NILOX/694205_NILOX_07NXR903MF201_cable_serial_RS232_9.jpg

Figura 1.19. Cable y conectores para construcción de cable para comunicación reloj biométrico a computador.

3. Hosting para almacenar datos y sistema mínimos.

- Hosting plus

Panel de control para manejo de la cuenta: CPanel

Espacio en Disco: 500MB

Bases de Datos MySQL / PostgreSQL: 5

Administrador de Bases de Datos en línea PhpMyAdmin

PHP 5/ PERL / CGI ok

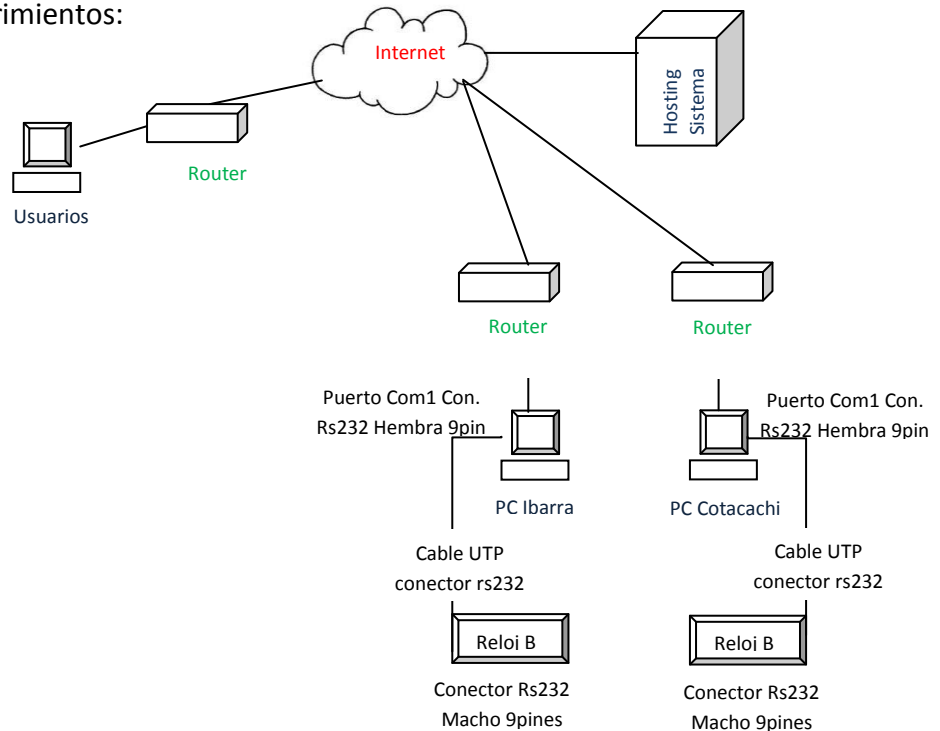
Software:

Los Requisitos en software son:

- Equipo Servidor ó hosting.
 - Sistema Operativo Windows Server 2003 min. ó Suse 9 min.
 - XAMP, elementos activos de la herramienta:
 - Apache.
 - Php 5.
 - MySQL 5.
 - Framework Symfony V. 1.4.
- Equipo para desarrollo.
 - Sistema operativo Windows.
 - Navegadores actuales.
 - NetBeans V. 6.9.
 - Power Designer V. 15.
 - Symfony V.1.4.
 - MySQL Workbench V. 5.2.
- Equipo de Usuario.
 - Sistema operativo Windows XP.
 - Navegadores actuales (versiones nuevas).

1.4.4. Requisitos de Telecomunicaciones.

Para la Intercomunicación del Sistema y los usuarios se deben tener unos requerimientos:



Fuente: Propia.

Figura 1.20. Infraestructura de telecomunicación.

- Las 2 oficinas Ibarra y Cotacachi deben estar conectados a sus respectivos relojes Biométricos por los puertos seriales (Puerto RS232), ver figura 1.18.
- Las 2 oficinas de Ibarra y Cotacachi deben poseer Internet para poder cargar los datos de sus respectivos relojes hacia el servidor, no es necesario tener una gran velocidad o ancho de banda para esta acción. ver figura 1.18.
- Los empleados y autoridades para consulta deben poseer internet en sus equipos. Preferentemente el servicio de internet con banda ancha ADSL, ver figura 1.18.

Capítulo II

Elaboración:

Diseño de los Procesos y Funciones del Sistema Web.

Entender cómo se va a construir.

2.1. Descripción detallada del sistema.

2.1.1. Programas, módulos reutilizables y Objetos.

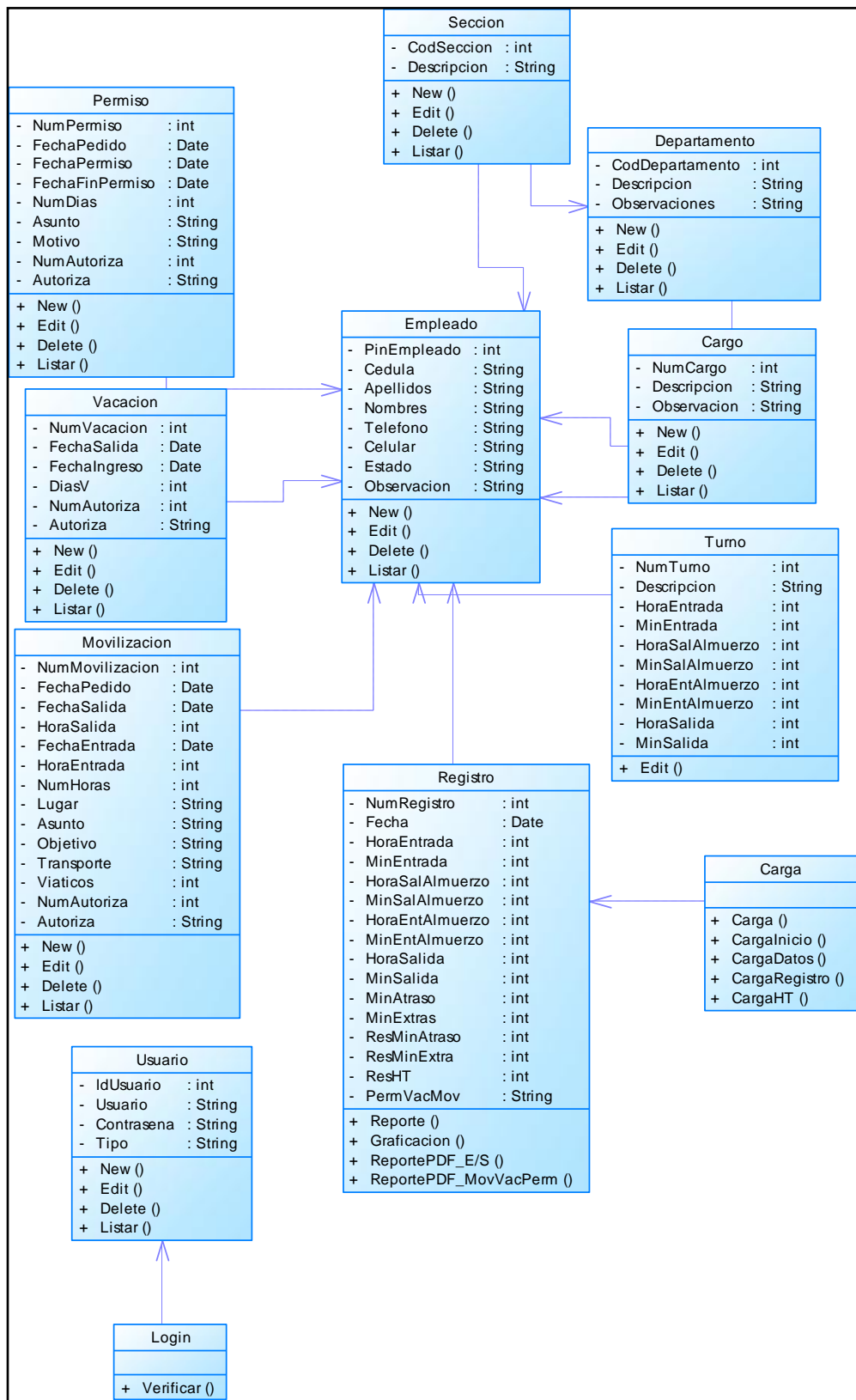
En esta etapa se explica el software que se va a utilizar para desarrollar este sistema.

	Herramientas	¿Por qué?
<u>Base de Datos:</u>	MySQL	Escalabilidad y flexibilidad, Alto rendimiento, Alta disponibilidad. Robusto soporte transaccional. Fortalezas en Web. Fuerte protección de datos.
<u>Plataforma</u>	PHP	La Comunidad PHP, rendimiento, portabilidad, etc.
<u>IDE</u>	NetBeans 6.9	Entorno de desarrollo modular, compatible total con PHP.
<u>Servidor Aplicación</u>	Apache	Es robusto y compatible con PHP y NetBeans.
<u>Framework:</u>	<u>Symfony</u>	Symfony es maduro, estable, profesional y está muy bien documentado. Compatible con NetBeans 6.8.

Fuente: Propia.

Tabla 2.1. Herramientas a utilizar en el desarrollo

Luego se presenta un diagrama de clases para mostrar las clases, atributos y relaciones entre ellas. A continuación se detalla el diagrama:



Fuente: Propia.

Figura 2.1. Diagrama de clases del sistema.

2.1.1.1. Prototipo de Interfaz de usuario.

A continuación se describe un prototipo de la interfaz:

Formulario de Seguridad:

Deberá contener:

- Campo ingreso de nombre de usuario.
- Campo contraseña del usuario o administrador.



Fuente: Propia.

Figura 2.2. Prototipo de Formulario de Seguridad.

Formulario para Usuarios:

Solo usuarios registrados tendrán un usuario y contraseña para acceder al sistema.

Deberá contener:

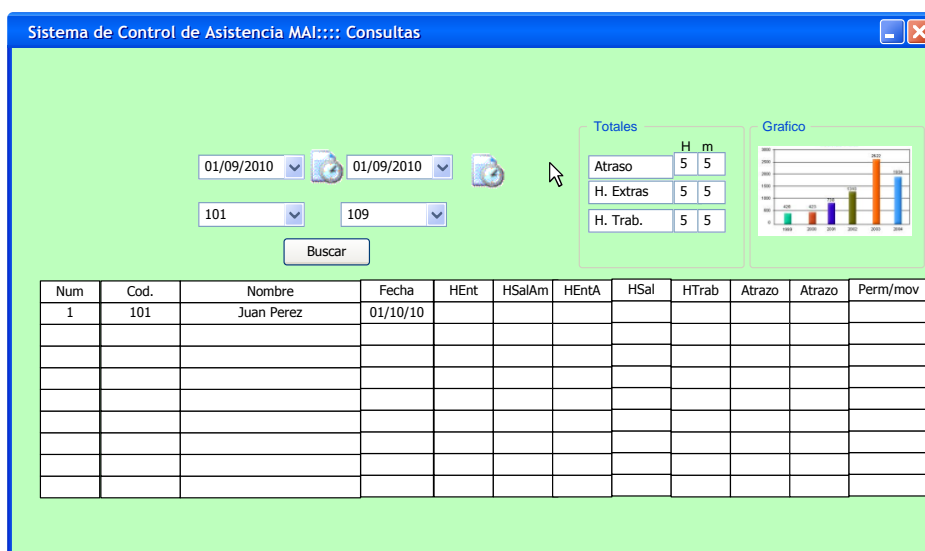
Parámetros requeridos para consulta:

- Fecha Inicial Consulta.
- Fecha Final Consulta.
- Cod. Empleado.

Resultado de Consulta:

- Núm. Reg.
- Código.
- Nombre.

- Fecha.
- Hora y min Ingreso.
- Hora y min Salida Almuerzo.
- Hora y min Ingreso Almuerzo.
- Hora y min Salida.
- Atrasos Minutos y horas.
- Extras Minutos y horas.
- Trabajadas Minutos y horas.
- Si ha tenido permiso, movilización y vacación el empleado.



Fuente: Propia.
 Figura 2.3. Prototipo de Formulario de Usuario.

Formulario para Administradores:

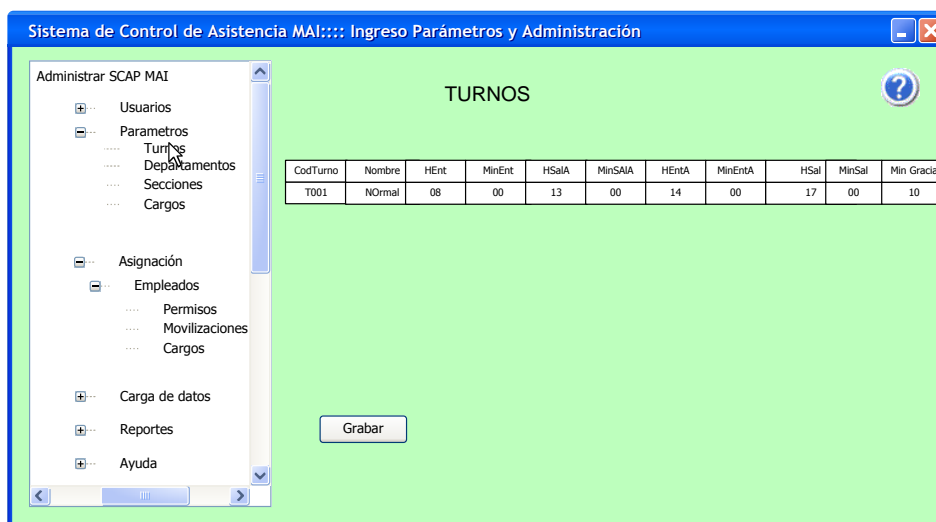
PARÁMETROS.

Formulario Administrar para administrar parámetros del Usuario.

- Usuario (Nombre de usuario en este caso será su pin).
- Contraseña.
- Tipo (Empleado/Administrador).

Formulario Administrar para administrar parámetros del turno.

- Descripción (Normal ó Especial).
- Hora Entrada al Trabajo.
- Hora Salida al Almuerzo.
- Hora Entrada del almuerzo.
- Hora Salida Trabajo.
- Minutos Gracia para entrar al trabajo, para no tomar como atraso.

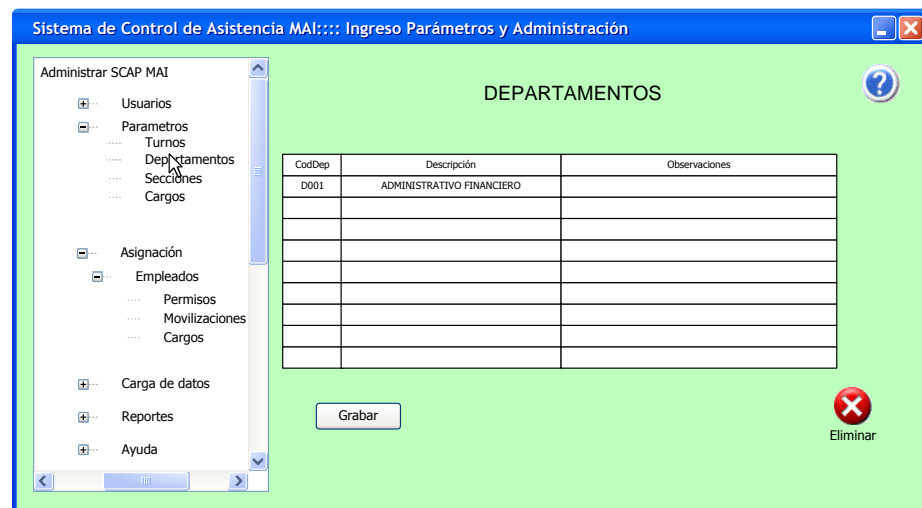


Fuente: Propia.

Figura 2.4. Prototipo de Formulario de parámetros (Turno).

Formulario para administrar el parámetro departamento.

- Descripción (Nombre del Departamento).
- Observaciones (Si las Hubiere).

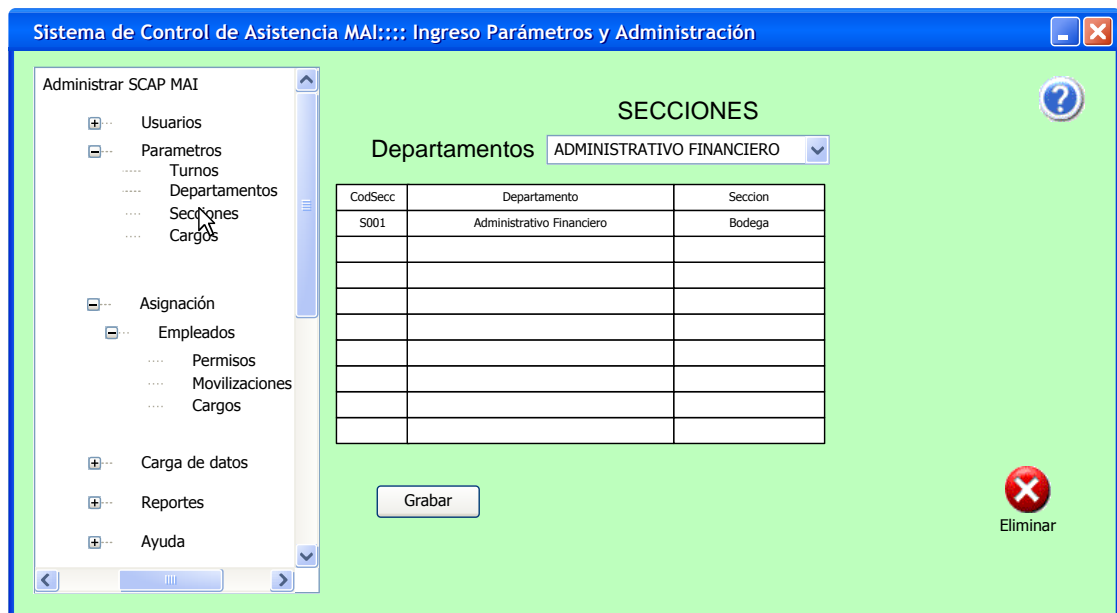


Fuente: Propia.

Figura 2.5. Prototipo de Formulario de parámetros (Departamentos).

Formulario para administrar el parámetro secciones de los departamentos

- Departamento (Es importante haber creado antes este parámetro)
- Descripción (Nombre de la sección).



Fuente: Propia.

Figura 2.6. Prototipo de Formulario de parámetros (Secciones).

Formulario para administrar el parámetro cargos de los empleados

- Descripción (Nombre del cargo)
- Observaciones (Si las Hubiere).

The screenshot shows a web application window titled 'Sistema de Control de Asistencia MAI: Ingreso Parámetros y Administración'. The main content area is green and titled 'CARGOS'. On the left, there is a navigation menu with options like 'Usuarios', 'Parametros', 'Turnos', 'Departamentos', 'Secciones', 'Cargos', 'Asignación', 'Empleados', 'Permisos', 'Movilizaciones', 'Cargos', 'Carga de datos', 'Reportes', and 'Ayuda'. The 'Cargos' option is selected. The main area contains a table with two columns: 'CodCargo' and 'Nombre Cargo'. The first row has 'C001' and 'SERVIDOR PUBLICO 1'. Below the table are buttons for 'Grabar' and 'Eliminar'.

CodCargo	Nombre Cargo
C001	SERVIDOR PUBLICO 1

Fuente: Propia.

Figura 2.7. Prototipo de Formulario de parámetros (Cargos).

ASIGNACIÓN.

Formulario Administrar empleado.

Deberá contener los siguientes campos para crear al empleado:

- Pin empleado (Código único de cada empleado creado en el reloj biométrico).
- Departamento.
- Sección.
- Cargo.
- Turno.
- Cédula.
- Apellidos.

- Nombres.
- Teléfono.
- Celular.
- Estado (Activo /Pasivo).
- Observación.

Cod	Depart	Ciudad	Cedula	Nombre	Apellido	Telf	Cel	Estado	Observación
101	Imbabura	Ibarra	100242285	Juan	Perez	2612555	092225	Casado	

Fuente: Propia.

Figura 2.8. Prototipo de Formulario de Asignación (Empleados).

Formulario para Asignar Permisos.

- Empleado pin empleado.
- Fecha pedido.
- Fecha permiso.
- Fecha fin permiso.
- Asunto.
- Motivo.
- Num autoriza.
- Autoriza.

Administración SCAP MAI

- Usuarios
- Parametros
 - Turnos
 - Departamentos
 - Secciones
 - Cargos
- Asignación
 - Empleados
 - Permisos
 - Movilizaciones
 - Vacaciones
- Carga de datos
- Reportes
- Ayuda

Asignar

Empleado: Juan Pérez

Fecha Salida: 17/09/2010

Fecha Entrada: 19/09/2010

Asunto: Calamidad Domestica

Detalle Asunto	Num Autoriza	Autoriza	Observación
Enfermedad a la cabeza	092225	Ing. Fuentes	

Grabar

Eliminar

Fuente: Propia.

Figura 2.9. Prototipo de Formulario de Asignación (Permisos).

Formulario para Asignar Movilizaciones.

- Empleado pin empleado.
- Fecha pedido.
- Fecha salida.
- Hora salida.
- Fecha entrada.
- Hora entrada.
- Lugar.
- Asunto.
- Objetivo.
- Transporte.
- Viáticos.
- Num autoriza.
- Autoriza.

Sistema de Control de Asistencia MAI::: Asignación Movilizaciones

Administrar SCAP MAI

- Usuarios
- Parametros
 - Turnos
 - Departamentos
 - Secciones
 - Cargos
- Asignación
 - Empleados
 - Permisos
 - Movilizaciones
 - Vacaciones
- Carga de datos
- Reportes
- Ayuda

Asignar

Empleado: Juan Pérez

Fecha Salida: 17/09/2010

Fecha Entrada: 19/09/2010

Lugar: Cotacahi

Objetivo	Detalle Asunto	Transporte	Viáticos	Num Autoriza	Autoriza
Inspección	Inspeccionar la reserva	Terrestre	10,00	76567	Ing. Fuentes

Grabar

Eliminar

Fuente: Propia.

Figura 2.10. Prototipo de Formulario de Asignación (Movilizaciones).

Formulario para Asignar Vacaciones.

- Empleado pin empleado.
- Fecha salida.
- Fecha ingreso.
- Num autoriza.
- Autoriza.

Sistema de Control de Asistencia MAI::: Asignación Vacaciones

Administrar SCAP MAI

- Usuarios
- Parametros
 - Turnos
 - Departamentos
 - Secciones
 - Cargos
- Asignación
 - Empleados
 - Permisos
 - Movilizaciones
 - Vacaciones
- Carga de datos
- Reportes
- Ayuda

Asignar

Empleado: Juan Pérez

Fecha Salida: 17/09/2010

Fecha Entrada: 19/09/2010

Num Autoriza	Autoriza	Observación
092225	Ing. Fuentes	

Grabar

Eliminar

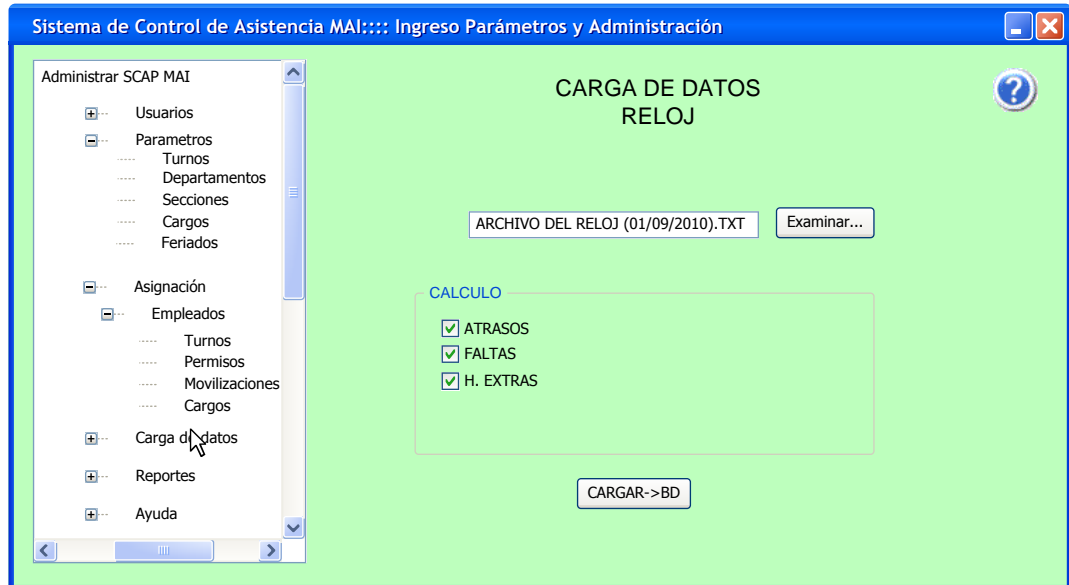
Fuente: Propia.

Figura 2.11. Prototipo de Formulario de Asignación (Vacaciones).

Formulario para cargar datos del reloj al sistema.

CARGAR.

- Cargar Buscar archivo.
- Calcular, será necesario tener un rango de fechas (inicio/fin), para que optimice el tiempo de cálculo.



Fuente: Propia.

Figura 2.12. Prototipo de Formulario de Carga de datos.

Formulario para Consultar datos.

CONSULTAS.

Parámetros requeridos para consulta:

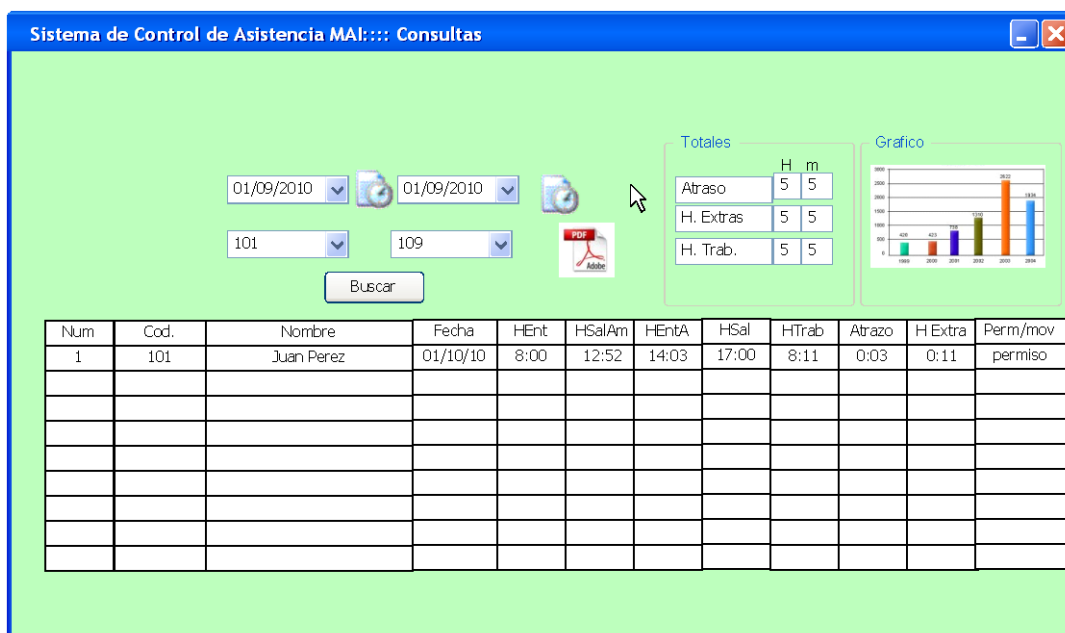
- Fecha Inicial Consulta.
- Fecha Final Consulta.
- Cod. Empleado (inicio búsqueda).
- Cod. Final Empleado (fin búsqueda).

Resultado de consulta:

- Núm. Reg.
- Código.
- Nombre.

- Fecha.
- Hora y min Ingreso.
- Hora y min Salida Almuerzo.
- Hora y min Ingreso Almuerzo.
- Hora y min Salida.
- Atrasos Minutos y horas.
- Extras Minutos y horas.
- Trabajadas Minutos y horas.
- Si ha tenido permiso, movilización y vacación el empleado.

Tendrá un Gráfico estadístico y botón para exportar a PDF adicionales.

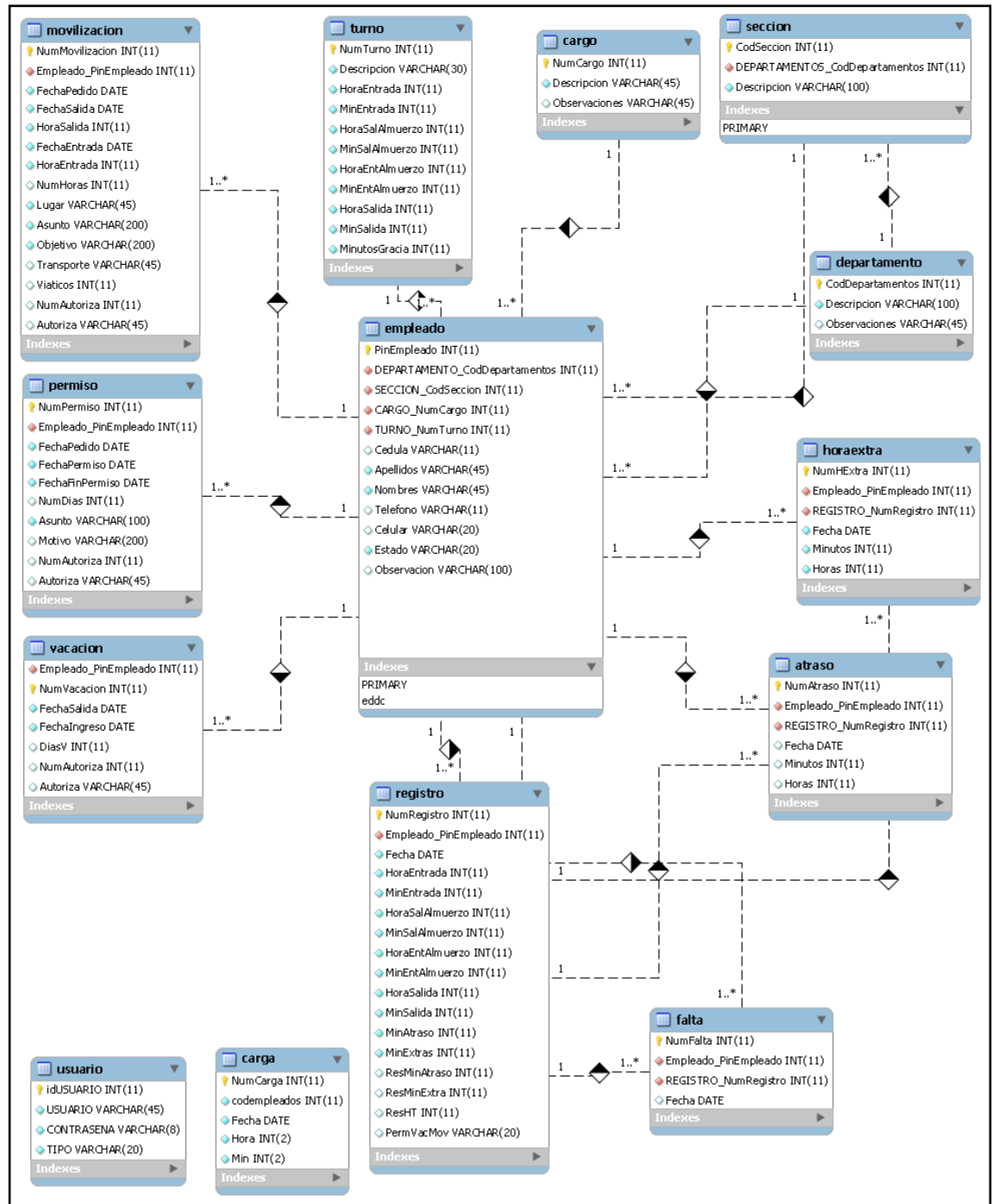


Fuente: Propia.

Figura 2.13. Prototipo de Formulario de Consulta.

2.1.2. Modelo Relacional.

2.1.2.1. Ficheros y Base de datos.



Fuente: Propia.

Figura 2.14. Diagrama Entidad Relación.

2.1.2.2. Diccionario de datos.

En el Anexo Núm. 1 se describen las tablas y tipos de datos que la componen además de una descripción de los datos que se almacenarán en ésta.

2.1.2.3. Carga de sistema y tiempos de respuesta.

La carga de los datos al sistema se lo realiza cada mes, entonces, el flujo de datos dependiendo de los empleados es aproximadamente de 80 a 100 registros por empleado.

En la situación actual se tendría:

	Número de empleados.	Número de Registros mensuales generados por cada empleado	Número de registros que se generan en la base de datos por mes	Tiempo de demora ideal
Sistema Actual	24	4 timbras X 20 días. Aprox. 80 c/u.	1920 registros	10 a 20 minutos
Sistema propuesto	24	4 timbras X 20 días. Aprox. 80 c/u.	1920 registros	< 10 minutos

Fuente: Propia.

Tabla 2.2. Carga y tiempos de respuesta del sistema.

2.1.2.4. Interfaces, tanto humanas como de máquinas.

Humanas:

- Gama de colores acorde a la institución, el verde predomina.
- La estética debe ser simple pero clara y de fácil manejo.
- Como las consultas se realizan por fechas debe tener una herramienta que lo facilite, fecha desplegable.
- Los gráficos estadísticos deben ser legibles, de tamaño apropiado para su análisis.
- La generación de pdf debe ser rápida y con colores que demuestren alguna observación.

Máquinas:

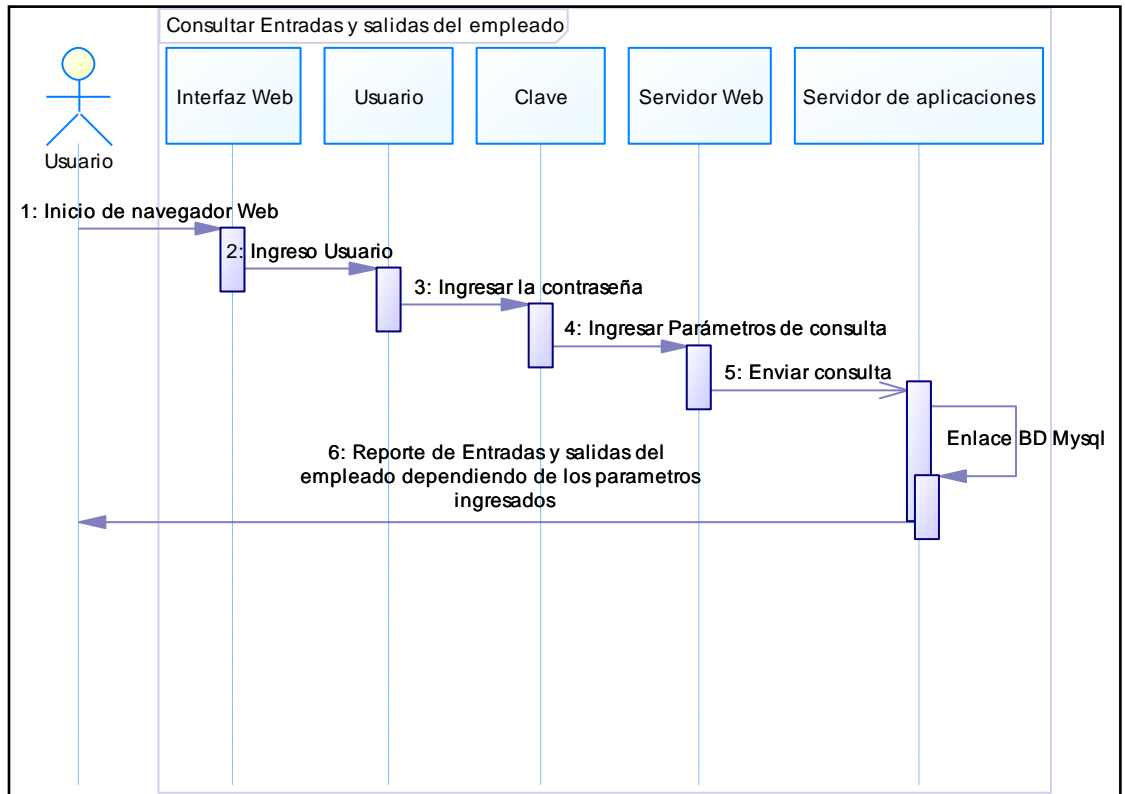
- La interfaz para interactuar con el usuario serán el mouse y teclado.

2.2. Descripción de los controles del sistema propuesto.

2.2.1. Modelo de Implementación.

Los esquemas para ver los detalles de los objetos y clases para la ejecución de la implementación se los representa con los diagramas de secuencia, que a continuación se detallan:

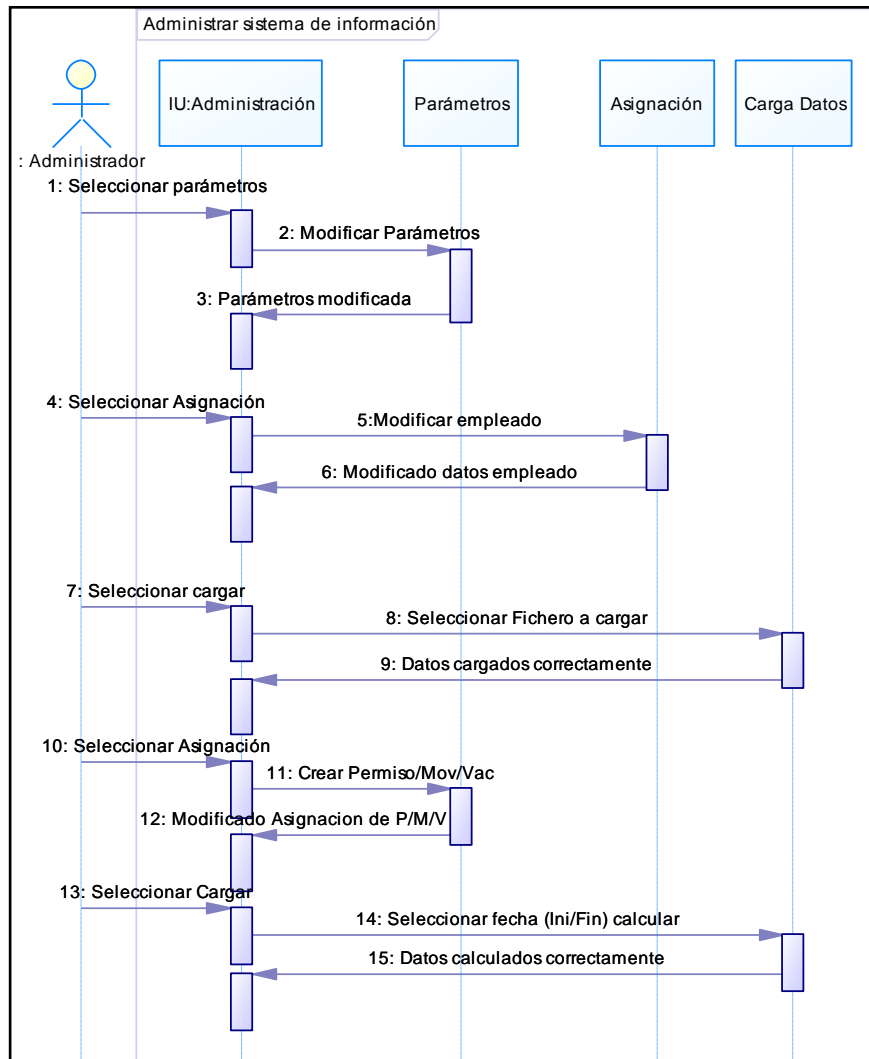
Diagramas de secuencia para Usuario registrado:



Fuente: Propia.

Figura 2.15. Diagrama de secuencia para Consultar Entradas y Salidas de empleado.

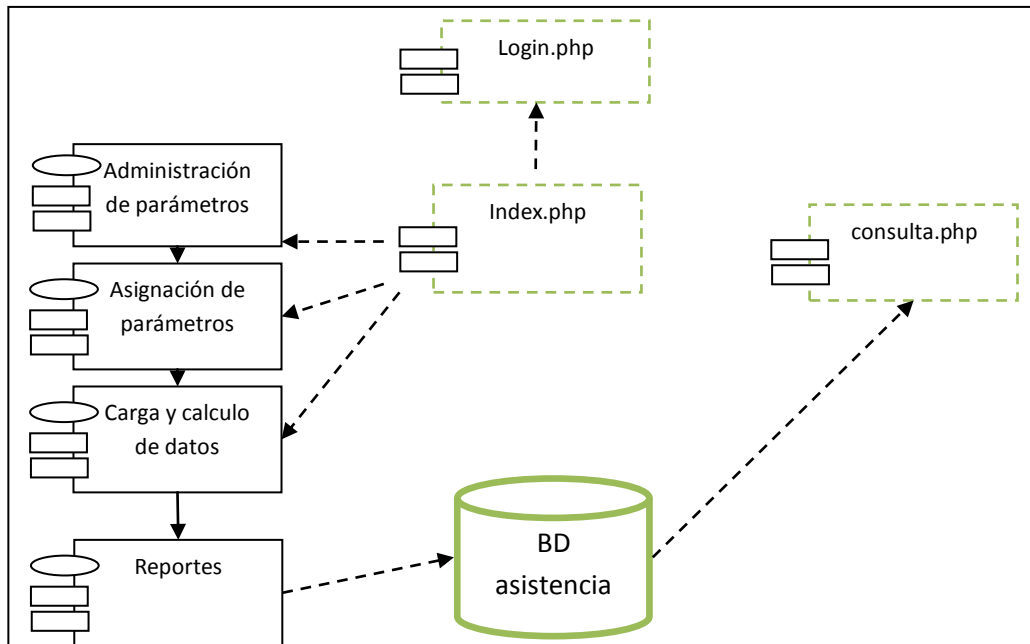
Diagramas de secuencia para Administración del sistema:



Fuente: Propia.

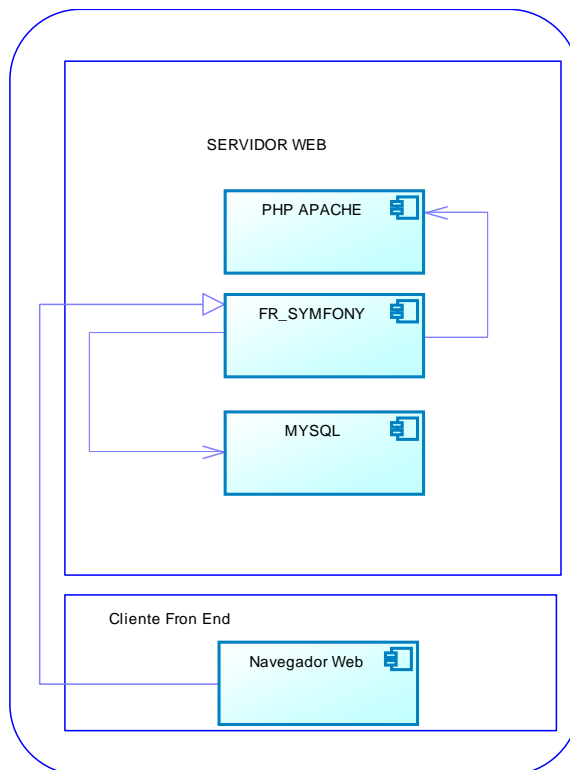
Figura 2.16. Diagrama de secuencia para la administración del sistema.

La implementación del sistema tendrá que contener los siguientes módulos de procesos y ficheros:



Fuente: Propia.

Figura 2.17. Diagrama de componentes del sistema.



Fuente: Propia.

Figura 2.18. Diagrama de despliegue del sistema.

2.2.2. Estándares de Programación.

Los Estándares de programación son los siguientes:

- Base de datos

La base de datos tendrá el nombre de “asistencia”, y para todos los enlaces requeridos se la llamará de esa manera.

- Tablas

Todas las tablas generadas tendrán la estructura siguiente:

Nombre de la Tabla: Mayúsculas y Singular, ej.: DEPARTAMENTO

Los Atributos: Sólo la primera letra de cada palabra en mayúscula, sin espacios, ej.: Hora Entrada

Claves Primarias: Sólo la primera letra de cada palabra en mayúscula, sin espacios, ej.: Pin Empleado

Claves Foráneas: La primera palabra representa a la tabla en donde se encuentra, seguido por una raya baja y luego por el nombre del atributo, ej.: DEPARTAMENTO_CodDepartamento.

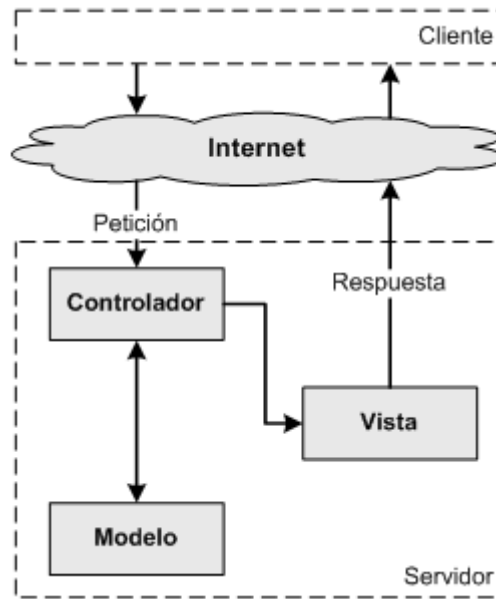
2.2.3. Técnicas de Implementación.

Este sistema realizado en Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por tres niveles:

- El **Modelo** representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La **Vista** transforma el modelo en una página web que permite al usuario interactuar con ella.
- El **Controlador** se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La Figura 2.19. Ilustra el funcionamiento del patrón MVC.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, e-mail, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación.

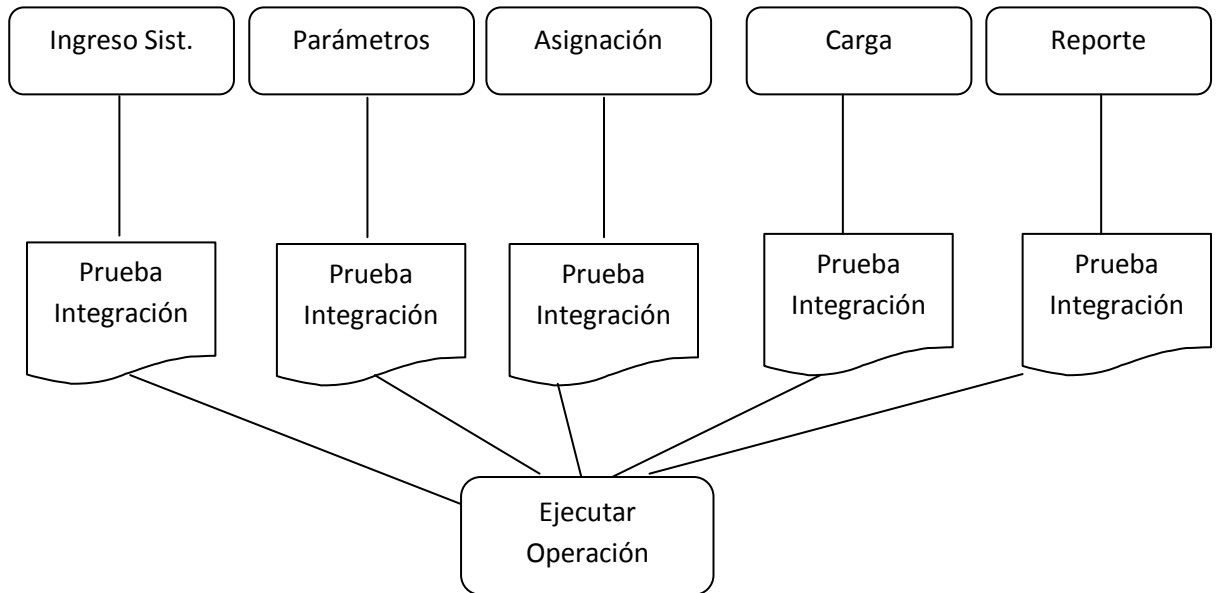


Fuente: Imagen Web⁵

Figura 2.19. Diagrama de secuencia para la administración del sistema (fuente).

2.2.4. Casos de prueba.

Se recomienda realizar pruebas en su ejecución de los siguientes módulos:



Fuente: Propia.

Figura 2.20. Gráfico de casos de prueba propuesto para el sistema.

⁵Nota rápida sobre MVC: <http://lobotuerto.com/blog/2008/04/07/una-nota-rapida-sobre-mvc/>

Capítulo III

Construcción:

Desarrollo e implementación del Sistema Web.

Explicar cómo se construyó el sistema.

3.1. Diseño final del sistema.

El prototipo de la pantalla inicial que puede cambiar conforme se añada requerimientos.



Fuente: Propia.

Figura 3.1. Pantalla del diseño final del sistema etapa Seguridad.



Fuente: Propia.

Figura 3.2. Pantalla del diseño final del sistema etapa Administración.



Fuente: Propia.

Figura 3.3. Pantalla del diseño final del sistema etapa Consulta.

Para dar una idea más clara de los contenidos del sistema a continuación se detalla una tabla representando el mapa del sitio y los accesos que van a tener para los usuarios.

PARÁMETROS	ASIGNACIÓN	CARGAR	CONSULTAS Y REPORTES	AYUDA	Exportar a pdf Entrada/Salidas y Permisos Movilización Vacaciones
Departamento	Empleados	Cargar	Entradas y salidas (Grafico Estadístico)	Manual de usuario	
Sección	Permisos	Calcular			
Turno					
Cargo					
Usuarios					

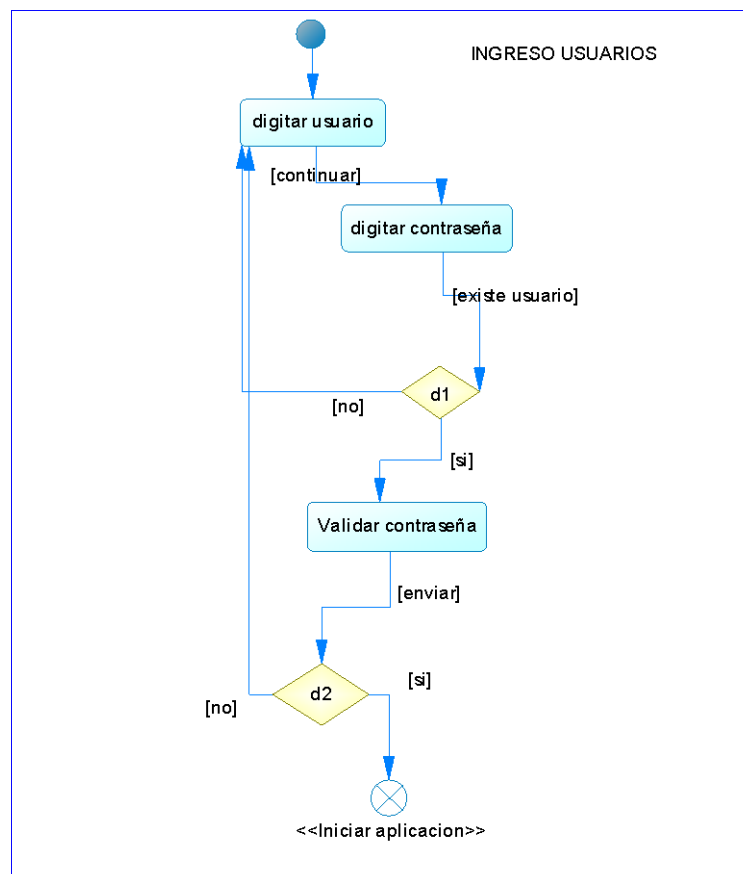
Fuente: Propia.

Tabla 3.1. Representación del mapa del sitio.

3.2. Diagramas definitivos del sistema.

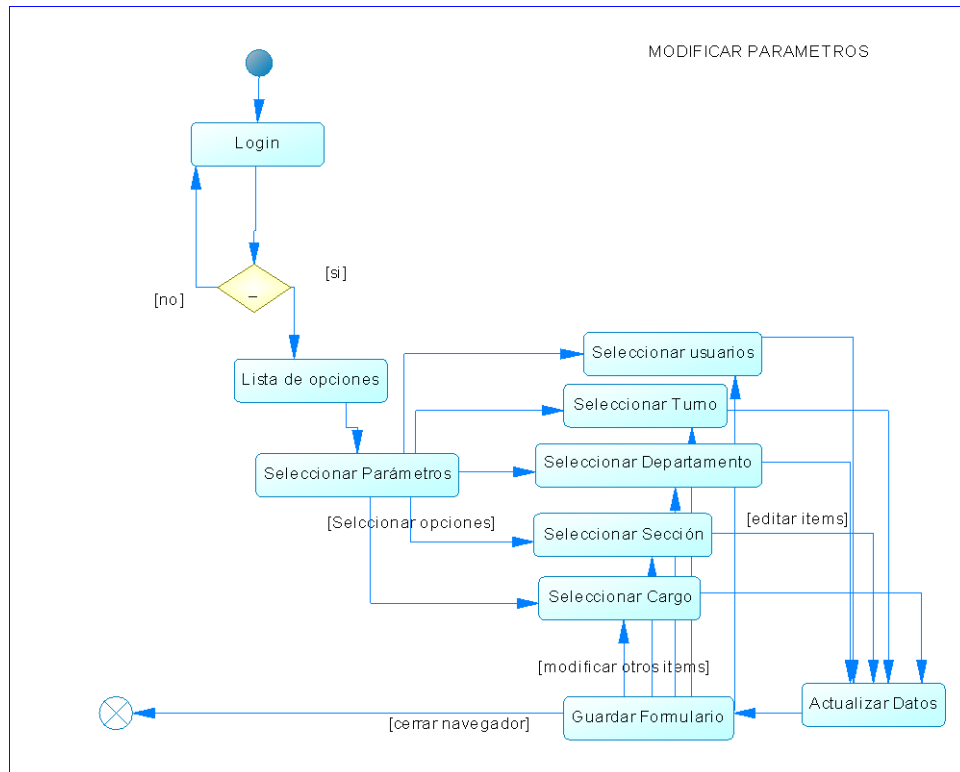
Para entender mejor el diseño del sistema en el manejo de flujo y contenidos, administración y demás secciones.

Diagrama de Actividades PARÁMETROS (ADMINISTRADOR)



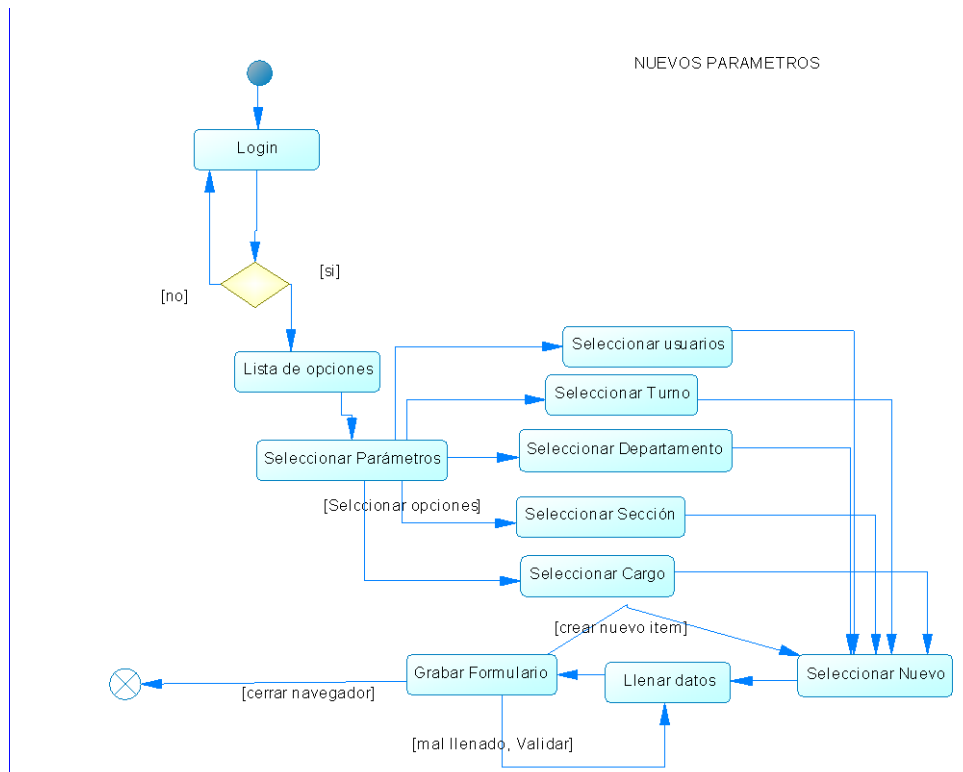
Fuente: Propia.

Figura 3.4. Diagrama de actividades de parámetros modificación.



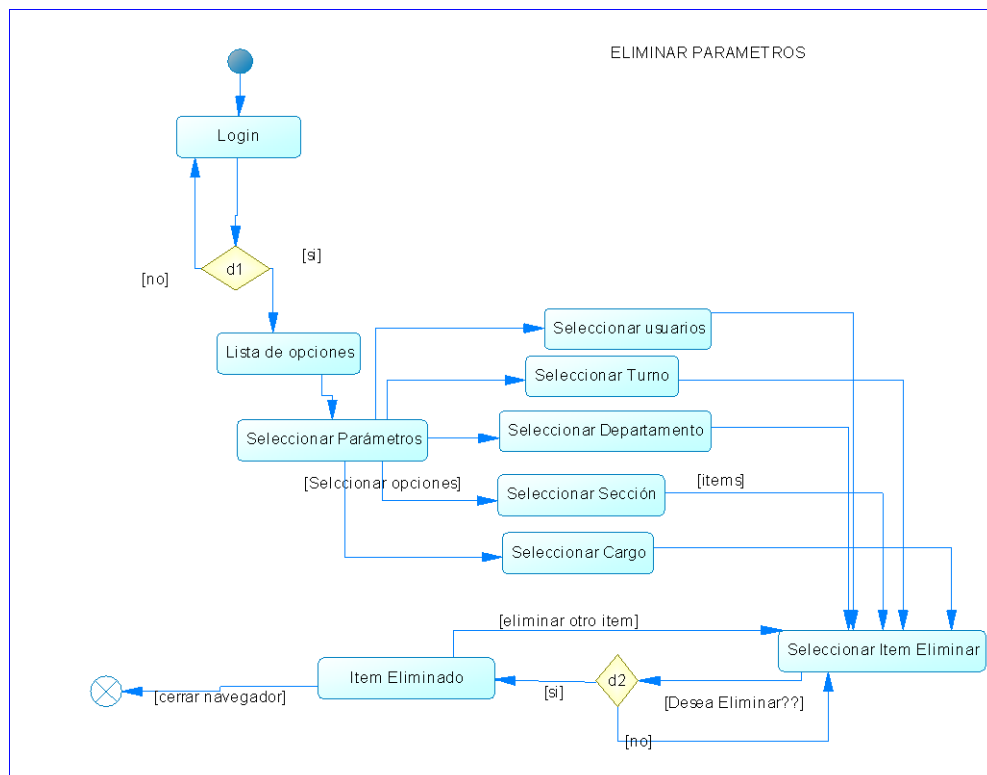
Fuente: Propia.

Figura 3.5. Diagrama de actividades de parámetros modificación.



Fuente: Propia.

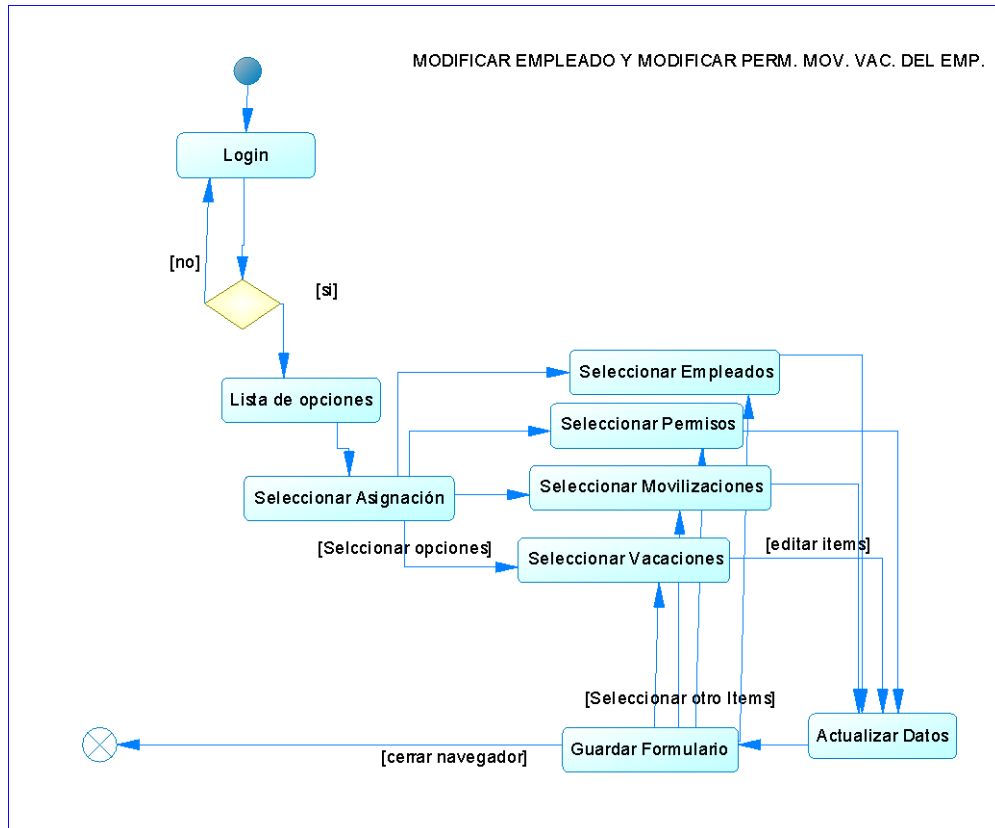
Figura 3.6. Diagrama de actividades de parámetros nuevo.



Fuente: Propia.

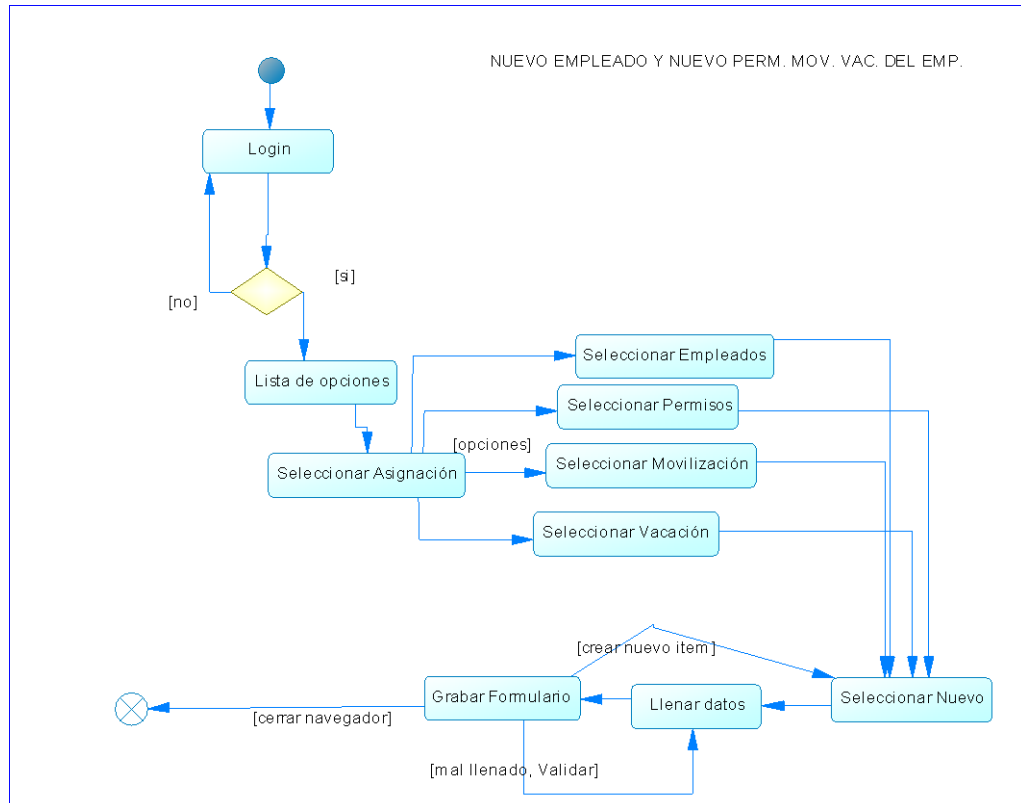
Figura 3.7. Diagrama de actividades de parámetros eliminar.

Diagrama de Actividades ASIGNACIÓN (ADMINISTRADOR)



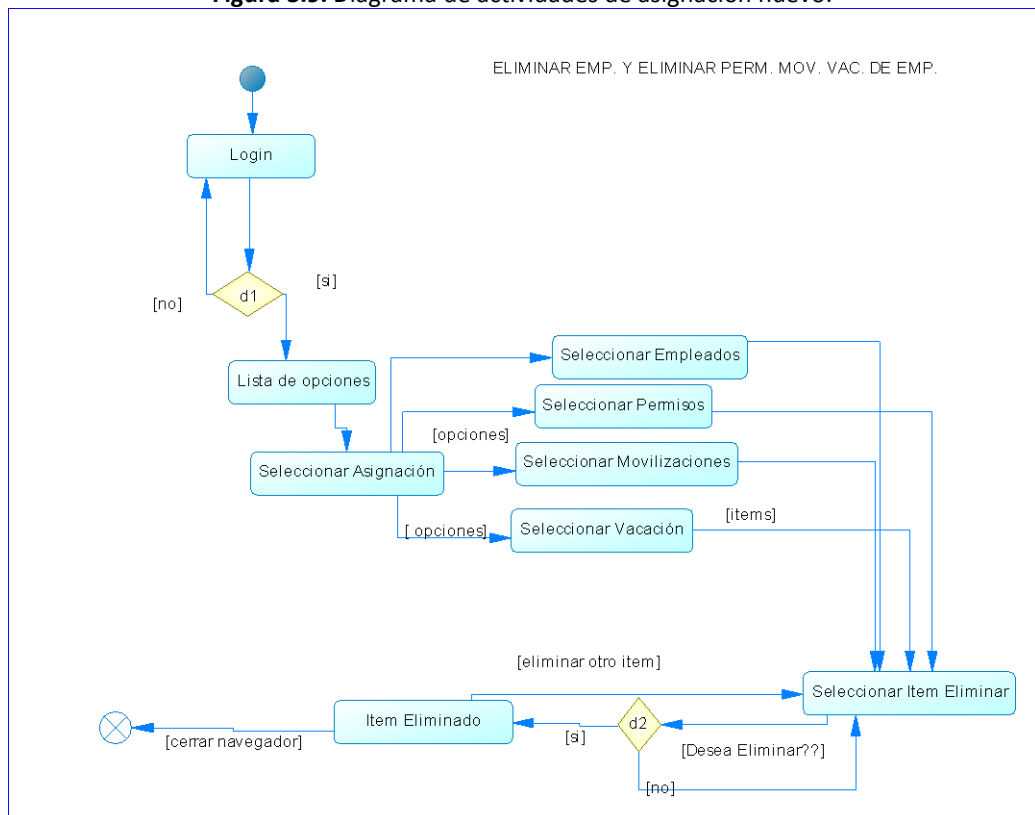
Fuente: Propia.

Figura 3.8. Diagrama de actividades de asignación modificar.



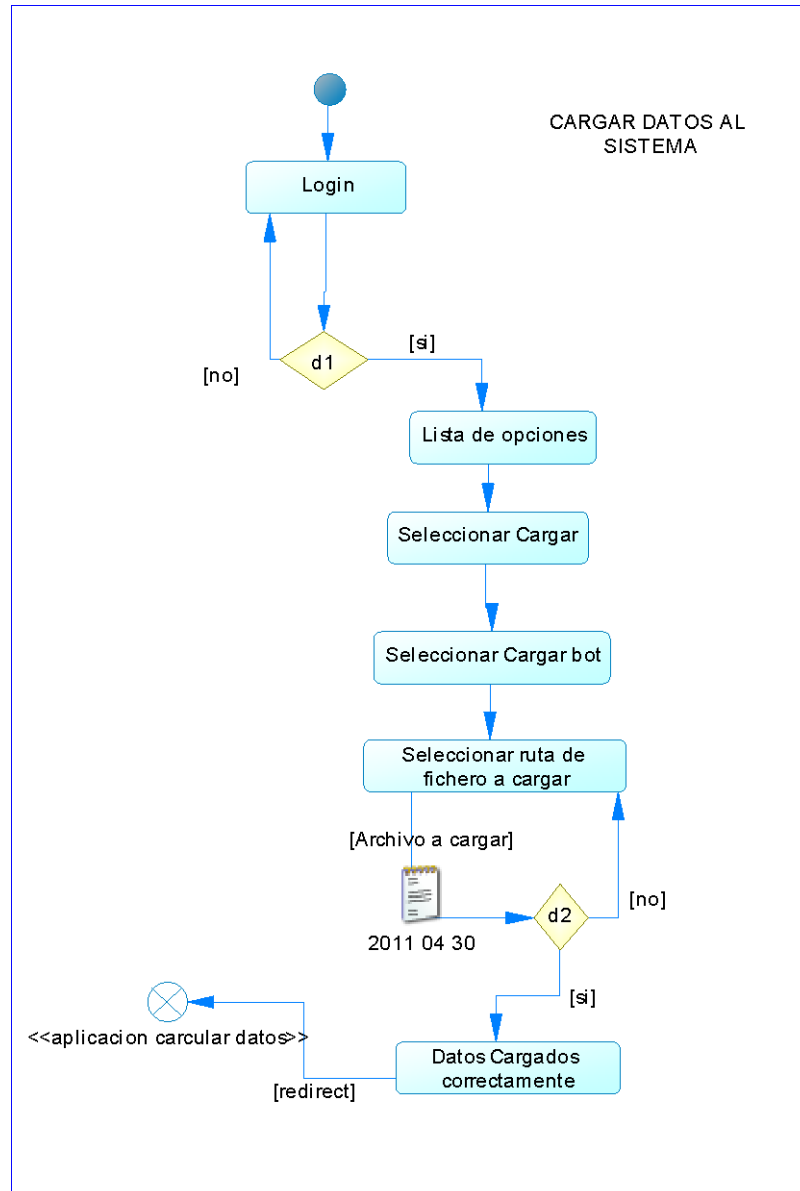
Fuente: Propia.

Figura 3.9. Diagrama de actividades de asignación nuevo.



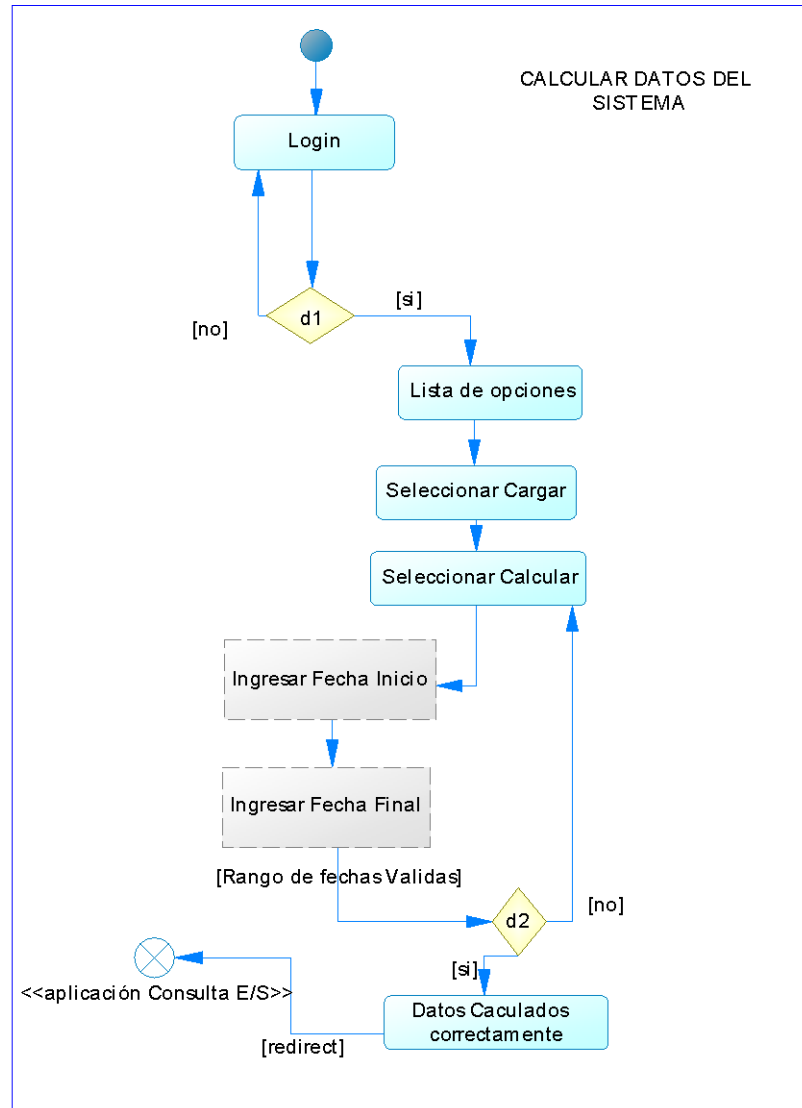
Fuente: Propia.

Figura 3.10. Diagrama de actividades de asignación eliminar.



Fuente: Propia.

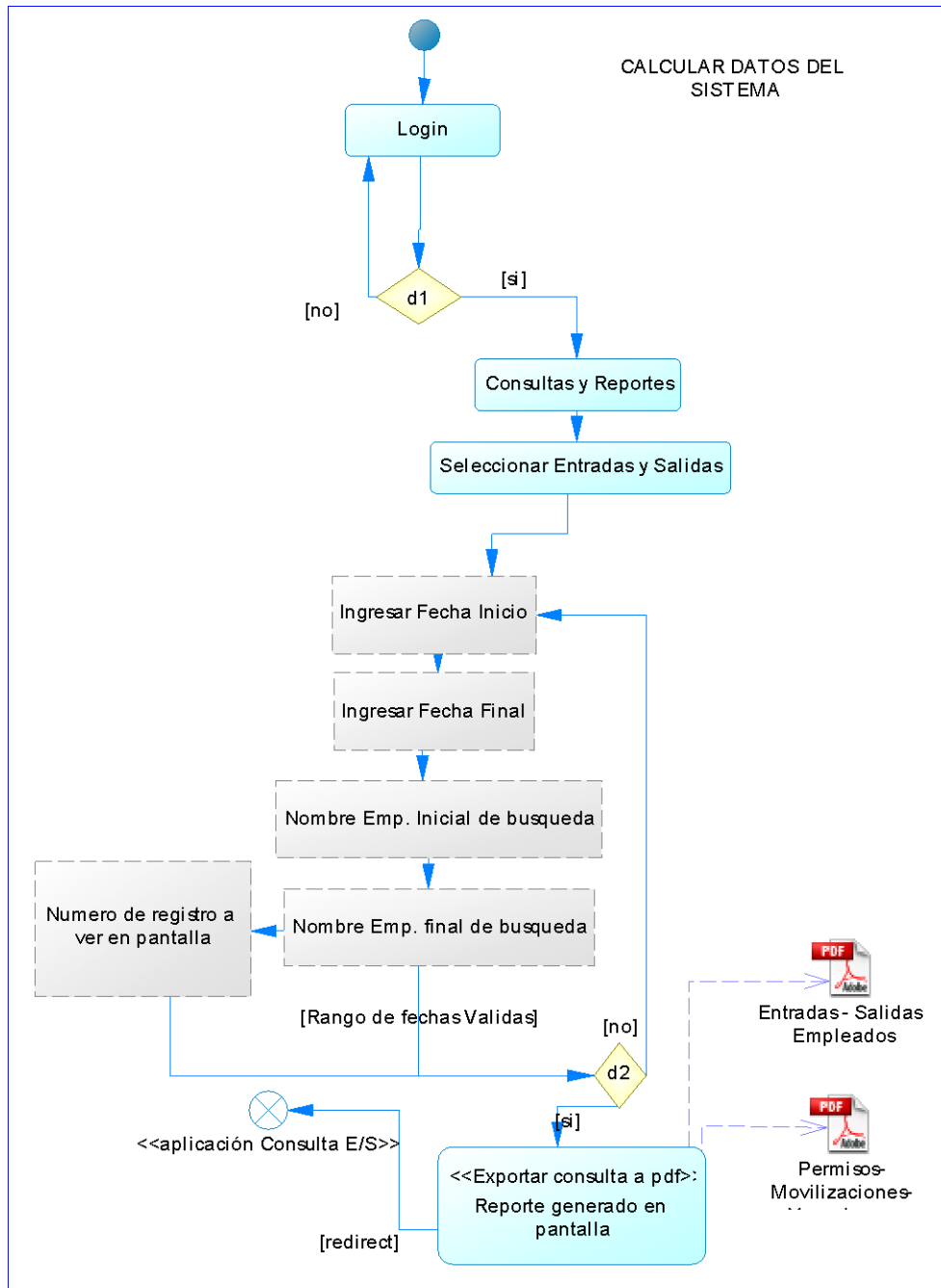
Figura 3.11. Diagrama de actividades Cargar datos al sistema.



Fuente: Propia.

Figura 3.12. Diagrama de actividades Calcular datos al sistema.

Diagrama de Actividades (CONSULTA Y REPORTE).



Fuente: Propia.

Figura 3.13. Diagrama de actividades Reporte de datos.

3.3. Descripción detallada de la lógica del sistema.

En esta etapa se describe la forma en que se basa la construcción de la lógica del sistema, esto se lo demostrará a continuación.

Antes de detallar la lógica es indispensable revisar el anexo Núm. 4 en “Instalación del programa WAD”, que sirve para descargar la información del reloj biométrico.

El anexo Núm. 7 para detallar la construcción del cable de datos para la conexión entre el reloj biométrico y el computador.

Luego para crear, editar y borrar parámetros, empleados, permisos, movilizaciones y vacaciones, se utiliza una forma común para todos estos parámetros que a continuación se detalla:

Al ingresar siempre aparece listado los registros existentes, con un recorrido a la base de datos.

En la parte de creación existe el botón nuevo.

- Los datos serán ingresados en un formulario que contiene campos (input) que alojarán los datos ingresados por el usuario.
- La validación se lo hace mediante funciones de validación que posee el framework Symfony, el componente setValidator ayuda a la validación de todos estos campos, con mensajes personalizados que ayudan a visualizar al usuario los errores cometidos a ingresar alguna información.
- Luego de que los datos se han ingresado correctamente se procede a guardar los cambios con el botón guardar.
- Caso seguido, lista todos los datos ingresados y el botón nuevo aparece en la parte inferior de la lista.

- Para la parte de edición existe un botón en el listado en donde se puede editar.
- Para borrar tiene de igual forma su botón que se encuentra en cada registro listado, para la eliminación como es normal el programa pregunta con un cuadro de dialogo si desea eliminar el registro.

Carga de datos:

Para la carga de fichero se hizo el estudio de un plugin que tiene Symfony que es sfCsvPlugin, con el cual se puede leer archivos separados por comas, el cual fue fundamental para la realización de este sistema.

En la parte de módulo de carga de datos se valida si el fichero está vacío, con la condición: si fichero es igual a nulo. "if (\$archivo <> ")".

Además se compara con los parámetros asignados por el administrador de las horas de entrada y salida de la tabla turno, y los minutos de gracia, para realizar el cálculo de atrasos y horas extras. Con esta condición "if (\$Hora >= \$turn[0]["horaentrada"])", que es la encargada de verificar esto .

Es muy importante esta parte de la revisión de los permisos, vacaciones y movilizaciones que ha tenido el empleado para llenar en la base de datos en las fechas que fueron solicitadas para poder luego reportarlas, esto se lo realiza en la parte del cálculo con una consulta a las tablas pertinentes, Doctrine_Query::create().

Cálculo y Reportes

En esta parte se realiza una consulta a la base de datos en la tabla que realizó los cálculos, es decir, sólo se hace un reporte de la tabla, esto ayuda a que la consulta sea sumamente rápida con esta función "foreach (\$pager-

>getResults() as \$registro)”, que es la encargada de realizar la búsqueda en la tabla registro que es donde se almacenan los cálculos.

La generación de los pdf se lo realizó manualmente personalizando colores y formatos, para que se genere rápido y específicamente con el formato solicitado.

Esto se lo realizó llamando a la clase “new FPDF()”, quien abre un formato para generar un pdf.

Para crear los gráficos se utilizó la herramienta “ezcGraphBarChart” la cual permitió personalizar la graficación de la figura estadística de los datos.

Como en todo momento se utiliza la introducción de fecha, se utilizó un plugin que tiene Symfony para generar calendarios dinámicos, sfjQueryUIPlugin, el cual permite ingresar fechas pero solo en los formularios creados por Symfony, es por eso que, para las consultas y reportes se generó un formulario personalizable de JavaScript llamado calendar.js el cual es de código abierto y se lo pudo modificar para que funcione en la consulta.

Por último, para hacer más fácil la navegación por los registros se utilizó un plugin sfFormExtraPlugin para utilizar la paginación, la cual ayuda a ver al usuario por páginas la información.

3.4. Descripción de entradas y salidas.

En esta parte se detalla la información que entra y sale del sistema por medio de archivos generados y pantallas.

3.4.1. Ficheros.

Para la realización de este sistema se crearon los siguientes archivos:

Configuración de la aplicación:

El archivo routing.yml: Para configuración del ruteo de inicio.

El archivo security.yml: Para habilitar la seguridad del sistema.

El archivo settings.yml: Para señalar el nombre del módulo para el ruteo de inicio.

Configuración del Sistema:

ProjectConfiguration.class: Para cargar los plugins necesarios del sistema.

Databases.yml: Para configurar parámetros de la conexión a la base de datos.

Properties.ini: Para configurar el nombre y autor del sistema.

Configuración de CSS , JavaScript e imágenes:

Se agregó y modificó los estilos, archivos JavaScript e imágenes que se necesita para desarrollar la aplicación

Para desarrollo del sistema aplicación Frontend:

Ingreso:

Módulo login:

- Vista:
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Index
- Formulario:
 - IdentificaForm.class

Parámetros:

Módulo Departamento:

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - DepartamentoForm.class
- Modelo
 - DepartamentoTable.class

Los demás módulos y ficheros de parámetros se encuentran en el anexo Núm. 3 (Sección, Turno, Cargo, Usuarios), por tener el mismo formato no es relevante seguir detallándolos aquí.

Asignación:

Módulo Empleado:

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index

- Formulario:
 - EmpleadoForm.class
- Modelo
 - EmpleadoTable.class

Los demás módulos y ficheros de parámetros se encuentran en el anexo Núm. 3 (Permisos, Movilizaciones, Vacaciones), por tener el mismo formato no es relevante seguir detallándolos aquí.

Carga:

Módulo Carga y cálculo:

- Vista:
 - Index.php
 - Partials
 - CargaDatosSuccess.php
 - CargaHTSuccess.php
 - CargaRegistroSuccess.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, index, CargaDatos, CargaHT, CargaRegistro.
- Formulario:
 - EmpleadoForm.class
- Modelo
 - EmpleadoTable.class

Consulta y Reportes:

Módulo Registro:

- Vista:
 - Index.php
 - Partials
 - _ListaReg.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Index, Rep, Repp.
 -
- Formulario:
 - RegistroForm.class
- Modelo
 - RegistroTable.class

3.4.2. Pantallas.

INGRESO

ENTRADA:

Usuario =Código Reloj
(Numérico)

Contraseña= Alfanumérico

SALIDA:

Si administrador=

parámetros . Si es usuario Consulta

Fuente: Sistema Actual.

Figura 3.14. Descripción E/S Pantalla Ingreso.

PARÁMETROS

Fuente: Sistema Actual.

Figura 3.15. Descripción E/S Pantalla Parámetros Departamento Nuevo

Coddepartamentos	Descripcion	Observaciones	Editar	Borrar
1	ADMINISTRATIVO FINANCIERO			
2	PATRIMONIO NATURAL			

Fuente: Sistema Actual.

Figura 3.16. Descripción E/S Pantalla Parámetros Departamento Listado.

Entrada: Ingreso de datos en los campos descripción y observación.

Salida: Listado de los departamentos existentes adicionado el nuevo registro.

Las demás pantallas de parámetros se encuentran en el anexo núm. 3, ya que son de similar entrada y salida de información en su estructura y formato.

ASIGNACIÓN

NUEVO EMPLEADO

PinEmpleado	<input type="text"/>
Departamento coddepartamentos	ADMINISTRATIVO FINANCIERO
Seccion codseccion	SECRETARIA
Cargo numcargo	SERVIDOR PUBLICO DE APOYO 1
Turno numturno	NORMAL
Cédula	<input type="text"/>
Apellidos	<input type="text"/>
Nombres	<input type="text"/>
Telefono	<input type="text"/>
Celular	<input type="text"/>
Estado	<input type="text"/>
Observacion	<input type="text"/>
<input type="button" value="Regresar"/> <input type="button" value="Guardar"/>	

Fuente: Sistema Actual.

Figura 3.17. Descripción E/S Pantalla Asignación Empleado Nuevo.

Sistema de Control de Asistencia
Dirección Provincial del Ambiente de Imbabura

PARAMETROS ASIGNACION CARGAR CONSULTA Y REPORTES

LISTA DE EMPLEADOS

Cod.	Departamento	Seccion	TurnoEmplead	Cédula	Apellidos	Nombres	Telefono	Celular	Estado	Observacion	Editar	Borrar
101	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		FK	F			ACTIVO			
102	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		S	S			ACTIVO			
103	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		a	a			ACTIVO			
104	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		G	G			ACTIVO			
105	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		k	k			ACTIVO			
106	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		h	h			ACTIVO			
107	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		y	y			ACTIVO			

Fuente: Sistema Actual.

Figura 3.18. Descripción E/S Pantalla Asignación Empleado Listado.

Entrada: Ingreso de datos en los campos: pin empleado (Código numérico de 3 dígitos generado por el reloj), departamento, sección, cargo, turno, (por medio de combo box generando consulta de estos parámetros de la base de datos), cédula, nombres, apellidos, teléfono, celular, estado, observaciones, (datos informativos del empleados alfanuméricos).

Salida: Listado de los empleados existentes adicionado el nuevo registro.

Las demás pantallas de asignación se encuentran en el anexo núm. 3, ya que son de similar entrada y salida de información en su estructura y formato.

CARGA.

Cargar.



Fuente: Sistema Actual.

Figura 3.19. Descripción E/S Pantalla Cargar Datos al sistema.

Entrada: Ingreso de archivo de texto donde se encuentran las timbradas de los empleados, clic en imagen.

Salida: Mensaje en pantalla "Datos Cargados correctamente".

Calcular.



Fuente: Sistema Actual.

Figura 3.20. Descripción E/S Pantalla Calcular Atrasos, Horas extras, Horas Trabajadas.

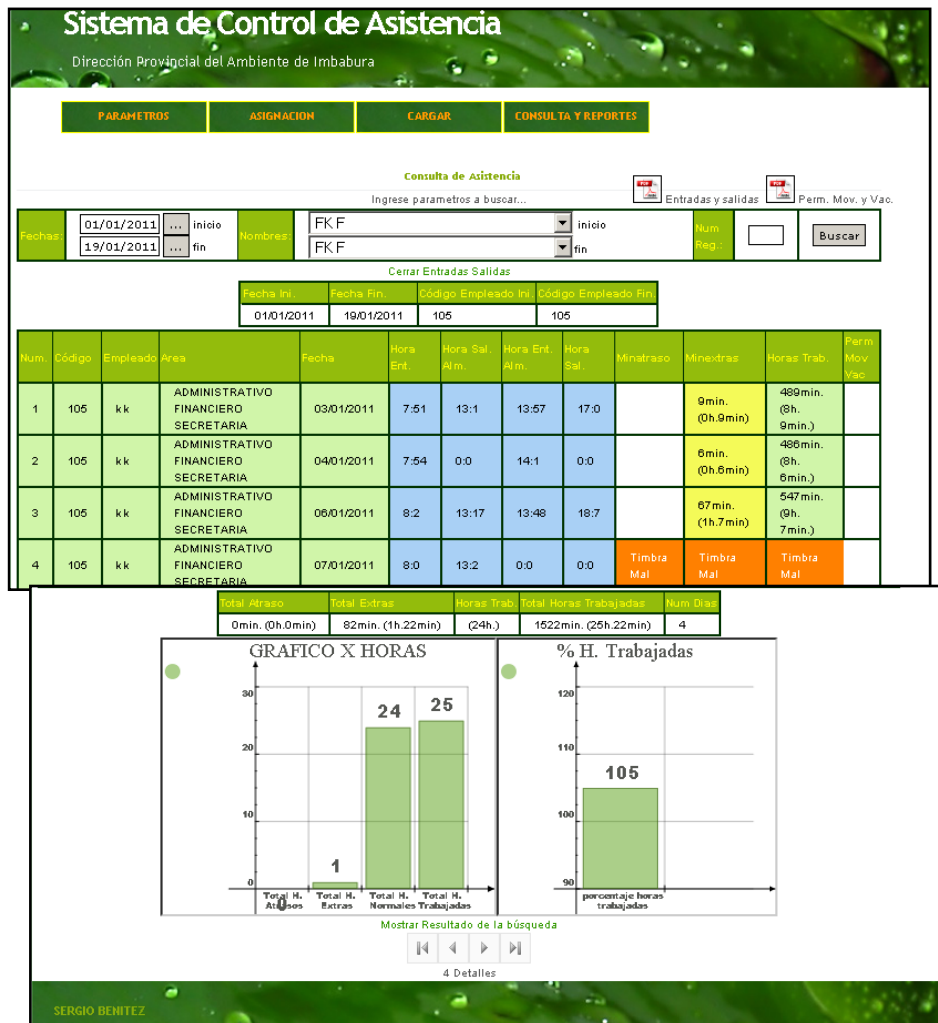
Entrada: Ingreso de rango de fechas para realizar la búsqueda, clic en calcular.

Salida: Mensaje en pantalla "Datos Calculados correctamente".

CONSULTA Y REPORTE.

Fuente: Sistema Actual.

Figura 3.21. Descripción E/S Pantalla Consulta y reporte de E/S empleados.



Fuente: Sistema Actual.

Figura 3.22. Descripción E/S Pantalla Resultado consulta y reporte de E/S empleados.

Cod.	Nombre	Departamento
101	FK F	ADMINISTRATIVO FINANCIERO

Num	Fecha	HoraEntrada	HoraSalAIm	HoraEntAIm	HoraSalida	Atrasos	Extras	H. Trab.	Observaciones	PIMV
1	2011-01-03	8:3	0:0	14:14	17:14	0	14	8:14		permiso
2	2011-01-04	8:7	13:23	14:3	15:57	63	0	6:57		
3	2011-01-05	8:12	13:12	13:59	17:18	0	16	8:16		
4	2011-01-06	8:12	13:15	13:55	17:40	0	38	8:38		
5	2011-01-07	8:9	13:10	14:4	17:25	0	25	8:25		movilizacion
6	2011-01-08	8:18	13:6	13:55	15:2	110	0	6:10		
7	2011-01-10	7:45	13:14	13:56	17:53	0	68	9:8		
8	2011-01-11	8:11	13:13	13:57	17:27	0	26	8:26		permiso
9	2011-01-12	8:3	13:12	13:53	17:17	0	17	8:17		
10	2011-01-13	8:7	13:8	13:56	17:44	0	44	8:44		
11	2011-01-14	8:3	0:0	14:16	17:31	0	31	8:31		
12	2011-01-17	8:7	13:19	13:54	0:0	0	0		Mia Tomorozo	Mia Tomorozo
13	2011-01-18	8:8	13:14	0:0	16:58	2	0	7:58		
14	2011-01-19	7:57	13:5	14:6	0:0	0	3		Mia Tomorozo	Mia Tomorozo

TOTAL Registros: 14

Fin Reporte
Sergio Benitez Web Master

Fuente: Sistema Actual.

Figura 3.23. Descripción E/S Pantalla Resultado consulta y reporte de E/S empleados en pdf.

Cod.	Nombre	Departamento
101	FK F	ADMINISTRATIVO FINANCIERO

Num	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias
1	permiso	2011-01-07	2011-01-07	2011-01-10		Otros	ASD			3
2	movilizacion	2011-01-07	2011-01-07	2011-01-10	DSFSD	SDF	DSF			42 horas
3	permiso	2011-01-07	2011-01-07	2011-01-10		Otros	ASD			3

Fin Reporte
Sergio Benitez Web Master

Fuente: Sistema Actual.

Figura 3.24. Descripción de Pantalla de Permisos, Movilizaciones y vacaciones de empleados en pdf.

Entrada:

- Escoger de rango de fechas para realizar la búsqueda.
- Escoger de rango de nombres.
- Clic en buscar.

Salida: Mensaje en pantalla “Número, nombre, departamento y sección, horas de entrada y salida, Minutos de atraso, extras y trabajadas, y si tiene permiso o no”.

Gráfico estadístico de Atrasos y Extras así como las horas trabajadas.

Gráfico estadístico de Porcentaje de Horas trabajadas.

En esta parte se puede exportar a pdf la consulta y además los permisos movilizaciones y vacaciones que ha tenido el empleado/os en el lapso de tiempo que consultó el empleado.

3.4.3. Listados.

Aquí se explica la lógica de la creación de funciones y líneas de código que permiten la introducción de parámetros, carga de datos y consulta de información hacia la base de datos. Revisar en el anexo núm. 4

3.5. Cadenas de Ejecución.

Se detalla la ruta en donde se encuentra los diferentes módulos y ficheros de ejecución de los diferentes procesos de ejecución del sistema.

3.5.1. Scripts.

La dirección raíz es www.asistenciamai.com/

Para Ingreso:

www.asistenciamai.com/

No es necesario poner login porque esta ruteado para que empiece por defecto la página en esta dirección al digitar la dirección raíz.

Para parámetros:

Departamentos

- Nuevo: /Departamento/new
- Editar: / Departamento/edit/coddepartamentos/1

- Eliminar: / Departamento/delete/coddepartamentos/1
- Listar: /Departamento

Sección

- Nuevo: /Seccion/new
- Editar:/ Seccion /edit/codseccion/1
- Eliminar: / Seccion /delete/codseccion /1
- Listar: / Seccion

Turno

- Editar:/ Turno /edit/codturno/1

Cargo

- Nuevo: /Cargo /new
- Editar:/ Cargo /edit/numcargo/1
- Eliminar: / Cargo /delete/numcargo /1
- Listar: / Cargo

Usuario

- Nuevo: /Usuario /new
- Editar:/ Usuario /edit/idusuario/1
- Eliminar: / Usuario /delete/idusuario /1
- Listar: / Usuario

Para Asignación:

Empleados

- Nuevo: /Empleado/new
- Editar:/ Empleado /edit/pinempleado/1
- Eliminar: / Empleado /delete/pinempleado /1
- Listar: /Empleado

Permisos

- Nuevo: /Permiso/new
- Editar:/ Permiso /edit/numpermiso /1

- Eliminar: / Permiso /delete/numpermiso /1
- Listar: / Permiso

Movilización:

- Nuevo: /Movilizacion /new
- Editar:/ Movilizacion /edit/nummovilizacion /1
- Eliminar:/Movilizacion /delete/nummovilizacion /1
- Listar: / Movilizacion

Vacación:

- Nuevo: /Vacacion /new
- Editar:http: / Vacacion /edit/numvacacion/1
- Eliminar: http: / Vacacion /delete/numvacacion /1
- Listar: / Vacacion

Para Cargar Datos:

Carga

- Inicio: /Carga

Calcular

- Inicio: /Carga/CalcularDatos

Para Consultas y reportes:

Entradas y salidas de personal

- Inicio: /Registro

Pdf Entradas y salidas

- Inicio: /Registro/Rep

Pdf Permisos Movilizaciones y Vacaciones

- Inicio: /Registro/Repp

3.6. Resultados de Pruebas de cada módulo.

Para la comprobación del funcionamiento de los módulos se ha trabajado en primera instancia con datos de muestra, arrojando como resultado después de varios intentos y correcciones, un proceso exitoso en todas las pruebas de creación de nuevos registros, edición y eliminación de los mismos, así como la validación requerida en cada módulo.

Luego de esto, se procedió a limpiar y cargar nuevamente con datos reales, los resultados fueron exitosos, ya que en la muestra se utilizó un archivo de texto real para cargar los datos.

3.7. Resultados de pruebas del Sistema.

Generalmente si los módulos se encuentran bien es de esperarse que el sistema integralmente funcione de igual forma.

Las únicas reformas que se hizo fueron en la interfaz visual en la que se sugirió algunas modificaciones en el tamaño de fuente y formato de la consulta.

3.8. Resultados de pruebas de Integración.

Al subir al servidor se realizó la modificación a las configuraciones de enrutamiento y enlaces, lo cual arrojó una integración exitosa.

Cabe recalcar que se probó la interfaz en 3 navegadores, Mozilla 3, Internet Explorer desde versión 8, y Google Chrome, adaptado a estos 3 navegadores la visualización de la interfaz.

El entrenamiento a los usuarios y administrador para integrarse al nuevo sistema fue relativamente fácil ya que conocían el funcionamiento del

anterior sistema, con lo cual el nuevo era mucho más fácil y concreto en lo que pedían, se lo realizó en una capacitación de menos de 1 hora a los administradores y usuarios en 15 minutos.

3.9. Manual de operadores del sistema.

VER ANEXO NUM 5

3.10. Manual de usuario del sistema.

VER ANEXO NUM 6

Capítulo IV

Transición:

Pruebas, Implantación y Mantenimiento del Sistema Web.

**Explicar resultados obtenidos en la
implantación del sistema.**

4.1. Pruebas.

4.1.1. Plan de pruebas del sistema.

En esta tabla se detalla en forma general los casos de prueba que se van analizar.

Para ver las especificaciones consultar el Anexo Núm. 9.

Caso de prueba	Descripción
Base de datos de prueba	Tablas con los campos y datos de prueba.
Caso de prueba de Administrar Parámetros	Evaluar la administración (Nuevo, Editar, Eliminar y Listar), de los parámetros (Departamento, Sección, Turno, Cargo, Usuarios)
Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones	Evaluar la Administración del empleado (Nuevo, Editar, Eliminar y Listar), y la asignación de Permisos, Movilizaciones y Vacaciones.
Cargar y calcular Datos generados del reloj biométrico hacia el sistema	Comprobar la descarga de los datos por el cable, la carga al sistema y el cálculo de la respectiva información.
Consultar y Reportar la asistencia del personal	Consultar las entradas y salidas del personal con sus respectivos atrasos, extras y permisos generados por los cálculos. La transformación de la consulta en pantalla a formato pdf.

Fuente: Propia.

Tabla 4.1. Plan de pruebas del sistema.

4.1.2. Informe de los resultados de las pruebas.

En esta tabla se detalla en forma general resultados de los casos de prueba que se han analizado. Para ver las especificaciones consultar el Anexo Núm. 9.

Caso de prueba	Descripción	Resultado
Base de datos de prueba.	Tablas con los campos y datos de prueba.	Prueba superada con éxito.
Caso de prueba de Administrar Parámetros.	Evaluar la administración (Nuevo, Editar, Eliminar y Listar), de los parámetros (Departamento, Sección, Turno, Cargo, Usuarios).	Prueba superada con éxito.
Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones.	Evaluar la Administración del empleado (Nuevo, Editar, Eliminar y Listar), y la asignación de Permisos, Movilizaciones y Vacaciones.	Prueba superada con éxito.
Cargar y calcular Datos generados del reloj biométrico hacia el sistema.	Comprobar la descarga de los datos por el cable, la carga al sistema y el cálculo de la respectiva información.	Prueba superada con éxito.
Consultar y Reportar la asistencia del personal.	Consultar las entradas y salidas del personal con sus respectivos atrasos, extras y permisos generados. Por los cálculos. Y la transformación de la consulta en pantalla a formato pdf.	La consulta se realizada con éxito. Los gráficos se realizan con éxito. Prueba superada con éxito La generación del pdf se realiza con éxito.

Fuente: Propia.

Tabla 4.2. Informe de resultados de las pruebas.

4.1.3. Acciones a tomar para corregir las desviaciones.

Caso de prueba	Desviación	Correcciones
Base de datos de prueba	Incompatibilidad del Idioma entre la aplicación y la base de datos	Agregar en el archivo de configuración de apache la línea <code>AddDefaultCharset UTF-8</code>
Caso de prueba de Administrar Parámetros	Prueba superada con éxito	No hay Acciones
Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones	Prueba superada con éxito	No hay Acciones
Cargar y calcular Datos generados del reloj biométrico hacia el sistema	Prueba superada con éxito	No hay Acciones
Consultar y Reportar la asistencia del personal	La consulta se realizada con éxito. Los gráficos se realizan con éxito. Prueba superada con éxito La generación del pdf se realiza con éxito.	No hay Acciones

Fuente: Propia.

Tabla 4.3. Acciones a tomar para corregir desviaciones.

4.2. Instalación.

Para la instalación del sistema sólo es necesario copiar el contenido de la carpeta CONTROL_ASISTENCIA a la raíz donde se alojan las páginas web, la carpeta Symfony en las librerías de la aplicación y la carpeta web de la aplicación en la carpeta pública del hosting. Más detalles revisar anexo Núm. 9 “Informe de Instalación”.

4.2.1. Planes de contingencia.

Si el servidor no cuenta con el software requerido, se entrega instaladores para su funcionamiento en el caso de php, apache y MySQL (XAMP), y la carpeta Symfony que se encuentra ya personalizada para el funcionamiento de este sistema.

Para la descarga de datos se deja el instalador de WAD, que es quien gestiona la descargar de los datos del reloj del biométrico.

4.2.2. Plan de revisión post – instalación.

Luego de la instalación es necesario revisar los parámetros de la base de datos, se tiene un script (asistencia.slq) para poder recuperar todos los parámetros que se han probado con éxito.

Además es indispensable que el cable que une al computador y reloj se encuentre bien instalado y realizar una prueba de conexión del computador al reloj por medio del programa que descarga los datos (WAD), al momento de abrirlo y hacer clic en conectar, el reloj debe prender su pantalla, así verificamos que existe conexión.

4.2.3. Informe de la instalación.

Ver anexo núm. 9.

4.2.4. Carta de aceptación del sistema.

Ver anexo núm. 8.

4.3. Mantenimiento.

4.3.1. Listado de fallos detectados.

Caso de prueba	Fallos Detectados
Base de datos de prueba	Incompatibilidad del Idioma entre la aplicación y la base de datos.
Caso de prueba de Administrar Parámetros	No se detecta.
Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones	No se detecta.
Cargar y calcular Datos generados del reloj biométrico hacia el sistema	No se detecta.
Consultar y Reportar la asistencia del personal	No se detecta.

Fuente: Propia.

Tabla 4.4. Listado de fallos detectados.

4.3.2. Listado de mejoras solicitadas por los usuarios.

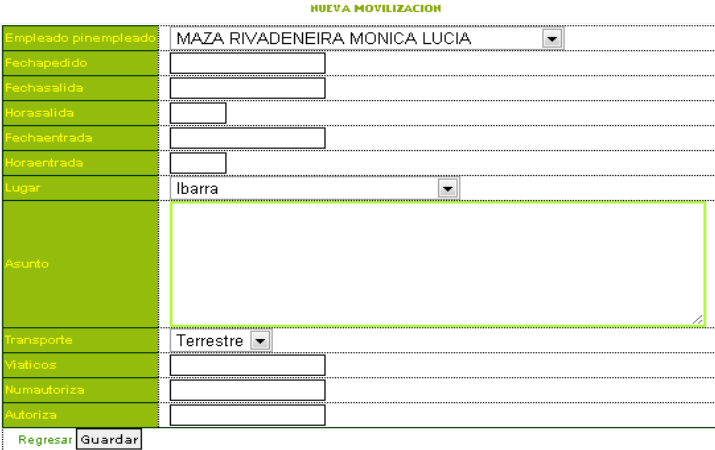
Caso de prueba	Mejoras solicitadas
Base de datos de prueba	En la tabla movilizaciones eliminar el campo Objetivo este campo no es necesario según las autoridades
Caso de prueba de Administrar Parámetros	No hay Cambios
Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones	En el formulario movilizaciones eliminar el campo Objetivo este campo no es necesario según las autoridades
Cargar y calcular Datos generados del reloj biométrico hacia el sistema	No hay cambios
Consultar y Reportar la asistencia del personal	No hay cambios

Fuente: Propia.

Tabla 4.5. Listado de mejoras solicitadas por los usuarios.

4.3.3. Trazo detallada de los cambios realizados.

Caso de prueba	Cambios Realizados
Base de datos de prueba	Eliminar el campo objetivo de la tabla movilizaciones de la base de datos con la secuencia <code>ALTER TABLE `movilizacion` DROP `objetivo`</code> con phpmyadmin .
Caso de prueba de Administrar Parámetros	No hay cambios
Caso de prueba de Crear Empleados y asignar Permisos,	Entrar al panel de la aplicación de NetBeans 6.9, y en <code>CONTROL_ASISTENCIA/apps/frontend/modules/Movilizacion/</code>

<p>Movilizaciones y Vacaciones</p>	<p>Templates/_form.php, eliminamos las siguientes líneas de código</p> <pre> <tr> <th><?php echo \$form['objetivo']->renderLabel() ?></th> <td> <?php echo \$form['objetivo']->renderError() ?> <?php echo \$form['objetivo'] ?> </td> </tr> </pre> <p>Ya esta suprimido el campo objetivo el resultado es:</p> 
<p>Cargar y calcular Datos generados del reloj biométrico hacia el sistema</p>	<p>No hay Cambios</p>
<p>Consultar y Reportar la asistencia del personal</p>	<p>No hay cambios</p>

Fuente: Propia.

Tabla 4.6. Traza detallada de los cambios realizados.

4.3.4. Actas de trabajo y aceptación.

Ver anexo núm. 8.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones, recomendaciones, glosario de términos y referencias.

**Reflexiones de apoyo para mejorar al
sistema.**

Conclusiones:

- Con el sistema web se ha solucionado en forma eficiente la consulta y reporte de la asistencia del personal de la institución, ya que es muy fácil de manejar, rápido en las consultas y reportes, y a medida de los requerimientos de la institución.
- Al ser un sistema Web permite integrar la información al sistema desde la ciudad de Cotacachi donde se encuentra los datos del reloj biométrico para poder procesarlos y consultarlos desde cualquier ubicación.
- El framework Symfony ayudó a organizarlos archivos de las clases, acciones, plantillas, etc., usadas en el proyecto, así como plugins, estilos, configuraciones, etc., Además se uso el IDE NetBeans para poder visualizarlos crearlos y modificarlos.
- El sistema puede ser entendido y modificado fácilmente ya que se utilizó estándares de programación y una arquitectura de software MVC, que organiza el código para ser ilustrado con claridad.
- El presente proyecto tiene mucho futuro, ya que, el gobierno va a controlar a sus empleados en todas las instituciones públicas y sería una gran oportunidad para poder implantarlo en otras instituciones.

Recomendaciones:

- Se recomienda que se que se escoja a las personas que van a administrar el sistema ya que es necesario manejar los datos con mucha reserva.
- Se recomienda a los funcionarios tratar de timbrar las 2 entradas y 2 salidas del día y a su vez comunicar los permisos, movilizaciones y vacaciones, antes de generar los reportes para que no exista desacuerdos entre empleados y autoridades por los reportes, ya que el sistema realiza los cálculos mediante la información que posee el dispositivo biométrico, y también los parámetros ingresados por el administrador, y si no tiene estos va a generar mensajes de mal timbrado y no va a obtener horas trabajadas en el día.
- Antes de comenzar un proyecto se debe analizar los requerimientos de usuario a fondo y si tiene otro sistema analizarlo y que sirva de base y guía para el proyecto, pero ante todo, hay que tomar mucho énfasis en depurar la base de datos que es la columna vertebral del proyecto.
- Hay diferentes marcas y tipos de relojes biométricos que pueden utilizar el presente sistema, sólo una pequeña modificación en una línea de código, puede reconocer el formato del archivo que se descarga de cualquier dispositivo.
- Lo recomendable en estos proyectos es empatar la información que genera el reloj biométrico así como los parámetros ingresados en el sistema.

Glosario de Términos:

- **Atributo:** Es una especificación que define una propiedad de un elemento o archivo.
- **Base de Datos:** Es un conjunto de datos pertenecientes a un mismo contexto.
- **Colector de datos biométrico:** Dispositivo que almacena número de usuario y el horario y la fecha en que se produjo la fichada.
- **Framework:** Es una estructura conceptual y tecnológica definida que brinda ayuda para desarrollo de programas.
- **Fichero:** Archivo realizado en un programa.
- **Hardware:** Corresponde a todas las partes perceptibles de una computadora.
- **Herramientas:** objeto elaborado a fin de facilitar la realización de un
- **Hosting:** Un servicio de almacenaje o de alojamiento en el internet para almacenar información.
- **Interacción:** Se refiere a una acción recíproca entre dos o más objetos.
- **Iteración:** Repetición de una serie de instrucciones.
- **Interfaz:** Es la apariencia en donde el usuario se comunica con el programa.
- **Línea de código:** Trozo de texto que es parte del sistema y que ejecuta una acción.
- **Medios biométricos:** Dispositivos que utilizan como medio de autenticación de una persona los rasgos dactilares, faciales, etc.
- **Metodología:** Reglas utilizadas para alcanzar los objetivos de un proyecto.
- **Módulo:** Es una porción de un programa de computadora.
- **Nodo:** Punto de intersección o unión de varios elementos.
- **Parámetro:** Es una variable que puede ser recibida por una acción del sistema.
- **Plataforma:** Sistema para crear programas.

- **Portabilidad:** El sistema se puede ejecutar en plataformas.
- **Reloj biométrico:** Dispositivo que utiliza la tecnología de identificación por huella digital y almacena en memoria el número de usuario y el horario y la fecha en que se produjo la fichada.
- **Servidor de Aplicaciones:** Computador principal que ejecuta ciertas aplicaciones.
- **Sistema Web:** Es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados subidos al internet.
- **Sitio web:** Es una colección de páginas web relacionadas y comunes a un lugar de Internet.

Referencias:

- [1] “Cómo instalar XAMPP en Windows”, <http://www.nosolocodigo.com/como-instalar-xampp-en-windows>
- [2] “Instalar symfony en windows”, <http://www.smirriaulosabe.com/instalar-symfony-en-windows/>
- [3] “Cómo configurar un proyecto en Symfony con NetBeans 6.9”, <http://www.n4gash.com/2010/configurar-proyecto-symfony-netbeans/>
- [4] “Mi primer proyecto con Symfony”, <http://www.tecnoretas.com/programacion/mi-primer-proyecto-con-symfony-127-parte-i/>
- [5] “Symfony la guía definitiva ”, http://librosweb.es/symfony_1_2/
- [6] “El tutorial Joobet”, http://librosweb.es/jobeeet_1_3/
- [7] “Symfony Avanzado”, <http://librosweb.es/mas-con-symfony/>
- [8] “Plugins de Symfony”, <http://www.symfony-project.org/plugins/>
- [9] “Learn Symfony” (Aprender Symfony), <http://symfony.com/doc/current/>
- [10] “Generador de pdf”, <http://www.fpdf.org/>

- [11] “Calendario en JavaScript”, http://www.webtaller.com/construccion/lenguajes/javascript/lecciones/calendario_en_javascript.php
- [12] “Manual de las librerías FPDF”, <http://www.desarrolloweb.com/manuales/manual-fpdf.html>
- [13] “Crear menús de navegación en CSS usando listas”, <http://www.cristalab.com/tutoriales/crear-menus-de-navegacion-en-css-usando-listas-c130/>

Libros:

- [1] DEBRAUWER Y VAN DER HEYDE, Laurent, Fien; “UML2 Iniciación, ejemplos, y ejercicios corregidos”. 2da Edición. Editorial ENI, 2009.
- [2] AMO, Fernando Alonso; “Introducción a la Ingeniería de Software – Modelos de desarrollo de programas”. Editorial DELTA. 2005.
- [3] PAVON PUERTAS, Jacobo, “Creación de un Portal con Php y MySQL”. 3ra. Edición. Editorial. 2010.
- [4] POTENCIER, Fabien; “Symfony la guía definitiva”. Creative Commons. Diciembre 2008.
- [5] POTENCIER, Fabien; “El Tutorial de Jobeet”. Creative Commons. Marzo 2009

ANEXOS

Prototipos, Documentos, Manuales y otros....

Documentos Adicionales del proyecto.

ANEXO NUM 1: Diccionario de Datos

Descripción de tablas de la Base de datos del sistema:

- **PARÁMETROS:**

Tabla USUARIO: Almacena datos para el ingreso del sistema

Nombre Variable	Tipo Dato	Descripción
idUSUARIO	Entero	Clave principal de la tabla, es auto incrementable
USUARIO	Alfanumérico lím. 45	Nombre del Usuario
CONTRASENA	Alfanumérico lima. 8	Contraseña del Usuario definida por el administrador
TIPO	Alfanumérico lima. 20	Puede ser administrador o empleado consultor

Tabla DEPARTAMENTO: Almacena datos del nombre de departamentos que pertenecen a la institución.

Nombre Variable	Tipo Dato	Descripción
CodDepartamentos	Entero	Clave principal de la tabla, es auto incrementable
Descripcion	Alfanumérico Lím. 100	Es el nombre del departamento de la institución.
Observaciones	Alfanumérico Lím. 45	Campo opcional de descripción

Tabla SECCIÓN: Almacena datos del nombre de la Sección que tiene los departamentos.

Nombre Variable	Tipo Dato	Descripción
CodSeccion	Entero	Clave principal de la tabla, es auto

		incrementable
CodDepartamento FK	Entero	Clave Foránea del Departamento. Para comprobar integridad y relación entre tablas.
Descripcion	Alfanumérico Lím. 100	Es el nombre del la sección del departamento.

Tabla TURNO: Almacena datos del nombre de la Sección que tiene los departamentos.

Nombre Variable	Tipo Dato	Descripción
NumTurno	Entero	Clave principal de la tabla, es auto incrementable
Descripcion	Alfanumérico Lím. 100	Es el nombre del la sección del departamento.
HoraEntrada	Entero	Hora de entrada al trabajo
MinEntrada	Entero	Minuto de entrada al trabajo
HoraSalAlmuerzo	Entero	Hora de Salida al almuerzo
MinSalAlmuerzo	Entero	Minuto de Salida al almuerzo
HoraEntAlmuerzo	Entero	Hora de Entrada del almuerzo
MinEntAlmuerzo	Entero	Minuto de entrada del almuerzo
HoraSalida	Entero	Hora de Salida del trabajo
MinSalida	Entero	Minuto de Salida del trabajo
MinutosGracia	Entero	Minutos de Gracia al entrar al trabajo, para no tomarlo como atraso.

Tabla CARGO: Almacena datos del nombre del cargo que existe en la institución.

Nombre Variable	Tipo Dato	Descripción
NumCargo	Entero	Clave principal de la tabla, es auto incrementable
Descripcion	Alfanumérico Lím. 45	Es el nombre del cargo.
Observaciones	Alfanumérico Lím. 45	Campo opcional de descripción

- **ASIGNACIÓN:**

Tabla EMPLEADO: Almacena datos de los empleados de la institución.

Nombre Variable	Tipo Dato	Descripción
PinEmpleado	Entero	Clave principal de la tabla, asignada manualmente y es el mismo código asignado en el reloj biométrico.
DEPARTAMENTO_Cod Departamentos	Entero	Clave Foránea del departamento al que pertenece el empleado. Para comprobar integridad.
SECCION_CodSeccion	Entero	Clave Foránea de la Sección al que pertenece el empleado. Para comprobar integridad.
CARGO_NumCargo	Entero	Clave Foránea del cargo del empleado. Para comprobar integridad.
TURNO_NumTurno	Entero	Clave Foránea del Turno del empleado. Para comprobar integridad.
Cedula	Alfanumérico Lím. 11	Es la cedula del empleado
Apellidos	Alfanumérico Lím. 45	Almacena apellidos del empleado

Nombres	Alfanumérico Lím. 45	Almacena nombres del empleado
Telefono	Alfanumérico Lím. 11	Almacena el número telefónico del empleado
Celular	Alfanumérico Lím. 20	Almacena el número telefónico celular del empleado
Estado	Alfanumérico Lím. 20	Almacena el Estado del empleado Activo o Pasivo.
Observaciones	Alfanumérico Lím. 45	Campo opcional de descripción

Tabla PERMISOS: Almacena datos de los permisos que piden los empleados.

Nombre Variable	Tipo Dato	Descripción
NumPermiso	Entero	Clave principal de la tabla, es auto incrementable
Empleado_PinEmpleado	Entero	Clave Foránea del empleado. Para comprobar integridad.
FechaPedido	Fecha (yyyy/mm/dd)	Es la fecha del Pedido
FechaPermiso	Fecha (yyyy/mm/dd)	Es la fecha de salida al permiso
FechaFinPedido	Fecha (yyyy/mm/dd)	Es la fecha que finaliza el Permiso
NumDias	Entero	Generado por caculos internos entre la fecha de permiso a la ficha final de ésta.
Asunto	Alfanumérico Lím. 100	Es una herramienta de tipo combobox para escoger entre las opciones de permiso
Motivo	Alfanumérico	Área de texto donde se describe el motivo del

	Text Area Lím. 200	permiso
NumAutoriza	Entero	Campo donde se almacena el número de autorización para autorizar el permiso
Autoriza	Alfanumérico Lím. 45	Nombre de la autoridad que faculta salir del trabajo.

Tabla VACACIÓN: Almacena datos de las vacaciones que piden los empleados.

Nombre Variable	Tipo Dato	Descripción
NumVacacion	Entero	Clave principal de la tabla, es auto incrementable
Empleado_PinEmpleado	Entero	Clave Foránea del empleado. Para comprobar integridad.
FechaSalida	Fecha (yyyy/mm/dd)	Es la fecha de salida a vacación.
FechaIngreso	Fecha (yyyy/mm/dd)	Es la fecha de Regreso de las Vacaciones pedidas.
DiasV	Entero	Generado por caculos internos entre las fechas de la vacación.
NumAutoriza	Entero	Campo donde se almacena el número de las vacaciones pedidas.
Autoriza	Alfanumérico Lím. 45	Nombre de la autoridad que faculta salir del trabajo.

Tabla MOVILIZACIÓN: Almacena datos de las movilizaciones que piden los empleados.

Nombre Variable	Tipo Dato	Descripción
NumMovilizacion	Entero	Clave principal de la tabla, es auto incrementable

Empleado_PinEmpleado	Entero	Clave Foránea del empleado. Para comprobar integridad.
FechaPedido	Fecha (yyyy/mm/dd)	Es la fecha del Pedido
FechaSalida	Fecha (yyyy/mm/dd)	Es la fecha de salida a la movilización.
HoraSalida	Entero	Hora de salida movilización
FechaEntrada	Fecha (yyyy/mm/dd)	Es la fecha que finaliza la movilización
HoraEntrada	Entero	Hora de Entrada de la movilización
NumHoras	Entero	Generado por caculos internos entre la fechas de inicio y fin de la movilización
Lugar	Alfanumérico Lím. 45	Es una herramienta de tipo combobox para las ciudades y provincias donde se movilizan
Asunto	Alfanumérico Text Area Lím. 200	Área de texto donde se describe el asunto de la movilización.
Objetivo	Alfanumérico Text Area Lím. 200	Área de texto donde se describe el/los Objetivo(s) de la movilización
Trasporte	Alfanumérico Lím. 45	Es una herramienta de tipo combobox para escoger el tipo de transporte utilizado
Viaticos	Entero	Dinero a responder por movilización
NumAutoriza	Entero	Campo donde se almacena el número de autorización para autorizar la movilización
Autoriza	Alfanumérico Lím. 45	Nombre de la autoridad que faculta salir a una movilizacion.

- **CARGAR:**

Tabla CARGA: Almacena datos temporales para cargar los datos generados por el reloj biométrico.

Nombre Variable	Tipo Dato	Descripción
NumCarga	Entero	Clave principal de la tabla, es auto incrementable

CodEmpleados	Entero	Clave Foránea del empleado creado en el reloj biométrico. Para comprobar integridad.
Fecha	Fecha	Es la fecha del instante timbrado
Hora	Entero	La hora del instante timbrado
Min	Entero	El minuto del instante timbrado

- **REGISTRO:**

Tabla REGISTRO: Almacena datos del movimiento de entradas y salidas del empleado, así como los parciales de los minutos de atrasos, horas extras, horas trabajadas y variable bandera.

Nombre Variable	Tipo Dato	Descripción
NumRegistro	Entero	Clave principal de la tabla, es auto incrementable
Empleado_PinEmpleado	Entero	Clave Foránea del empleado creado en el reloj
Fecha	Fecha	Es la fecha del instante timbrado
HoraEntrada	Entero	Hora de entrada al trabajo
MinEntrada	Entero	Minuto de entrada al trabajo
HoraSalAlmuerzo	Entero	Hora de Salida al almuerzo
MinSalAlmuerzo	Entero	Minuto de Salida al almuerzo
HoraEntAlmuerzo	Entero	Hora de Entrada del almuerzo
MinEntAlmuerzo	Entero	Minuto de entrada del almuerzo
HoraSalida	Entero	Hora de Salida del trabajo
MinSalida	Entero	Minuto de Salida del trabajo
MinAtraso	Entero	Parámetro de minutos de atraso calculados
MinExtras	Entero	Parámetro de minutos de horas extras calculados
ResMinAtraso	Entero	Minutos de atraso calculados internamente.
ResMinExtra	Entero	Minutos de horas extras calculados internamente.
ResHT	Entero	Minutos de Horas Trabajadas calculados internamente.
PermVacMov	Alfanumérico	Por medio de un parámetro bandera llenar los campos

ANEXO NUM 2: Lista de Riesgos

Historial de revisiones

Fecha	Versión	Descripción	Autor/Revisor
10/10/2010	1.0	Riesgos identificados al inicio de la fase de Incepción.	Sergio Benítez.
20/02/2011	2.0	Riegos identificados en la etapa de desarrollo.	Sergio Benítez.
01/05/2011	3.0	Riesgos Identificados en la etapa de implantación.	Sergio Benítez.

Propósito

Evaluar la posibilidad de recoger todos los requerimientos para el desarrollo del sistema Web.

Lista de Riesgos

Puntaje/ Magnitud	Descripción del riesgo e impacto	Estrategia de mitigación y/o plan de contingencia
7	Que el director no se encuentre para las reuniones de trabajo que están en cronograma.	Realizar las reuniones con la secretaria y personal que posibilite la recolección de requerimientos. Y hacer luego una sola reunión con el director para su aprobación.
7	Que no sea facilitado el SDK del reloj biométrico, para la descarga de datos del reloj.	Probar con el sistema actual de conversión de datos para realizar el prototipo. Y luego gestionar con el fabricante para que envíe el SDK.
6	El análisis del actual sistema puede llevar mucho tiempo.	Realizar pruebas al nuevo programa, paralelas con el análisis del anterior sistema.
6	El desarrollador no tiene la suficiente experiencia en la metodología RUP y técnicas orientadas a objetos. Esto podría resultar en un producto de baja calidad y en un proyecto ineficiente.	Realizar investigaciones y cursos sobre dudas en temas específicos de RUP y orientación a objetos.
6	El desarrollador no tiene suficiente experiencia en programación Módulo – vista– controlador que se requiere para utilizar el framework Symfony.	Realizar investigaciones y cursos sobre temas de programación MVC y el framework Symfony.
5	El módulo de Carga de datos no sea eficaz.	Examinar y consultar acerca de características de los equipos a utilizar Reloj Biométrico y PC al cual van conectadas.
4	Que el sistema no sea compatible con los browser existentes.	Realizar pruebas en la implantación y sugerir navegadores compatibles.

ANEXO NUM 3: Ficheros Creados para realización del sistema.

PARÁMETROS:

Módulo Sección:

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - SeccionForm.class
- Modelo
 - SeccionTable.class

Módulo Cargo:

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - CargoForm.class
- Modelo
 - CargoTable.class

Módulo Turno:

- Vista:
 - Edit.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, Show, Index
- Formulario:
 - TurnoForm.class
- Modelo
 - TurnoTable.class

Módulo Cargo:

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - CargoForm.class
- Modelo
 - CargoTable.class

Módulo Usuarios:

- Vista:
 - Edit.php
 - New.php
 - Show.php

- Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - UsuarioForm.class
- Modelo
 - UsuarioTable.class

ASIGNACIÓN

PERMISOS

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - PermisoForm.class
- Modelo
 - PermisoTable.class

MOVILIZACIONES:

- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php

- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - MovilizacionForm.class
- Modelo
 - MovilizacionTable.class

VACACIONES:

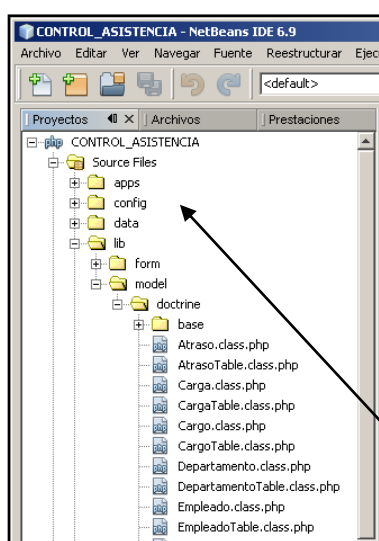
- Vista:
 - Edit.php
 - New.php
 - Show.php
 - Index.php
- Controlador:
 - Actions.class.php
 - Funciones de ejecución, Edit, New, Show, Index
- Formulario:
 - VacacionForm.class
- Modelo
 - VacacionTable.class

ANEXO NUM 4: Listado de Programas.

PARÁMETROS.

Para la administración de los parámetros de departamentos, secciones, cargos, turno y usuarios, administración de datos del empleado y asignación de permisos, movilizaciones y vacaciones, se sigue la misma forma de funciones script y líneas de código entonces se explicará con un solo módulo, el de empleados, la lógica y cómo funcionan.

Como se dijo en el desarrollo se usó el MVC para programar mediante el Framework Symfony que es lo que se va a explicar.



El proyecto se encuentra en la raíz del root (www) el proyecto se llama CONTROL_ASISTENCIA

MÓDULO EMPLEADO

Modelo:

En el modelo se realiza la consulta a la base de datos mediante el lenguaje DQL (Doctrine Query Language) que es similar en estructura a una consulta en MySQL, pero que varía en su sintaxis, a continuación se detalla:

Como se ve en la figura esta clase se encuentra en la carpeta lib/model/doctrine/EmpleadoTable.class.php.

Esta clase “class EmpleadoTable extends Doctrine_Table” tiene 2 funciones para realizar consultas a la base de datos:

1. Esta función recibe un parámetro ($\$ced$), para generar la consulta a la base de datos a la tabla del “Empleado”, retorna los valores de registro de la tabla que coincidan con el parámetro.

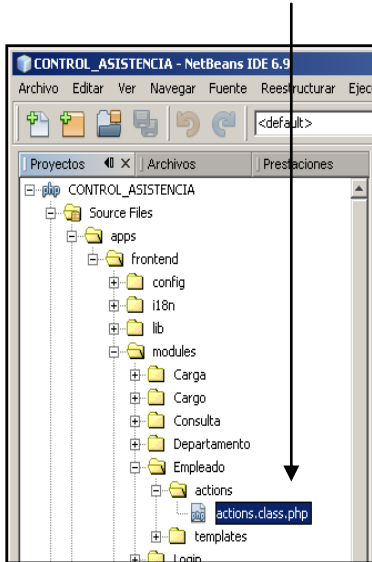
```
public static function getDatos($ced)
{
    $tipo = Doctrine::getTable('Empleado')->find($ced);
    return $tipo;
}
```

2. Esta función genera la consulta de todos los empleados ordenados por el código del empleado

```
public function getRetornaEmp()
{
    $q = Doctrine_Query::create()
        ->from('Empleado d')
        ->orderBy('d.pinempleado ASC');
    $q->execute();
    return $q; }
}
```

Controlador:

Para realizar las operaciones: **nuevo, editar, borrar, listar**, utilizamos el controlador que se encuentra en la aplicación dentro del módulo como se indica en la imagen.



En esta clase “class EmpleadoActions extends sfActions” se encuentran las funciones para las operaciones mencionadas.

Listar: la página principal muestra listado todos los registros y utiliza la función `public function executeIndex(sfWebRequest $request)` que se detalla a continuación.

```
public function executeIndex(sfWebRequest $request)
{
    $this->pager = new sfDoctrinePager('Empleado', 10); //numero a listar
    $this->pager->setQuery(Doctrine::getTable('Empleado')->getRetornaEmp());
    //retira los datos de los registros de la tabla
    $this->pager->setPage($request->getParameter('page', 1)); // es el numero de la
    página en la que inicia la paginación
    $this->pager->init(); // inicia
}
```

Aquí se carga la página principal del módulo, en este caso de “EMPLEADOS”, utilizando la paginación que se explica con comentarios en el código.

Nuevo: para crear un nuevo registro: `public function executeNew(sfWebRequest $request)`, este llama al formulario del empleado que se encuentra en `Lib/Form/Doctrine/EmpleadoForm.class`

```
public function executeNew(sfWebRequest $request)
{
    $this->form = new EmpleadoForm();
}
```

Dentro de este formulario, luego se procede a ejecutar la función “Create”, quien procesa la información que ingresa el usuario en la función `protected function processForm(sfWebRequest $request, sfForm $form)` enviando como parámetro el formulario y validando los campos del formulario, luego si es válido los datos graba y re-direcciona a la página principal index: `$empleado = $form->save(); $this->redirect('Empleado/index');`.

```
public function executeCreate(sfWebRequest $request)
{
    $this->forward404Unless($request->isMethod(sfRequest::POST));

    $this->form = new empleadoForm();

    $this->processForm($request, $this->form);

    $this->setTemplate('new');
}
protected function processForm(sfWebRequest $request, sfForm $form)
{
    $form->bind($request->getParameter($form->getName()), $request->getFiles($form->getName()));
}
```

```

if ($form->isValid())
{
    $empleado = $form->save();
    $this->redirect('Empleado/index');
}

```

Edición: para la edición se utiliza: `public function executeEdit(sfWebRequest $request),`

Se carga la tabla Empleados, enviado como parámetros en código del empleado: `"Doctrine::getTable('empleado')->find(array($request->getParameter('pinempleado'))"`, para luego cargar el formulario del registro solicitado `$this->form = new empleadoForm($empleado);`

Por último: `public function executeUpdate(sfWebRequest $request)` esta función es para validar y actualizar después de haber cargado los datos en el formulario.

```

public function executeEdit(sfWebRequest $request)
{
    $this->forward404Unless($empleado = Doctrine::getTable('empleado')->find(array($request->getParameter('pinempleado'))), sprintf('Object empleado does not exist (%s).', $request->getParameter('pinempleado')));
    $this->form = new empleadoForm($empleado);
}

public function executeUpdate(sfWebRequest $request)
{
    $this->forward404Unless($request->isMethod(sfRequest::POST) || $request->isMethod(sfRequest::PUT));
    $this->forward404Unless($empleado = Doctrine::getTable('empleado')->find(array($request->getParameter('pinempleado'))), sprintf('Object empleado does not exist (%s).', $request->getParameter('pinempleado')));
    $this->form = new empleadoForm($empleado);
    $this->processForm($request, $this->form);
    $this->setTemplate('edit');
}

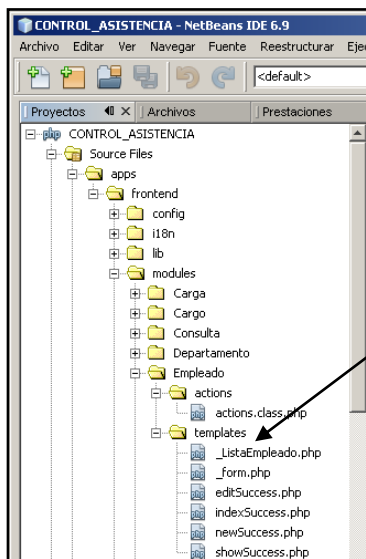
```

Borrar: para eliminar se utiliza la función `public function executeDelete(sfWebRequest $request),` la cual valida `Doctrine::getTable('empleado')->find(array($request->getParameter('pinempleado')))` y borra el registro para luego redireccionar a la página principal `$this->redirect('Empleado/index');`

```

public function executeDelete(sfWebRequest $request)
{
    $request->checkCSRFProtection();
    $this->forward404Unless($empleado = Doctrine::getTable('empleado')->find(array($request->getParameter('pinempleado'))), sprintf('Object empleado does not exist (%s).', $request->getParameter('pinempleado')));
    $empleado->delete();
    $this->redirect('Empleado/index');
}

```



Vista:

La forma de los formulario se lo realiza en estos archivos, aquí se utiliza como lenguaje predominante HTML y Java script, luego php para algunas condiciones de cálculos.

Estos ficheros están en la carpeta template del módulo como se indica en la imagen.

Para todas las acciones hay un formulario los generados para editar, crear, borrar y listar, a continuación:

Nuevo: la vista está en el fichero newSuccess.php, la cual contiene la llamada a la acción de executeNew del controlador.

```
<h1>NUEVO EMPLEADO</h1>
<?php include_partial('form', array('form' => $form)) ?>
```

Aquí lo único que contiene es el título del formulario `<h1>NUEVO EMPLEADO</h1>` y la llamada al formato del formulario `<?php include_partial('form', array('form' => $form)) ?>`, este se encuentra en la carpeta `lib/form/doctrine/EmpleadoForm.class.php`, la cual contiene la estructura de formulario, aquí se usa una clase `class EmpleadoForm extends BaseEmpleadoForm` que contiene una función `public function configure()` la cual contiene los tipos de campos creados para la introducción y validación de datos.

Esta parte se puede manipular para generar la validación de los datos y los tipos de valores que se va a introducir en cada campo, a continuación la explicación:

Entradas (Input).

- Entrada texto.
`sfWidgetFormInputText`
- Entrada de texto desplegable.
`sfWidgetFormInputCheckbox`
- Entrada de contraseñas.
`sfWidgetFormInputPassword`
- Entrada de texto de mayor longitud.
`sfWidgetFormTextarea`
- Entrada de fichero.
`sfWidgetFormInputFile`

Elección:

- Elección de datos desplegable.
`sfWidgetFormSelectCheckbox`

Fechas:

- Para ingreso de fechas (utilizando plugin `sfJQueryUIPlugin`).
`sfWidgetFormDateJQueryUI`
Para poder seleccionar fecha por medio de un calendario dinámico.

Para validar los widgets se utiliza:

```
$this->setValidator('nombres', new sfValidatorString(array(),  
array(  
    'required' => 'Debe ingresar Nombres empleado',  
)));
```

El cual permite sacar el mensaje en pantalla del campo que se requiera, por defecto viene “required”. Pero se puede agregar el texto que se desea.

Otro de las validaciones que se requieren es que todo el texto sea en mayúsculas, para eso utilizamos `'onkeyup' => 'javascript:this.value=this.value.toUpperCase();` en cada campo que se ingrese texto.

EmpleadoForm.class.php

```
public function configure() {  
    $this->widgetSchema['pinempleado'] = new sfWidgetFormInputText(array(), array(  
        'size' => 4,  
    ));  
    $this->setValidator('pinempleado', new sfValidatorString(array(  
        'min_length' => 1,  
        'max_length' => 4  
    ), array(  
        'required' => 'Debe ingresar el Ping del Reloj Biométrico',  
        'min_length' => 'ingresar mínimo 1 numero',  
        'max_length' => 'ingresar máximo hasta 6000'  
    ));  
    $this->widgetSchema['nombres'] = new sfWidgetFormInputText(array(), array(  
        'onkeyup' => 'javascript:this.value=this.value.toUpperCase();', 'size' => 45));  
    $this->setValidator('nombres', new sfValidatorString(array(),  
    array(  
        'required' => 'Debe ingresar Nombres empleado',  
    ));  
    $this->widgetSchema['apellidos'] = new sfWidgetFormInputText(array(), array(  
        'onkeyup' => 'javascript:this.value=this.value.toUpperCase();', 'size' => 45));  
    $this->setValidator('apellidos', new sfValidatorString(array(),  
    array(  
        'required' => 'Debe ingresar Apellidos empleado',  
    ));  
    $this->widgetSchema['telefono'] = new sfWidgetFormInputText(array(), array(  
        'onkeyup' => 'javascript:this.value=this.value.toUpperCase();', 'size' => 11));  
    $this->widgetSchema['celular'] = new sfWidgetFormInputText(array(), array(  
        'onkeyup' => 'javascript:this.value=this.value.toUpperCase();', 'size' => 9));  
    $this->widgetSchema['estado'] = new sfWidgetFormInputText(array(), array(  
        'onkeyup' => 'javascript:this.value=this.value.toUpperCase();');  
    $this->setValidator('estado', new sfValidatorString(array(),  
    array(  
        'required' => 'Debe ingresar Activo o Pasivo',  
    ));  
    $this->widgetSchema['observacion'] = new sfWidgetFormTextarea(array(), array(  
        'onkeyup' => 'javascript:this.value=this.value.toUpperCase();');
```

Editar: la vista está en el fichero editSuccess.php, la cual contiene la llamada a la acción de executeEdit del controlador.

```
<h1>EDITAR EMPLEADO</h1>
<?php include_partial('form', array('form' => $form)) ?>
```

Nuevamente, este archivo contiene el título del formulario `<h1> EDITAR EMPLEADO</h1>` y la llamada al formato del formulario `<?php include_partial('form', array('form' => $form)) ?>`, este se encuentra en la carpeta lib/form/doctrine/EmpleadoForm.class.php, el cual ya se explico anteriormente.

Borrar: la vista está en el fichero _ListaEmpleado.php, la cual contiene la llamada a la acción de executeDelete del controlador.

```
Esto se logra con la llamada de la acción echo link_to('', 'Empleado/delete?pinempleado=' . $emplead->getPinempleado(), array('method' => 'delete', 'confirm' => 'Seguro desea Eliminar?')) ?> que se añade a cada registro, cuando se lista los datos de la tabla en la acción Listar.
```

Listar: la vista está en el fichero index.php que a su vez llama al fichero _ListaEmpleado.php:
Index.php

```
<?php echo include_partial('ListaEmpleado',array('pager'=>$pager));?>
```

Se ve como se llama al código parcial que se encuentra en el fichero _ListaEmpleado.php, adjuntando parámetros para generar la paginación (\$pager).

_ListaEmpleado.php

La estructura de cómo se ven los datos al listarlos, a continuación se explica.

Menú Parcial:

Llamada del menú parcial parámetros, asignación, carga de datos y reportes.

```
<?php include(sfConfig::get('sf_app_template_dir').'/miMenu.php') ?>
```

Encabezado de la tabla

```
<h1>LISTA DE EMPLEADOS</h1>
```

```

<table>
<thead>
<tr>
<th>Cod.</th>
<th>Departamento</th>
<th>Seccion</th>
<!--<th>Cargo</th-->
<th>TurnoEmplead</th>
<th>Cédula</th>
<th>Apellidos</th>
<th>Nombres</th>
<th>Teléfono</th>

<th>Estado</th>

<th>Editar</th>
<th>Borrar</th>
</tr>
</thead>
<tbody>

```

Buscar y listar todos los registros de los empleados:

```

<?php foreach ($pager->getResults() as $Empleado): ?>
<tr>
<td><a href="<?php echo url_for('Empleado/show?pinempleado=' . $Empleado->getPinempleado()) ?>"><?php
echo $Empleado->getPinempleado() ?></a></td>
<?php $datos = DepartamentoTable::getDatos($Empleado->getDepartamentoCoddepartamentos()) ?>
<td><?php echo $datos['descripcion'] ?></td>
<?php $datos = SeccionTable::getDatos($Empleado->getSeccionCodseccion()) ?>
<td><?php echo $datos['descripcion'] ?></td>
<?php $datos = TurnoTable::getDatos($Empleado->getTurnoNumturno()) ?>
<td><?php echo $datos['descripcion'] ?>
<br>
<?php echo $datos['horaentrada'], ':', $datos['minentrada'], '0 A ', $datos['horasalalmuerzo'], ':',
    $datos['minsalalmuerzo'], '0'?>

<?php echo $datos['horaentalmuerzo'], ':', $datos['minentalmuerzo'], '0 A ', $datos['horasalida'], ':',
    $datos['minsalida'], '0 ' ?>

</td>
<td><?php echo $Empleado->getCedula() ?></td>
<td><?php echo $Empleado->getApellidos() ?></td>
<td><?php echo $Empleado->getNombres() ?></td>
<td><?php echo $Empleado->getTelefono() ?></td>

<td><?php echo $Empleado->getEstado() ?></td>

```

Aquí se está llamado a la acción editar y borrar de cada registro.

```

<td><?php echo link_to('',
'Empleado/edit?pinempleado=' . $Empleado->getPinempleado()) ?></td>
<td><?php echo link_to('', 'Empleado/delete?pinempleado=' . $Empleado->getPinempleado(), array('method' => 'delete',
'confirm' => 'Seguro desea Eliminar?')) ?></td>
</tr>

```

Terminación de la búsqueda.

```

<?php endforeach; ?>
</tbody>
</table>

```

Aquí se está llamado a la acción nuevo, por un link:

```

<table><td><a href="<?php echo url_for('Empleado/new') ?>">Nuevo</a></td></table>

```


Paginación:

En esta parte se detalla la forma de funcionamiento:

```
<div class="pagination">
<!--Primera página-->
<?php
echo link_to_remote(image_tag('/images/page-first.gif'),
array('update' => 'contenido',
      'url' => 'Empleado/index?page=' . $pager->getFirstPage(),
      ));
//Anterior
echo link_to_remote(image_tag('/images/page-prev.gif'),
array('update' => 'contenido',
      'url' => 'Empleado/index?page=' . $pager->getPreviousPage(),
      ));
//Páginas centradas-->
if ($pager->haveToPaginate()) {
    $links = $pager->getLinks();
    foreach ($links as $page) {
        $page == $pager->getPage(); ?>&nbsp;
    }
}
//Siguiente-->
echo link_to_remote(image_tag('/images/page-next.gif'),
array('update' => 'contenido',
      'url' => 'Empleado/index?page=' . $pager->getNextPage(),
      ));
//Ultima página-->
echo link_to_remote(image_tag('/images/page-last.gif'),
array('update' => 'contenido',
      'url' => 'Empleado/index?page=' . $pager->getLastPage(),
      ));
?>
</div>

<div class="pagination_desc">
<strong><?php echo $pager->getNbResults() ?></strong> Detalles

<?php if ($pager->haveToPaginate()): ?>
    - pÃgina <strong><?php echo $pager->getPage() ?></strong> <strong><?php echo $pager->getLastPage() ?></strong>
<?php endif; ?>
</div>
```

CONFIGURACIONES Y PROCEDIMIENTOS PREVIOS.

Para el funcionamiento de estos módulos hay que recalcar que se generó el esquema de la base de datos, los módulos, formularios y modelos por medio de comandos que se ejecutan con Symfony, a continuación una explicación:

Symfony doctrine:build-schema

Genera el esquema propio de Doctrine Symfony de la base de datos de MySQL para poder usarlos con el lenguaje DQL (Doctrine Query Language).

Symfony doctrine:build-model

Genera los modelos para todas las tablas

Symfony doctrine:build-form

Genera los formularios base para todas las tablas

Symfony doctrine:generate-module

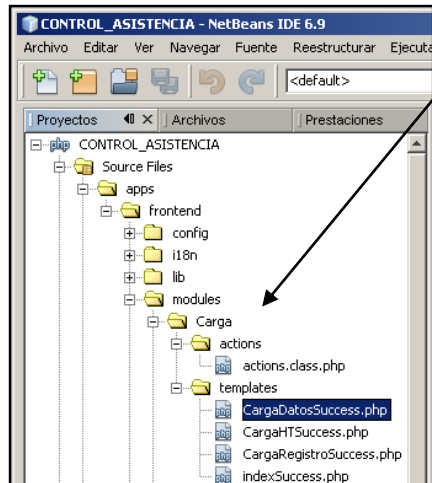
Genera los módulos, se tiene que especificar el nombre de la tabla y de la aplicación, por ejemplo:

with-show frontend Empleado empleado

Se genera el módulo empleado en la aplicación frontend, los datos son extraídos de la tabla Empleado.

CARGA DE DATOS AL SISTEMA.

Se creó un módulo que se llama "Carga", en la cual se encuentra creado 3 acciones importantes de explicar y la página principal.



VISTA:

Se crearon 4 plantillas, pero solo 2 contienen código (CargarDatosSuccess.php e Index.php), los otros 2 solo se declaran porque son obligatorios para llamar a la acción que se necesita.

Index.php

Llamada al menú parcial:

```
<?php include(sfConfig::get('sf_app_template_dir').'/miMenu.php') ?>
```

Cabecera:

```
<h1>CARGAR DATOS AL SISTEMA</h1>
```

En esta parte se realiza el formulario de para cargar el archivo de texto que genera el reloj biométrico.

```
<div id="texto"> Buscar archivo a cargar...
<form action="<?php echo url_for('Carga/index')?>" method="POST">
<table align="center">
```

En esta parte se llama a un formulario (se encuentra en lib/form/ArchivoForm.class.php) generado manualmente en donde se utiliza una clase que contiene un campo para la búsqueda de archivo mediante un cuadro de dialogo abrir.

```
<?php echo $formulario1?>
<tr>
<td ><input align="center" type="image" src="/images/icono-descarga.gif" alt="Cargar" title="Cargar" width=40
height=39/></td>
</tr>
```

```

<!--<td ><input type="image" src="/images/loading.gif" alt="Editar" title="Editar" width=30 height=29/></td-->
</table>
</form>
</div>

```

ArchivoForm.class.php

```

public function configure() {
    $this->setWidgets(array(
'archivo' => new sfWidgetFormInputFile(),
    )
    );
}

```

Aquí se genera la herramienta para el tipo de datos fichero que se necesita generar.

CargaDatosSuccess.php

Se crea una plantilla para calcular los datos después de haberlos cargado.

Llamada al menú parcial:

```
<?php include(sfConfig::get('sf_app_template_dir') . '/miMenu.php') ?>
```

Cabecera:

```
<h1>Caculos de Atrasos, Horas Extras, Horas Trabajadas, </h1>
```

Llamada de Librerías necesarias para generar el calendario dinámico con javascript para la introducción de fechas.

```

<!--Hoja de estilos del calendario -->
<link rel="stylesheet" type="text/css" media="screen" href="/css/calendar-green.css" />

<!-- librería principal del calendario -->
<script type="text/javascript" src="/js/calendar.js"></script>

<!-- librería para cargar el lenguaje deseado -->
<script type="text/javascript" src="/js/lang/calendar-es.js"></script>

<!-- librería que declara la función Calendar.setup, que ayuda a generar un calendario en unas pocas líneas de código -->
<script type="text/javascript" src="/js/calendar-setup.js"></script>

```

Aquí la forma de la acción del formulario para ingresar el rango de fechas para calcular entre estas fechas los atrasos, extras, etc.

```

<form action="<?php echo url_for('Carga/CargaDatos') ?>" method="POST">

<table>

<th>Fecha Inicial:</th>
<td><input type="text" name="fechaini" id="campo_fecha" value="<?php echo $sf_request-
>getParameter('fechaini') ?>" size="10" />
<input type="Submit" value="..." id="lanzador"/>
</td>
<th>Fecha Final:</th>
<td><input type="text" name="fechafin" id="campo_fecha1" value="<?php echo $sf_request-
>getParameter('fechafin') ?>" size="10" />
<input type="Submit" value="..." id="lanzador1"/></td>

```

```
<td><input type="submit" id="btnBuscar" name="txtBuscar" value="Calcular" /></td>
```

```
</table>  
</form>
```

script que define y configura el calendario

```
<script type="text/javascript">  
Calendar.setup({  
  inputField : "campo_fecha", // id del campo de texto  
  ifFormat : "%d/%m/%Y", // formato de la fecha que se escriba en el campo de texto  
  button : "lanzador" // el id del botón que lanzará el calendario  
});  
</script>  
<script type="text/javascript">  
Calendar.setup({  
  inputField : "campo_fecha1", // id del campo de texto  
  ifFormat : "%d/%m/%Y", // formato de la fecha que se escriba en el campo de texto  
  button : "lanzador1" // el id del botón que lanzará el calendario  
});  
</script>  
</div>
```

ACCIONES:

Aquí se detalla la lógica que se utilizó para la carga y cálculo del archivo de texto generado por el reloj biométrico.

```
public function executeIndex(sfWebRequest $request) {
```

Se crea una nueva instancia de tipo ArchivoForm.

```
$this->formulario1 = new ArchivoForm();  
$archivo = $request->getParameter("archivo");
```

Se verifica que tenga el nombre del archivo.

```
if ($archivo <> "") {  
  $NumCarga = "";
```

Se hace una consulta a la tabla carga y se verifica que se encuentre vacía.

```
$q = Doctrine_Query::create()  
  ->select('u.*')  
  ->from('carga u');  
$users = $q->fetchArray();
```

```
if (count($users) == 0) {
```

Si esto ocurre se procede a leer el archivo por medio de sfCsvReader del plugin cargado (sfCsvPlugin) el cual permite la lectura de archivos de texto separados con comas.

```
$reader = new sfCsvReader('textcarga/' . $archivo); //se lee el CVS  
$reader->setSelectColumns('Codigo,Fecha,Hora');  
$reader->open();
```

Se realiza un ciclo while para poder recorrer por todos los registros de las timbradas.

```
while ($data = $reader->read()) { ///recorre todos los piquetes
```

```
//obtención de los datos.
```

```
    $CodEmpleado = $data['Codigo'];
```

```
// fecha original en formato americano (03/01/11).
```

```
    $fecha = $data['Fecha'];  
    $dia = substr($fecha, 0, 2);  
    $mes = substr($fecha, 3, 2);  
    $ano = substr($fecha, -2);
```

```
// fecha final realizada el cambio de formato a las fechas europeas para la  
inserción a la base.
```

```
    $fecha = $ano . '/' . $mes . '/' . $dia;
```

```
//hora para poder extraer la hora y minutos ya que el archivo genera  
(07:33:35) con segundos y no nos interesa la parte de segundos.
```

```
    $horacompleta = $data['Hora'];  
    $Hora = substr($horacompleta, 0, 2);  
    $Min = substr($horacompleta, 3, 2);
```

```
Luego de ir extrayendo los datos del archivo de texto se procede a cargar a la  
tabla temporal carga tal y como viene en el formato.
```

```
    $Carga->setCodempleados($CodEmpleado);  
    $Carga->setFecha($fecha);  
    $Carga->setHora($Hora);  
    $Carga->setMin($Min);  
    $Carga->save();  
}
```

```
Se cierra la lectura del archivo de texto.
```

```
$reader->close();
```

Se re direcciona para cargar la tabla registro, aquí el problema es que el formato de la tabla queda de la siguiente forma: (código, fecha, y timbrada). Y lo que se necesita es cargar en una tabla de la siguiente forma: (código, fecha, hora entrada, hora salida almuerzo, hora entrada almuerzo y hora salida), es por eso que llamamos a la siguiente acción.

```
$this->redirect('Carga/CargaRegistro');
```

```
public function executeCargaRegistro(sfWebRequest $request) {
```

```
Se necesita realizar unas consultas de la tabla carga que tiene ya los datos  
cargados y turno para sacar parámetros necesarios que sirvan de límite para  
calcular los atrasos y extras.
```

```

$q = Doctrine_Query::create()
->select('u.*')
->from('carga u');
$car = $q->fetchArray();

$q1 = Doctrine_Query::create()
->select('u.*')
->from('turno u');
$turn = $q1->fetchArray();
set_time_limit(0);
//////////

```

Se obtiene con un ciclo for los datos de la tabla temporal carga.

```

for ($cont = 0; $cont < count($car); $cont++) {
// // fecha original en formato americano
$codigo = $car[$cont]['codempleados'];
$fecha = $car[$cont]['fecha'];
$día = substr($fecha, -2);
$mes = substr($fecha, 5, 2);
$año = substr($fecha, 0, 4);

$Hora = $car[$cont]['hora'];
$Min = $car[$cont]['min'];
$atraso = ($Hora - $turn[0]["horaentrada"]) * 60;

```

Llenado entrada trabajo, se compara si la hora esta en el rango de entrada

```

if ($Hora >= 6 && $Hora < 12) {

```

Se carga los datos en la tabla registro

```

$Registro = new Registro();
$Registro->setEmpleadoPinEmpleado($codigo);
$Registro->setFecha($fecha);
$Registro->setHoraentrada($Hora);
$Registro->setMinentrada($Min);

```

Luego de creado los registros se actualiza (consulta UPDATE), los campos de las horas dependiendo las condiciones.

//////////cálculo atraso y extras entrada trabajo//////////

```

if ($Hora >= $turn[0]["horaentrada"]) {
    $Registro->setMinatraso(-($atraso));
}
if ($Hora >= $turn[0]["horaentrada"] && $Min > $turn[0]["minutosgracia"]) {
    $Registro->setMinatraso(-(($Min - $turn[0]["minutosgracia"]) + $atraso));
}
if ($Hora < $turn[0]["horaentrada"]) {
    $Registro->setMinatraso(-($Min + $atraso));
}

$Registro->save();
}

```

//////////llenado almuerzo salida////

```

if ($Hora == 12 && $Min > 0 || $Hora == 13 && $Min < 30) {
$q2 = Doctrine_Query::create()
->update('registro r')
->set('r.horasalalmuerzo', $Hora)
->set('r.minsalalmuerzo', $Min)
->where('r.empleado_pinEmpleado =?', $codigo)

```

```

        ->andwhere('r.fecha =?', $fecha)
        ->execute();
    }
}
//////////llenado almuerzo entrada//////////

if ($Hora == 13 && $Min >= 40 || $Hora == 14 && $Min > 0) {
    $q2 = Doctrine_Query::create()
        ->update('registro r')
    ->set('r.horaentalmuerzo', $Hora)
        ->set('r.minentalmuerzo', $Min)
    ->where('r.empleado_pinempleado =?', $codigo)
    ->andwhere('r.fecha =?', $fecha)
        ->execute();
}

```

//////////llenado salida trabajo//////////

```

if ($Hora >= 15) {
    $q2 = Doctrine_Query::create()
    ->update('registro r')
    ->set('r.horasalida', $Hora)
        ->set('r.minsalida', $Min)
    ->where('r.empleado_pinempleado =?', $codigo)
    ->andwhere('r.fecha =?', $fecha)
        ->execute();
}

```

//// Calculo atrasos y extras en salida trabajo//

```

    $extra = ($Hora - $turn[0]["horasalida"]) * 60;
if ($Hora > $turn[0]["horasalida"]) {

    $q1 = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.minextras', $extra - 'r.minextras')
    ->where('r.empleado_pinempleado =?', $codigo)
        ->andwhere('r.fecha =?', $fecha)
    ->execute();
}
if ($Min > $turn[0]["minsalida"]) {

    $q1 = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.minextras', $Min + $extra - 'r.minextras')
    ->where('r.empleado_pinempleado =?', $codigo)
        ->andwhere('r.fecha =?', $fecha)
        ->execute();
}
}
}

```

//////////BORRAR TABLA AUX CARGA //////////

```

    $q = Doctrine_Query::create()
        ->delete('carga a')
    ->where('a.numcarga > 0')
        ->execute();

```

Mensaje de carga completada.

```

echo '<script>alert("Datos Cargados a la Base de Datos con éxito.....")</script>';

```

Por último se direcciona al cálculo en donde se necesita en rango de fechas.

```

$this->redirect('Carga/CargaDatos');
}

```


Este paso es secuencial a la carga ya que en esta parte se calcula los minutos de atrasos, extras, horas trabajadas. Pero se lo ha dividido para que el cálculo lo haga dependiendo la necesidad del reporte, para que no se haga de toda la tabla sino en un rango de fechas.

```

public function executeCargaDatos(sfWebRequest $request) {
//Se verifica que exista la fecha inicio y final para proceder a sacar los datos
de la tabla.
if ($request->getParameter('fechaini') <> 0 && $request->getParameter('fechafin') <> 0) {
    $fechai = $request->getParameter('fechaini');
    $fechaf = $request->getParameter('fechafin');
// Fecha inicio realizada el cambio de formato a las fechas europeas para
ingresar a BD.

$fecha1 = $fechai;
    $dia = substr($fecha1, 0, 2);
    $mes = substr($fecha1, 3, 2);
    $ano = substr($fecha1, -4);
    $fecha1 = $ano . '/' . $mes . '/' . $dia;
// Fecha final realizada el cambio de formato a las fechas europeas para
ingresar BD.
$fecha2 = $fechaf;
    $dia = substr($fecha2, 0, 2);
    $mes = substr($fecha2, 3, 2);
    $ano = substr($fecha2, -4);
    $fecha2 = $ano . '/' . $mes . '/' . $dia;
//Con los parámetros de fechas se busca en la tabla registro para empezar
cálculo de los datos.
$q = Doctrine_Query::create()
    ->select('u.*')
->from('registro u')
    ->where('u.fecha >= ?', $fecha1)
    ->andWhere('u.fecha <= ?', $fecha2);
$car = $q->fetchArray();

if ($request->isMethod('post')) {
set_time_limit(0);
//Se obtiene con un ciclo for los datos de la tabla registro que están en el
rango solicitado.
for ($cont = 0; $cont < count($car); $cont++) {
    $nreg = $car[$cont]['numregistro'];
    $HA = $car[$cont]['minatraso'];
    $HE = $car[$cont]['minextras'];

//Y se actualiza realizado una resta entre Horas de Atraso – Horas extras
para actualizar el campo de minutos de atraso de la tabla registro.
if ($HA < 0 && $HE < 0) {
        $q2 = Doctrine_Query::create()
            ->update('registro r')
->set('r.resminatraso', abs($HA - $HE))
->set('r.resminextra', 0)
            ->where('r.numregistro =?', $nreg)
            ->execute();
    }

if ($HA >= 0 && $HE >= 0) {
        $q2 = Doctrine_Query::create()
            ->update('registro r')
->set('r.resminextra', $HA + $HE)
->set('r.resminatraso', 0)
            ->where('r.numregistro =?', $nreg)
    }
}
}
}

```



```

    $RT = ((8 * 60) - $RA + $RE);
    $q2 = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.resht', $RT)
        ->where('r.numregistro =?', $nreg)
->execute();

/////////Búsqueda de permisos vacaciones movilizaciones ///////////
$pin = $car[$cont]['empleado_pinempleado'];
$fec = $car[$cont]['fecha'];
$q = Doctrine_Query::create()
    ->select('u.*')
    ->from('permiso u')
->where('u.empleado_pinempleado ="' . $pin . '"')
    ->andWhere('u.fechapermiso ="' . $fec . '"');
$reg = $q->fetchArray();
$q1 = Doctrine_Query::create()
    ->select('v.*')
->from('vacacion v')
->where('v.empleado_pinempleado ="' . $pin . '"')
    ->andWhere('v.fechasalida ="' . $fec . '"');
$reg1 = $q1->fetchArray();
$q2 = Doctrine_Query::create()
    ->select('m.*')
->from('movilizacion m')
    ->where('m.empleado_pinempleado ="' . $pin . '"')
    ->andWhere('m.fechasalida ="' . $fec . '"');
$reg2 = $q2->fetchArray();

if ($reg == null && $reg1 == null && $reg2 == null) {
    $p = 0;
    $f = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.permvacmov', $p)
        ->where('r.numregistro =?', $nreg)
        ->execute();
}

if ($reg != null) {
    $p = 1;
    $f = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.permvacmov', $p)
        ->where('r.numregistro =?', $nreg)
        ->execute();
}

if ($reg1 != null) {
    $p = 2;
    $f = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.permvacmov', $p)
        ->where('r.numregistro =?', $nreg)
        ->execute();
}

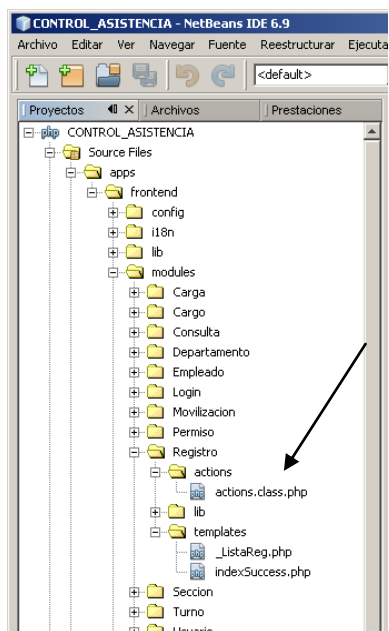
if ($reg2 != null) {
    $p = 3;
    $f = Doctrine_Query::create()
        ->update('registro r')
        ->set('r.permvacmov', $p)
        ->where('r.numregistro =?', $nreg)
->execute();
}
}
}echo '<script>alert("Datos Re-Calculados con éxito.....")</script>';

Por último se re direcciona a la etapa de reportes.
$this->redirect('registro/index');
}

```

CONSULTAS Y REPORTE AL SISTEMA

Esta parte es en donde se realiza la consulta de los resultados de los datos ingresados y los cálculos hechos con estos.



VISTA:

Tenemos 2 archivos:

Index.php que llama al archivo con código parcial ListaReg.php, con el siguiente código,

```
<?php echo  
include_partial('ListaReg',array('pager'=>$pager,'empleados'=>$empleados))  
;?>
```

_ListaReg.php

Esta contiene el código de la estructura de la presentación de los datos que a continuación se explica

Llamada Menú parcial

```
<?php include(sfConfig::get('sf_app_template_dir') . '/miMenu.php') ?>
```

Llamada de los archivos JavaScript para la generación de calendario dinámico.

```
<link rel="stylesheet" type="text/css" media="screen" href="/sfjQueryUIPlugin/css/ui-lightness/jquery-ui.css" />  
<!--<script type="text/javascript" src="/sfjQueryReloadedPlugin/js/jquery-1.3.2.min.js"></script-->  
<script type="text/javascript" src="/sfjQueryUIPlugin/js/jquery-ui.js"></script>  
<!--Hoja de estilos del calendario -->  
<link rel="stylesheet" type="text/css" media="screen" href="/css/calendar-green.css" />
```

```
<!-- librería principal del calendario -->  
<script type="text/javascript" src="/js/calendar.js"></script>
```

```
<!-- librería para cargar el lenguaje deseado -->  
<script type="text/javascript" src="/js/lang/calendar-es.js"></script>
```

```
<!-- librería que declara la función Calendar.setup, que ayuda a generar un calendario en unas pocas líneas de código -->  
<script type="text/javascript" src="/js/calendar-setup.js"></script>
```

Cabecera.

```
<h1>Consulta de Asistencia</h1>
```

```
<div id="titulo">Ingrese parámetros a buscar...</div>
```

```
<div id="texto">
```

Formulario para ingreso de parámetros utilizando el calendario dinámico y la busca de empleados por código para que aparezcan en una lista desplegable.

```

<form action="<?php echo url_for('Registro/index') ?>" method="POST">

<table bgcolor="#D0F5A9">

<th>Fechas:</th>
<td><input type="text" name="query" id="campo_fecha" value="<?php echo $sf_request->getParameter('query')
?>" size="12" />
<input type="Submit" value="..." id="lanzador"/>inicio

<input type="text" name="query1" id="campo_fecha1" value="<?php echo $sf_request->getParameter('query1')
?>" size="12" />

<input type="Submit" value="..." id="lanzador1"/>fin </td>
<th>Nombres:</th>
<td>
<select name="pinc" id="pinc" dir="ltr">
<option value="0">Seleccione nombre...</option>
<?php foreach ($empleados as $usuario): ?>
<option value="<?php echo $usuario->getPinEmpleado() ?>"><?php echo $usuario->getApellidos(), ' ', $usuario-
>getNombres() ?></option>

<?php endforeach; ?>

</select>inicio
<select name="pinc1" id="pinc1" dir="ltr">
<option value="0">Seleccione nombre...</option>
<?php foreach ($empleados as $usuario): ?>
<option value="<?php echo $usuario->getPinEmpleado() ?>"><?php echo $usuario->getApellidos(), ' ', $usuario-
>getNombres() ?></option>
<?php endforeach; ?>

</select>fin
</td>
<th>Num Reg.:</th>
<td><input type="text" name="pag" value="<?php echo $sf_request->getParameter('pag') ?>" size="3" /></td>

<td><input type="submit" id="btnBuscar" name="txtBuscar" value="Buscar" /></td>

</table>

</form>
</div>

```

Script para generar el calendario dinámico.

```

<div id="res" style="display:none;"><a href="javascript:cerrar();"> Cerrar Entradas Salidas</a>
<!-- script que define y configura el calendario-->
<script type="text/javascript">
Calendar.setup({
inputField : "campo_fecha", // id del campo de texto
ifFormat : "%d/%m/%Y", // formato de la fecha que se escriba en el campo de texto
button : "lanzador" // el id del botón que lanzará el calendario
});
</script>
<script type="text/javascript">
Calendar.setup({
inputField : "campo_fecha1", // id del campo de texto
ifFormat : "%d/%m/%Y", // formato de la fecha que se escriba en el campo de texto
button : "lanzador1" // el id del botón que lanzará el calendario
});
</script>

```

/////DECLARACIÓN VARIABLES////////////////////////////////////-->

```

<?php
    $extratotal = 0;
    $atrasototal = 0;
$hf = 0;
    $hm = 0;
    $mit = 0;
    $mim = 0;
    $misa = 0;
    $hsa = 0;
$ht = 0;
    $mintt = 0;
    $cont = 0;
    $num = 0;
    $cont2 = $sf_request->getParameter('pinc');
?>

//////////CABECERA TABLA DE CONSULTA-->
<table>
<thead>
<tr>
<th>Num.</th>

<th>Fecha</th>
<th>Hora Ent.</th>
<th>Hora Sal. Alm.</th>
<th>Hora Ent. Alm.</th>
<th>Hora Sal.</th>
<th>Minatraso</th>
<th>Minextras</th>
<th>Horas Trab.</th>
<th>Perm Mov Vac</th>
</tr>

</thead>
<tbody>

<?php
    $band = 1;
INICIO DE OBTENCIÓN DE DATOS DE LA TABLA REGISTRO
foreach ($pager->getResults() as $registro): $num = $num + 1;
if ($cont2 == $registro->getEmpleadoPinempleado()) {

<tr>
<td bgcolor="#D0F5A9"><?php echo $num ?></td>

<td bgcolor="#D0F5A9"><?php
$fecha2 = $registro->getFecha();
    $dia = substr($fecha2, -2);
    $mes = substr($fecha2, 5, 2);
    $ano = substr($fecha2, 0, 4);
// fecha realizada el cambio de formato AMERICANO
    $fecha2 = $dia . '/' . $mes . '/' . $ano;
echo $fecha2;

    ?></td>
<td bgcolor="#A9D0F5"><?php echo $registro->getHoraentrada(), '|', $registro-
>getMinentrada() ?></td>
<td bgcolor="#A9D0F5"><?php echo $registro->getHorasalalmuerzo(), '|', $registro-
>getMinsalalmuerzo() ?></td>
<td bgcolor="#A9D0F5"><?php echo $registro->getHoraentalmuerzo(), '|', $registro-
>getMinentalmuerzo() ?></td>
<td bgcolor="#A9D0F5"><?php echo $registro->getHorasalida(), '|', $registro->getMinsalida()
?></td>

////////REPORTE H. EXTRAS ATRASOS//////////
<?php if ($registro->getResMinAtraso() <> 0) {
?>
<td bgcolor="#FA5858">

```

```

<?php echo $registro->getResMinAtraso(), 'min.', '(', (integer) ($registro->getResMinAtraso()
/ 60), 'h.', $registro->getResMinAtraso() % 60, 'min');
$atrasototal = $atrasototal + $registro->getResMinAtraso(); ?>
</td>
<td>
<?php echo "; ?>
</td>
<?php } elseif ($registro->getResMinExtra() <> 0) { ?>
<td>
<?php echo "; ?>
</td>
<td bgcolor="#F4FA58">
<?php
echo $registro->getResMinExtra(), 'min.', '(', (integer) ($registro->getResMinExtra() / 60), 'h.',
$registro->getResMinExtra() % 60, 'min');
$extratotal = $extratotal + $registro->getResMinExtra();
?></td>
<?php } else {
?>
<td bgcolor="#FF8000"><?php echo '<span style="color:#FFFFFF;">' . "Timbra Mal" .
'</span>'; ?></td>
<td bgcolor="#FF8000"><?php echo '<span style="color:#FFFFFF;">' . "Timbra Mal" .
'</span>'; ?></td>

<?php } ?>

```

////REPORTE H. TRABAJADAS////////////////////////////////////

```

<?php if ($registro->getResMinAtraso() == 0 && $registro->getResMinExtra() == 0) {
?>
<td bgcolor="#FF8000"><?php echo '<span style="color:#FFFFFF;">' . "Timbra Mal" .
'</span>'; ?>
</td>
<?php } else {
?>
<td bgcolor="#D0F5A9">

<?php
echo $registro->getResHT(), 'min. ', '(', (integer) ($registro->getResHT() / 60), 'h. ', $registro-
>getResHT() % 60, 'min. ');
$ht = $ht + $registro->getResHT();
$cont = $cont + 1
?>

</td>

```

//////PERMISO VACACIONES MOVILIZACIONES////////////////////////////////-->

```

<?php
if ($registro->getPermVacMov() <> 0) {
if ($registro->getPermVacMov() == 1) {
?><td bgcolor="#F6E3CE"><?php
echo 'permiso';
}

if ($registro->getPermVacMov() == 2) {
?><td bgcolor="#F6E3CE"><?php
echo 'Vacacion';
}
if ($registro->getPermVacMov() == 3) {
?><td bgcolor="#F6E3CE"><?php
echo 'Movilización';
}
} else {
?><td><?php
echo ";";
}
}

```

```

    }
    ?>
</td>
</tr>
</tbody>
<?php } else {
?>

<td bgcolor="#D7DF01" width="100%" colspan="10" align="center"><?php echo '
Código -> ', $registro->getEmpleadoPinempleado() ?>
<?php $datos = EmpleadoTable::getDatos($registro->getEmpleadoPinempleado())
?>
<?php echo ' Nombres -> ', $datos['nombres'], ' ', $datos['apellidos'] ?>
<?php
                $dep = $datos['departamento_coddepartamentos'];
                $sec = $datos['seccion_codseccion'];
                $datos = DepartamentoTable::getDatos($dep);
                $datos1 = SeccionTable::getDatos($sec)

                ?>
<?php echo ' Departamento -> ', $datos['descripcion'], ' Sección -> ',
$datos1['descripcion'] ?>

</td>

<?php
                $num = 0;
                $cont2 = $registro->getEmpleadoPinempleado();
            } endforeach;
        ?>

</table>

```

<!--////////// resultado de totales HE HA HT-->

```

<table>
<tr>

<th>Total Atraso</th>
<th>Total Extras</th>
<th>Horas Trab. </th>
<th>Total Horas Trabajadas</th>
<th>Num Dias</th>

</tr><td><?php echo $atrasototal, 'min.', ' ', (integer) ($atrasototal / 60), 'h.', $atrasototal % 60,
'min'); ?></td>
<td><?php echo $extratotal, 'min.', ' ', (integer) ($extratotal / 60), 'h.', $extratotal % 60, 'min'); ?></td>
<td><?php echo ' ', ($cont * 8), 'h.'; ?></td>
<td><?php echo $ht, 'min.', ' ', (integer) ($ht / 60), 'h.', $ht % 60, 'min'); ?></td>
<td><?php echo $num; ?></td>
</table>

```

```

<?php
    $grafatraso = abs((integer) ($atrasototal / 60));
    $grafextras = abs((integer) ($extratotal / 60));

```

//////////GENERACIÓN DE GRÁFICOS ESTADÍSTICOS

```

if ($atrasototal <> 0 || $extratotal <> 0) {

    $graph = new ezcGraphBarChart();
    $graph->palette = new tutorialCustomPalette();
    $graph->title = 'GRAFICO X HORAS';
    $graph->options->font->maxFontSize = 16;
    $graph->title->font->maxFontSize = 18;
    $graph->options->font->name = 'ARIAL BLACK';
    $graph->data[] = new ezcGraphArrayDataSet(

```



```

array(
    'Total H. Atrasos' => $grafatraso,
        'Total H. Extras' => $grafextras,
        'Total H. Normales' => $cont * 8,
        'Total H. Trabajadas' => (integer) ($ht / 60),
    )
);
$graph->data[]->highlight = true;

$graph->options->fillLines = 210;
$graph->options->highlightSize = 18;

$graph->render(300, 250, 'images/grafico.svg');
echo "<iframe src=\"'/images/grafico.svg\" width='300' height='250'></iframe>";
}

```

SCRIPT PARA CERRAR Y ABRIR RESULTADOS

```

</div>
<script type="text/javascript">

function mostrarDiv() {

div = document.getElementById('res');

        div.style.display = "";

    }

function cerrar() {

div = document.getElementById('res');

div.style.display='none';

    }

</script>
<div id="mostrarDiv"><a href="javascript:mostrarDiv();">Mostrar Resultado de la búsqueda</a></div>
<!--<div id="res" style="display:none;"> Este es un div oculto <br><a href="javascript:cerrar();"> presiona aquí para
cerrar</a></div-->

<!--//////////////////////////////////PAGINACIÓN////////////////////////////////// -->

<div class="pagination">

        </div>

```

ACTIONS.

Tenemos 3 funciones

EXECUTEINDEX: sirve para llamar a la página principal IndexSuccess

```
public function executeIndex(sfWebRequest $request) {
```

Se realiza una consulta a la tabla de empleados para utilizarlos en una lista desplegable de la búsqueda.

```

$this->empleados = Doctrine::getTable('empleado')
    ->createQuery('a')
    ->execute();

```

Se recoge los parámetros de las fechas y códigos de los empleados para la búsqueda.

```
if ($request->getParameter('pag') == null) {
    $pag = 24;
} else {
    $pag = $request->getParameter('pag');
}

if ($request->getParameter('pinc') == null) {
    $pinc = '';
    $pinc1 = '';
} else {
    $pinc = $request->getParameter('pinc');
    $pinc1 = $request->getParameter('pinc1');
}

if ($request->getParameter('query') == null && $request->getParameter('query1') == null) {
    $fechai = '%';
    $fechaf = '%';
} else {
    $fechai = $request->getParameter('query');
    $fechaf = $request->getParameter('query1');
}
```

Se envía los parámetros para reportarlos en pantalla en forma paginada.

```
$this->pager = new sfDoctrinePager('Registro', $pag);
$this->pager->setQuery(Doctrine::getTable('Registro')->getParameter($fechai, $fechaf, $pinc, $pinc1));
    $this->pager->setPage($request->getParameter('page', 1));
$this->pager->init();
}
```

EXECUTEREP: esta sirve para transformar la búsqueda de entrada y salida de empleado en pdf.

```
public function executeRep(sfWebRequest $request) {

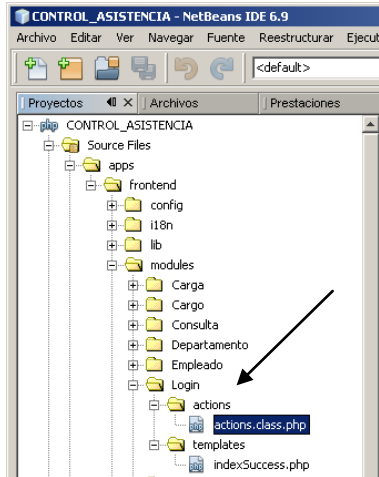
    // armando el reporte
    $pdf = new FPDF();
}
```

EXECUTEREPP: esta sirve para transformar la búsqueda de permisos, movilizaciones y vacaciones en pdf

```
public function executeRepp(sfWebRequest $request) {
    // armando el reporte
    $pdf = new FPDF();
}
```

SEGURIDAD

Para la verificación del login y contraseña se utilizó un módulo con los siguientes archivos:



Vista:

Index.php

Es la plantilla de la página principal en donde se grafica el formulario.

```
<div id="titulo">Identificación Usuario</div>
<br/>
<div id="texto">
<form action="<?php echo url_for('login/index')?>" method="POST">
<table align="center">
<?php echo $formulario?>
<tr>
<td colspan="2"><input type="Submit" value="Iniciar sesión"/></td>
</tr>
</table>

</form>

</div>
```

Actions

Aquí se encuentra la clase LoginActions y tiene 1 función donde se valida el ingreso del Usuario.

```
public function executeIndex(sfWebRequest $request) {
```

Se llama al formulario Identifica form que contiene 3 campos uno String, el siguiente tipo password y una lista desplegable para el tipo.

```
$this->formulario = new IdentificaForm();
    $parm = "";
```

Luego se genera una búsqueda de todos los registros de la tabla usuarios.

```
$q = Doctrine_Query::create()
    ->select('u.*')
    ->from('usuario u');
$users = $q->fetchArray();
```

Aquí se obtiene los datos en variables para luego comparar con los ingresados.

Obtiene datos ingresados.

```
if ($request->isMethod('post')) {
    $usuario = $request->getParameter("usuario");
    $contrasena = $request->getParameter("contrasena");
```

Busca en todos los usuarios el usuario que coincida con los valores buscados.

```
for ($i = 0; $i < count($users); $i++) {
    if ($usuario == $users[$i]['usuario'] && $contrasena == $users[$i]['contrasena'] && $users[$i]['tipo'] ==
        'ADMINISTRADOR') {
        $param = 'A';
    }

    if ($usuario == $users[$i]['usuario'] && $contrasena == $users[$i]['contrasena'] && $users[$i]['tipo'] ==
        'EMPLEADO') {
        $param = 'E';
    }
}

if ($param == 'A') {
    $this->getUser()->setAuthenticated(true);
    $this->redirect('empleado/index');
}

if ($param == 'E') {
    $this->getUser()->setAuthenticated(true);
    $this->redirect('consulta/index');
}

else{ echo '<script>alert("usuario o contraseña no válidos")</script>';}
}
```

Es importante configurar el archivo de configuración routing.yml que se encuentra en la carpeta config, para que la página siempre empiece consultando el login y password, así se introduzca la ruta para entrar sin digitar el usuario y contraseña

```
# default rules
homepage:
    url: /
    param: { module: login, action: index }
```

ANEXO NUM 5: Manual del Operador del Sistema Web - Control Asistencia-MAE

INTRODUCCIÓN.....	3
REQUERIMIENTOS TÉCNICOS	4
SUGERIDAS.....	4
INDISPENSABLES.....	4
CONFIGURACIÓN DEL NAVEGADOR	4
Mozilla FireFox.....	4
ACCESO A LA SESIÓN DEL OPERADOR	6
ADMINISTRACIÓN PARÁMETROS.....	7
DEPARTAMENTOS	7
SECCIÓN.....	8
TURNOS	9
CARGOS	10
USUARIOS	10
ADMINISTRACIÓN ASIGNACIÓN.....	11
EMPLEADOS.....	12
PERMISOS	13
MOVILIZACIONES	14
VACACIONES.....	15
ADMINISTRACIÓN CARGAR DATOS	16
INSTALACIÓN PROGRAMA WAD PARA	
DESCARGA DE DATOS DEL RELOJ BIOMÉTRICO	16
CARGAR	18
CALCULAR	18
ADMINISTRACIÓN, CÁLCULOS Y REPORTE.....	19
ENTRADAS Y SALIDA.....	19
CERRAR LA SESIÓN	22

SOLUCIÓN DE PROBLEMAS	23
PROBLEMAS CON EL PLUGIN DE JAVA	23
Síntomas.....	23
Prueba	23
Diagnóstico.....	23
Cómo resolver	23

INTRODUCCIÓN

Este manual está dirigido a los Operadores que trabajarán en conjunto con el sistema; resume toda la funcionalidad disponible para la correcta operación del mismo.

Cod.	Departamento	Seccion	Turno/Emplead	Cédula	Apellidos	Nombres	Telefono	Celular	Estado	Observacion	Editar	Borrar
101	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		FK	F			ACTIVO			
102	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		S	S			ACTIVO			
103	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		a	a			ACTIVO			
104	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		G	G			ACTIVO			
105	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		k	k			ACTIVO			
			NORMAL 8:00 A									

REQUERIMIENTOS TÉCNICOS.

Para el correcto funcionamiento de la Sesión de Operador del sistema es importante tomar en consideración las siguientes recomendaciones:

Sugeridas.

PC con procesador Pentium, 2000 MHz, 10024 MB en RAM. Windows /XP/WIN7

Indispensables.

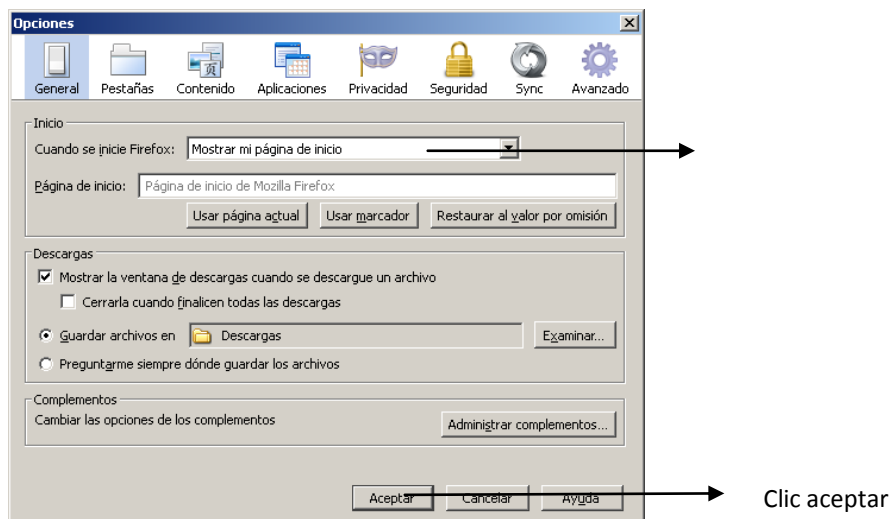
- Conexión a Internet
- Microsoft Internet Explorer 7.01+, Mozilla 3+, Google Chrome

Configuración del navegador.

Cada navegador consta de sus propias configuraciones. En ésta sección describiremos las configuraciones con las que deberá contar el navegador Mozilla FireFox.

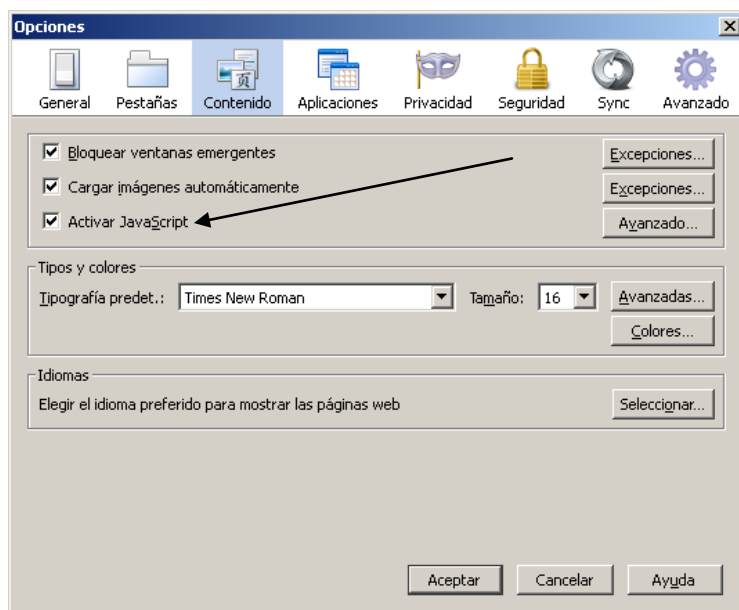
Es importante revisar algunas configuraciones de su navegador para garantizar la óptima funcionalidad del sistema. A continuación se incluye una descripción de las configuraciones básicas con las que deberá contar.

- 1.- Desde su navegador seleccione del Menú Herramientas la opción de 'Opciones de Internet'.



Funcionamiento Script del sistema

En la misma ventana seleccionamos contenido y debe estar activado Activar JavaScript, clic aceptar



ACCESO A LA SESIÓN DEL OPERADOR

Para acceder a la consola, el operador deberá seguir las siguientes instrucciones:

1. Abrir su Navegador. (Microsoft Internet Explorer 7+, Mozilla 3+, Chrome).
2. Entrar a la dirección de internet correspondiente a la instancia del sistema (www.asistenciamai.com).
3. En la página Usuario y contraseña digite su información para acceder si esta registrado como administrador:



Por último Enter o clic en Iniciar sesión.

Enseguida se carga la siguiente pantalla, por defecto se carga la administración de empleados, pero se puede escoger en el menú la opción para administrar.

Sistema de Control de Asistencia

Dirección Provincial del Ambiente de Imbabura

PARAMETROS ASIGNACION CARGAR CONSULTA Y REPORTE

LISTA DE EMPLEADOS

Cod.	Departamento	Seccion	Turno	Empleado	Cédula	Apellidos	Nombres	Telefono	Celular	Estado	Observacion	Editar	Borrar
101	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00		FK	F			ACTIVO			
102	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00		S	S			ACTIVO			
103	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00		a	a			ACTIVO			
104	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00		G	G			ACTIVO			
105	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00		k	k			ACTIVO			
106	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00		h	h			ACTIVO			

En el menú se encuentra botones que referencian a las secciones que tiene el sistema y dentro de estas opciones de cada sección como se indica en la imagen, solo falta poner el mouse sobre el menú para que aparezcan las opciones de cada sección.





ADMINISTRACIÓN DE PARÁMETROS

DEPARTAMENTOS

En esta sección el operador podrá consultar los departamentos existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier departamento que así se lo desee haciendo un

editar borrar

clic en estos botones  , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de diálogo que aparece. Si existen muchos registros se utiliza

 1 2 3

la paginación para poder observarlos, en cada página están 10 registros

26 Detalles - página 1/3



Coddepartamentos	Descripcion	Observaciones	Editar	Borrar
1	ADMINISTRATIVO FINANCIERO			
2	PATRIMONIO NATURAL			

Además se encuentra el botón “Nuevo” para agregar un departamento como se muestra en la Imagen. Y además se encuentra validado, si no se ingresa los campos requeridos. Por último tenemos el botón guardar.



NUEVO DEPARTAMENTO

• Debe ingresar un Departamento

Descripcion

Observaciones



Regresar Guardar

Al crear un nuevo registro aparece en la parte final de la lista.

SECCIÓN

En esta sección el operador podrá consultar las secciones existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier sección que así se lo desee haciendo un clic en estos



botones  , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación




para poder observarlos en cada página están 10 registros

Además se encuentra el botón “Nuevo” para agregar una sección como se muestra en la Imagen. Y además se encuentra validado si no se ingresa los campos requeridos, en el nombre del departamento solo hace falta escogerlo si ya se ingreso anteriormente. Por último tenemos el botón guardar.

Al crear un nuevo registro aparece en la parte final de la lista.

TURNO

Se puede editar y borrar cualquier dato del turno , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios.


Número	Descripción	Hora Entr	Min Entr	Hora Sal Alm	Minuto Sal Alm	Hora Entr Alm	Minuto Entr Alm	Hora Sal	Minuto Sal	Minutos Gracia	Editar
1	NORMAL	8	0	13	0	14	0	17	0	10	

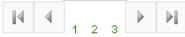
De igual manera se encuentra validado si alguno de estos parámetros no está ingresado.

EDITAR TURNO	
Descripción	NORMAL
Hora entrada	8
Min entrada	0
Hora salida muerzo	13
Min salida muerzo	• Ingrese Minuto Salida Almuerzo
Hora salida muerzo	14
Min salida muerzo	0
Hora salida	17
Min salida	0
Minutos gracia	10
Regresar	Guardar

CARGOS

En esta sección el operador podrá consultar los cargos existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier cargo que así se lo desee haciendo un clic en estos

botones , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación

para poder observarlos en cada página están 10 registros  26 Detalles - página 1/3



Además se encuentra el botón “Nuevo” para agregar un cargo como se muestra en la Imagen. Y además se encuentra validado si no se ingresa los campos requeridos. Por último tenemos el botón guardar.





Al crear un nuevo registro aparece en la parte final de la lista.

USUARIOS

En esta sección el operador podrá consultar los usuarios existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier usuario que así se lo desee haciendo un clic en estos



botones  , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación



para poder observarlos en cada página están 10 registros

26 Detalles - página 1/3

Sistema de Control de Asistencia
Dirección Provincial del Ambiente de Imbabura

PARAMETROS ASIGNACION CARGAR DATOS CONSULTA Y REPORTES

Departamentos Secciones Turnos Cargo **Usuarios**

LISTA USUARIOS

Idusuario	Usuario	Contraseña	Editar	Borrar	Tipo
1	admin	admin1	ADMINISTRADOR		
2	101	101	EMPLEADO		
7	102	102	EMPLEADO		

Nuevo

SERGIO BENITEZ

Además se encuentra el botón “Nuevo” para agregar un usuario como se muestra en la Imagen. Y además se encuentra validado si no se ingresa los campos requeridos. Por último tenemos el botón guardar.

Sistema de Control de Asistencia
Dirección Provincial del Ambiente de Imbabura

Nuevo Usuario

Usuario • Debe ingresar usuario

Contraseña • Debe ingresar contraseña

Tipo EMPLEADO

Regresar Guardar

SERGIO BENITEZ

Al crear un nuevo registro aparece en la parte final de la lista.

PARÁMETROS ASIGNACIÓN

EMPLEADOS

En esta sección el operador podrá consultar los empleados existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier empleado que así se lo desee haciendo un clic en estos



botones en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación



para poder observarlos en cada página están 10 registros



LISTA DE EMPLEADOS

Cod.	Departamento	Seccion	Turno	Emples	Cédula	Apellidos	Nombres	Telefono	Calular	Estado	Observacion	Editar	Borrar
101	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00 14:00 A 17:00		FK	F			ACTIVO			
102	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00 14:00 A 17:00		S	S			ACTIVO			
110	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL	8:00 A 13:00 14:00 A 17:00		TTTT	TTT			A			



Además se encuentra el botón “Nuevo” para agregar un empleado como se muestra en la imagen. Y además se encuentra validado si no se ingresa los campos requeridos, en el área de texto se puede expandir con el mouse desde la parte derecha inferior. Por último tenemos el botón guardar.



NUEVO EMPLEADO

PinEmpleado • Debe ingresar el Ping del Reloj Biométrico

Departamento coddepartamento ADMINISTRATIVO FINANCIERO

Seccion codseccion SECRETARIA

Cargos multicargo SERVIDOR PUBLICO DE APOYO 1

Turno numturno NORMAL

Cédula

Apellidos • Debe ingresar Apellidos empleado

Nombres • Debe ingresar Nombres empleado

Telefono

Calular

Estado • Debe ingresar Activo o Pasivo



Observacion

Regresar | **Guardar**

Al crear un nuevo registro aparece en la parte final de la lista.

PERMISOS

En esta sección el operador podrá consultar los permisos existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier permiso que así se lo desee haciendo un clic en estos

botones  , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación para poder observarlos en cada página están 10 registros



Num	Empleado	Fecha pedido	Fecha permiso	Fecha fin permiso	Num días	Asunto	Motivo	Num autoriza	Autoriza	Editar	Borrar
6	F FK	2011-01-03	2011-01-03	2011-01-04	1	Calamidad	DFGDFGDSFSDGDFGDSDFGD				
1	F FK	2011-01-07	2011-01-07	2011-01-10	3	Otros	ASD				
2	F FK	2011-01-11	2011-01-11	2011-01-12	1	Calamidad	ZXCV				
3	F FK	2011-01-19	2011-03-21	2011-03-22	1	Asunto de Trabajo	GDFGD				
6	F FK	2011-03-31	2011-03-31	2011-04-02	2	Asunto Personal	GSDFGDSFGDFG				
7	S S	2011-04-18	2011-04-18	2011-04-20	2	Calamidad	DSAFASDFSDAFSAD FS DF SDA FASDFSDAFSDFSADF				

Además se encuentra el botón “Nuevo” para agregar un permiso como se muestra en la Imagen. Y además se encuentra validado si no se ingresa los campos requeridos, en el área de texto se puede expandir con el mouse desde la parte derecha inferior. Por último tenemos el botón guardar.

Al crear un nuevo registro aparece en la parte final de la lista.

MOVILIZACIONES

En esta sección el operador podrá consultar las movilizaciones existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier movilización que así se lo desee haciendo un clic



en estos botones, en la edición al terminar de modificar el registro hará falta de hacer clic en "Guardar" para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación



para poder observarlos en cada página están 10 registros

26 Detalles - página 1/3

Sistema de Control de Asistencia
Dirección Provincial del Ambiente de Imbabura

PARAMETROS ASIGNACION CARGAR CONSULTA Y REPORTES

Empleados Permisos **Movilizaciones** Vacaciones

LISTA DE MOVILIZACIONES

Nro.	Empleado	FechaPedi.	FechaSal	FechaEnt.	Horas	Lugar	Asunto	Transp.	Autoriza	Editar	Borrar
1	F FK	2011-01-07	2011-01-07 8:00	2011-01-10 17:00	42	DSFSD	SDF				
3	S S	2011-01-18	2011-01-18 8:00	2011-01-19 8:00	8	Tulcan	QDFQSDQDFQSDQDFQSDQDF DFSQDFKGLNDKFLSLKNDQDF DFSQDSKFLQKDFLQKDFLQKDF DSFKQSDNLFKQNLKQSKAD WRWERQWERWERQWERQ WEQWERWERWERQ	Terrestre			
2	F FK	2011-03-31	2011-03-31 8:00	2011-04-01 8:00	8	Antonio Ante	FQHJFGHJFGG	Terrestre			

Nuevo

3 Detalles

Además se encuentra el botón "Nuevo" para agregar una movilización como se muestra en la Imagen. Y además se encuentra validado si no se ingresa los campos requeridos, en el área de texto se puede expandir con el mouse desde la parte derecha inferior. Por último tenemos el botón guardar.

Sistema de Control de Asistencia
Dirección Provincial del Ambiente de Imbabura

NUEVO PERMISO

Empleado (pinpleado) F

Fecha pedido • Debe ingresar fecha pedido

Fecha permiso • Debe ingresar la fecha de permiso

Fecha fin permiso • Debe ingresar la fecha de fin del permiso

Asunto Asunto de Trabajo

Motivo

Num. autoriza

Autoriza


Regresar Guardar

Al crear un nuevo registro aparece en la parte final de la lista.

VACACIONES

En esta sección el operador podrá consultar las vacaciones existentes ya que se despliega un listado de estos. Se puede editar y borrar cualquier vacación que así se lo desee haciendo un clic en estos



botones , en la edición al terminar de modificar el registro hará falta de hacer clic en “Guardar” para que se guarden los cambios y para eliminar se necesitará aceptar la eliminación haciendo clic en el cuadro de dialogo que aparece. Si existen muchos registros se utiliza la paginación



para poder observarlos en cada página están 10 registros

26 Detalles - pÁgina 1/3

Num	Empleado	FechaSalida	FechaIngreso	Diasv	Numautoriza	Autoriza	Editar	Borrar
1	F FK	2011-01-25	2011-03-28	63	4	DSF		

Además se encuentra el botón “Nuevo” para agregar una vacación como se muestra en la Imagen. Y además se encuentra validado si no se ingresa los campos requeridos, en el área de texto se puede expandir con el mouse desde la parte derecha inferior. Por último tenemos el botón guardar.

Nueva Vacación

Empleado pinEmpleado	F
FechaSalida	<input type="text"/>
FechaIngreso	<input type="text"/>
Numautoriza	<input type="text"/>
Autoriza	<input type="text"/>

Regresar Guardar

Al crear un nuevo registro aparece en la parte final de la lista.

ADMINISTRAR CARGA DE DATOS

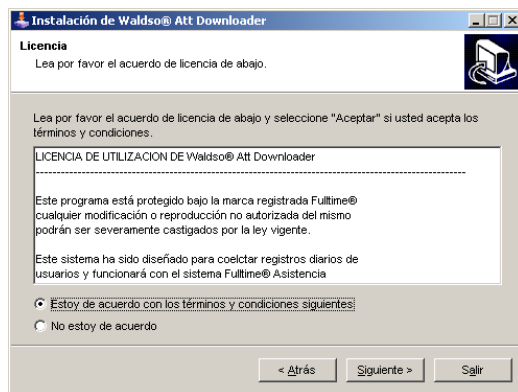
INSTALACIÓN PROGRAMA WAD PARA DESCARGA DE DATOS DEL RELOJ BIOMÉTRICO

Es importante recalcar que se debe instalar el programa WAD para descargar los datos del reloj biométrico al PC. En la siguiente imagen se muestra la forma de descargar el archivo.

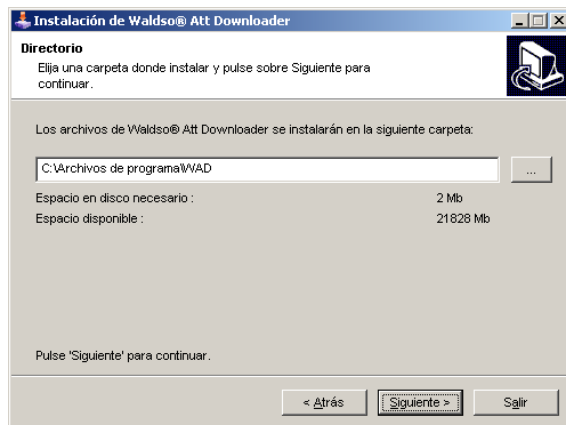
Instalación programa WAD, doble clic en instalarWAD.exe



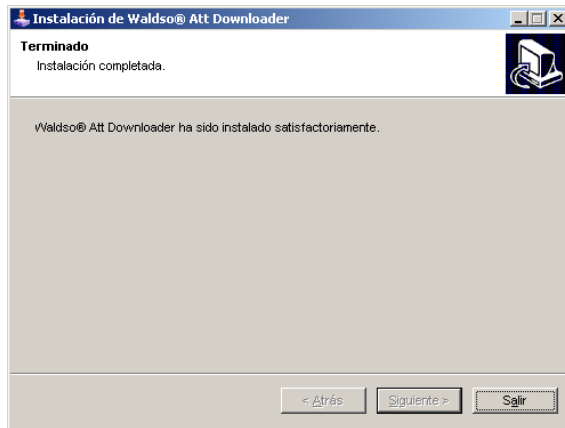
Siguiente,



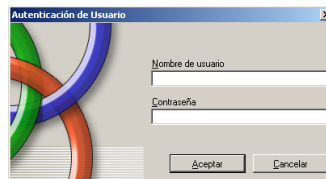
Estoy de acuerdo, siguiente



La ruta por defecto, siguiente

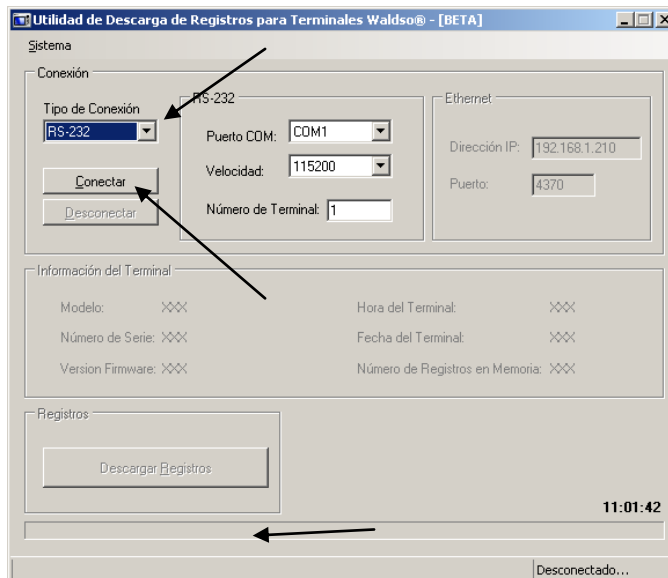


Clic Salir, luego tenemos un icono que aparece en el escritorio , lo abrimos con un doble clic.



Digitamos Nombre de Usuario y contraseña (Sólo al administrador se entrega este dato).

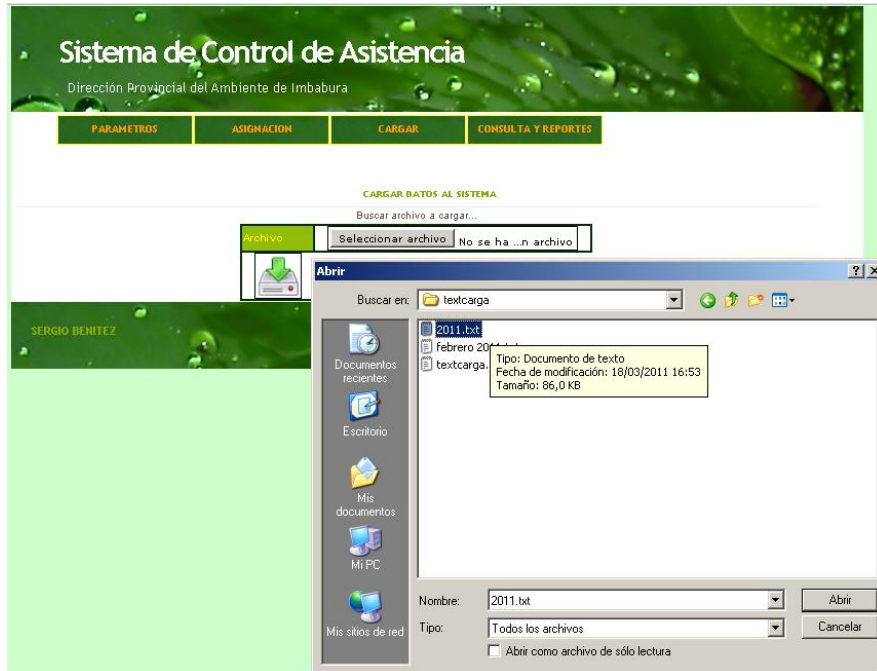
Se hace clic en conectar, antes comprobando que el tipo de conexión sea el de rs-232 para luego hacer clic en descargar archivos.




Por último comprobamos si se descargó el fichero en la ruta: C:\Archivos de programa\WAD\Log\2011-04-04.txt, este archivo toma el nombre de la fecha de descarga.

CARGAR DATOS

En esta sección se muestra un botón el cual abre un cuadro de diálogo en el cual se debe especificar la ruta del archivo el cual se encuentra en la ruta C:\Archivos de programa\WAD\Log\2011-04-04.txt, el cual debe contener el archivo que se ha descargado en ese momento del reloj biométrico.



Se abre el archivo, y luego clic en el botón  al final debe salir el mensaje de “Datos cargados correctamente”.

CALCULAR DATOS

Luego de haber cargado los datos se debe calcular para generar los atrasos, horas extras, horas trabajadas, y verificar permisos, movilizaciones y vacaciones.



Aquí solamente hay que limitar al rango que se ha cargado, es decir si se ha descargado solo de un mes se especifica el rango de la fecha de ese mes, luego presionamos calcular. Debe salir el mensaje “datos Calculados Correctamente”.

CONSULTAS Y REPORTE

Esta sección es la más utilizada del sistema que sirve para consultar los resultados de todos los procesos anteriores, la consulta de administrador por obvias razones tiene más opciones que la consulta de los empleados que se detalla en el manual de usuario.

Esta es la pantalla de consulta en donde vamos a ingresar los parámetros de búsqueda que son rango de fecha y rango de empleados:

Si deseamos podemos asignar el número de registros que aparecen en pantalla, en forma predeterminada es de 24.

A continuación hacemos clic en “buscar”.

Para observar los resultados debemos hacer clic en “mostrar resultado de la búsqueda”, para poder observar los resultados.

Aquí se detalla los ingresos y salidas que ha tenido el rango de empleados consultado en determinado intervalo de tiempo, así como los atrasos en minutos y horas y los extras que ha cumplido, además genera el número de horas trabajadas, y por último comprueba si en esta fechas el empleado ha tenido permisos, movilizaciones o vacaciones para ser tomados en cuenta.

Además verifica si el funcionario ha timbrado las 4 salidas, caso contrario genera en atrasos y extras un mensaje de que se ha timbrado mal.

Nos permite ver el gráfico estadístico de la tabla que se encuentra en pantalla haciendo fácil el control de la asistencia de los empleados.

Y la paginación si existen más registros que los predeterminados 24 registros, para poder observarlos se navega con los controles y se vuelve a hacer clic en mostrar resultados de la búsqueda.

Consulta de Asistencia

Ingrese parametros a buscar...

Entradas y salidas Perm. Mov. y Vac.

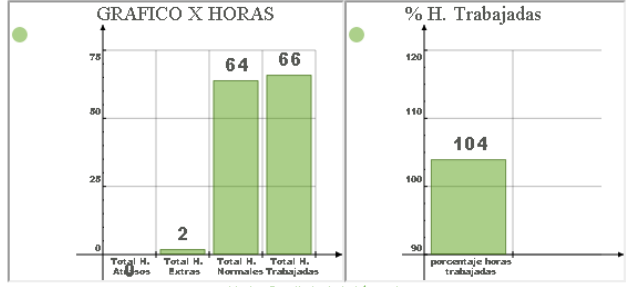
Fechas: 01/01/2011 inicio 31/01/2011 fin
 Nombres: FKF Inicio FKF fin
 Num Reg: Buscar

Cerrar Entradas Salidas

Fecha Ini.	Fecha Fin.	Código Empleado Ini.	Código Empleado Fin.
01/01/2011	31/01/2011	101	150

Num.	Código	Empleado	Area	Fecha	Hora Ent.	Hora Sal Alm.	Hora Ent. Alm.	Hora Sal.	Minatraso	Minextras	Horas Trab.	Perm Mov Vac
1	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	03/01/2011	8:3	0:0	14:14	17:14		14min. (0h.14min)	494min. (8h.14min.)	permiso
2	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	04/01/2011	8:7	13:23	14:3	15:57	63min. (1h.3min)		417min. (6h.57min.)	
3	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	05/01/2011	8:12	13:12	13:59	17:18		16min. (0h.16min)	496min. (8h.16min.)	
4	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	06/01/2011	8:12	13:15	13:55	17:40		38min. (0h.38min)	518min. (8h.38min.)	
5	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	07/01/2011	8:9	13:10	14:4	17:25		25min. (0h.25min)	505min. (8h.25min.)	Movilización
6	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	08/01/2011	8:18	13:6	13:55	15:2	110min. (1h.50min)		370min. (6h.10min.)	
7	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	10/01/2011	7:45	13:14	13:56	17:53		69min. (1h.9min)	548min. (9h.8min.)	
8	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	11/01/2011	8:11	13:13	13:57	17:27		26min. (0h.26min)	506min. (8h.26min.)	permiso
20	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	27/01/2011	8:0	13:21	14:7	17:29		0h.29min	506min. (8h.29min.)	
21	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	28/01/2011	8:4	13:13	14:1	0:0	Timbra Mal	Timbra Mal	Timbra Mal	
22	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	31/01/2011	7:51	13:11	14:10	17:49		58min. (0h.58min)	538min. (8h.58min.)	
23	102	S S	ADMINISTRATIVO FINANCIERO SECRETARIA	03/01/2011	7:51	13:15	13:58	17:28		37min. (0h.37min)	517min. (8h.37min.)	
24	102	S S	ADMINISTRATIVO FINANCIERO SECRETARIA	04/01/2011	7:54	13:23	14:6	17:30		36min. (0h.36min)	516min. (8h.36min.)	

Total Atraso	Total Extras	Horas Trab.	Total Horas Trabajadas	Num Dias
175min. (2h.55min)	525min. (8h.45min)	(168h.)	10430min. (173h.50min)	24



Mostrar Resultado de la búsqueda
 1 2 3 4 5



Por último tenemos los botones que nos permite exportar la consulta teniendo como resultado lo siguiente:



Archivo de entrada y salidas en pdf

DIRECCION PROVINCIAL DEL AMBIENTE DE IMABURA										
Sistema de Control de Asistencia										
Reporte Entradas/Salidas										
Cod.	Nombre	Departamento								
101	Fk F	ADMINISTRATIVO FINANCIERO								
Num.	Fecha	HoraEntrada	HoraSalAim	HoraEntAim	HoraSalida	Atrasos	Extras	H. Trab.	Observaciones	PMV
1	2011-01-03	8:3	0:0	14:14	17:14	0	14	8:14		Salida
2	2011-01-04	8:7	13:23	14:3	16:57	83	0	8:67		
3	2011-01-05	8:12	13:12	13:59	17:18	0	16	8:18		
4	2011-01-06	8:12	13:18	13:58	17:42	0	38	8:38		
5	2011-01-07	8:9	13:10	14:4	17:28	0	28	8:28		Mobilización
6	2011-01-08	8:18	13:8	13:58	18:2	110	0	8:10		
7	2011-01-10	7:48	13:14	13:58	17:53	0	88	8:8		Salida
8	2011-01-11	8:11	13:13	13:57	17:27	0	28	8:28		
9	2011-01-12	8:3	13:12	13:53	17:17	0	17	8:17		
10	2011-01-13	8:7	13:8	13:58	17:44	0	44	8:44		
11	2011-01-14	8:3	0:0	14:18	17:31	0	31	8:31		
12	2011-01-17	8:7	13:19	13:54	0:0	0	0		Ma. Troncos	Ma. Troncos
13	2011-01-18	8:8	13:14	0:0	16:58	2	0	7:58		
14	2011-01-19	7:57	13:8	14:8	0:0	0	3		Ma. Troncos	Ma. Troncos
15	2011-01-20	8:2	13:8	13:49	17:18	0	18	8:18		
16	2011-01-21	7:48	13:8	13:49	0:0	0	12		Ma. Troncos	Ma. Troncos
17	2011-01-24	8:7	13:18	13:53	17:11	0	11	8:11		
18	2011-01-25	8:7	13:14	13:54	17:44	0	44	8:44		Vacacion
19	2011-01-26	8:8	13:23	13:58	0:0	0	0		Ma. Troncos	Ma. Troncos
20	2011-01-27	8:0	13:21	14:7	17:28	0	28	8:28		
21	2011-01-28	8:4	13:13	14:1	0:0	0	0		Ma. Troncos	Ma. Troncos
22	2011-01-31	7:51	13:11	14:10	17:49	0	58	8:58		

Cod.	Nombre	Departamento								
102	S S	ADMINISTRATIVO FINANCIERO								
Num.	Fecha	HoraEntrada	HoraSalAim	HoraEntAim	HoraSalida	Atrasos	Extras	H. Trab.	Observaciones	PMV
1	2011-01-04	7:54	13:23	14:8	17:30	0	38	8:38		
2	2011-01-05	7:57	13:8	13:52	18:12	0	75	9:18		
3	2011-01-07	8:8	13:20	14:8	17:54	0	24	8:24		
4	2011-01-08	7:39	13:89	0:8	17:28	0	48	8:48		
5	2011-01-10	7:48	13:19	13:58	18:2	0	77	9:17		
6	2011-01-12	7:43	13:12	13:48	17:33	0	50	8:50		
7	2011-01-13	7:58	13:18	13:58	17:30	0	38	8:38		
8	2011-01-14	7:57	13:20	14:12	18:3	0	68	9:8		
9	2011-01-17	7:52	13:11	0:8	0:0	0	8		Ma. Troncos	Ma. Troncos
10	2011-01-18	7:48	13:8	13:48	17:28	0	38	8:38		Mobilización
11	2011-01-19	7:58	13:18	14:8	17:43	0	47	8:47		
12	2011-01-20	7:48	13:12	13:57	17:35	0	50	8:50		
13	2011-01-21	7:48	13:15	14:1	0:0	0	12		Ma. Troncos	Ma. Troncos
14	2011-01-24	7:33	0:0	0:0	0:0	0	27		Ma. Troncos	Ma. Troncos
15	2011-01-25	7:58	13:21	13:50	18:10	0	72	9:12		



Archivo de Permisos, movilizaciones y vacaciones en pdf

Como se nota tiene el mismo rango de empleados y fecha que se ha consultado.

DIRECCION PROVINCIAL DEL AMBIENTE DE IMABURA										
Sistema de Control de Asistencia										
Reporte Permisos Vacaciones y Movilizaciones										
Cod.	Nombre	Departamento								
101	Fk F	ADMINISTRATIVO FINANCIERO								
Num.	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias
1	permiso	2011-01-07	2011-01-07	2011-01-10		Otros	ASD			3
2	movilizacion	2011-01-07	2011-01-07	2011-01-10	DBFSO	SDF	DBF			42 horas
3	permiso	2011-01-07	2011-01-07	2011-01-10		Otros	ASD			3
4	vacacion		2011-01-28	2011-03-08					DBF	83

Cod.	Nombre	Departamento								
102	S S	ADMINISTRATIVO FINANCIERO								
Num.	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias
1	movilizacion	2011-01-07	2011-01-07	2011-01-10	DBFSO	SDF	DBF			42 horas

Cod.	Nombre	Departamento								
103	a a	ADMINISTRATIVO FINANCIERO								
Num.	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias

Cod.	Nombre	Departamento								
104	G G	ADMINISTRATIVO FINANCIERO								
Num.	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias

Cod.	Nombre	Departamento								
105	x x	ADMINISTRATIVO FINANCIERO								
Num.	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias

Cod.	Nombre	Departamento								
106	y y	ADMINISTRATIVO FINANCIERO								
Num.	tipo	Fecha Pedido	Fecha Salida	Fecha Regreso	Lugar	Asunto	Motivo	Transporte	Autoriza	Num. Dias

CERRAR SESIÓN

La sesión dura 10 minutos y luego automáticamente pasa a pedir nuevamente el usuario y contraseña para ingresar al sistema.

Si se desea cerrar la sesión, solo se cierra el navegador por completo es decir todas las páginas que se encuentren abiertas.

ANEXO NUM 6: Manual de Usuario del Sistema Web - Control Asistencia-MAE

INTRODUCCIÓN.....	2
PREREQUISITOS	
REGISTRO	2
CONFIGURACIÓN DEL NAVEGADOR	2
Mozilla FireFox.....	2
ACCESO A SECCIÓN DE USUARIO	3
CALCULOS Y REPORTE.....	4
ENTRADAS Y SALIDAS.....	4
CERRAR LA SESIÓN	6

INTRODUCCIÓN

Este manual está dirigido a los Usuarios que trabajarán en conjunto con el sistema; resume toda la funcionalidad disponible para la correcta operación del mismo.

PREREQUISITOS

REGISTRO

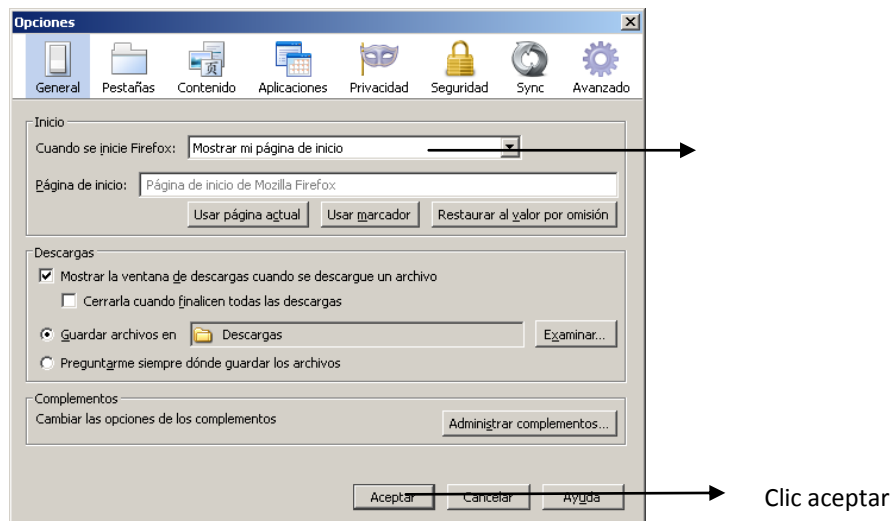
Debe acercarse a secretaria para que registren su usuario y contraseña para poder ingresar a consultar su asistencia.

CONFIGURACIÓN DEL NAVEGADOR

Cada navegador consta de sus propias configuraciones. En ésta sección describiremos las configuraciones con las que deberá contar el navegador Mozilla FireFox.

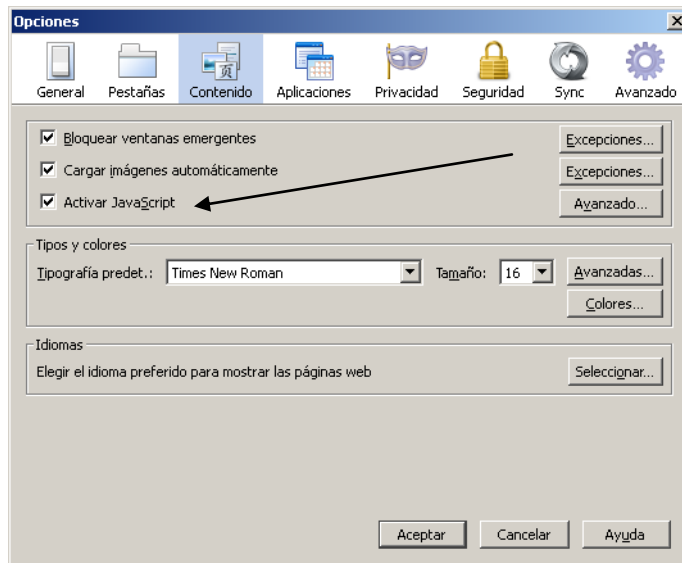
Es importante revisar algunas configuraciones de su navegador para garantizar la óptima funcionalidad del sistema. A continuación se incluye una descripción de las configuraciones básicas con las que deberá contar.

1.- Desde su navegador seleccione del Menú Herramientas la opción de 'Opciones de Internet'



Funcionamiento Script del sistema

En la misma ventana seleccionamos contenido y debe estar activado Activar JavaScript, clic aceptar



ACCESO A LA SESIÓN DE USUARIO

Para acceder a la consola, el usuario deberá seguir las siguientes instrucciones:

1. Abrir su Navegador. (Microsoft Internet Explorer 7+, Mozilla 3+, Chrome).
2. Entrar a la dirección de internet correspondiente a la instancia del sistema (www.asistenciamai.com).
3. En la página Usuario y contraseña digite su información para acceder si esta registrado como empleado:



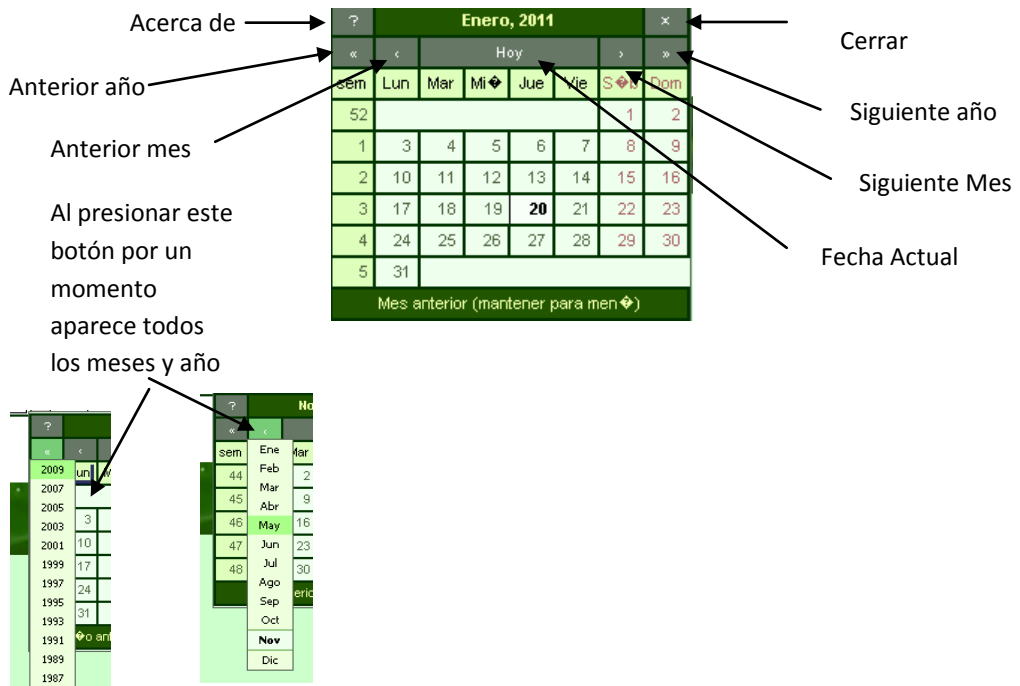
Por último Enter o clic en Iniciar sesión.

Enseguida se carga la siguiente pantalla:

CONSULTA Y REPORTE DEL SISTEMA



1. Hacer clic en fecha inicio... se despliegue una herramienta que permite seleccionar la fecha



2. Luego es necesario digitar el código que se le asigno para el reloj biométrico. Ver gráfico
3. Por último hacer clic en buscar. Ver gráfico
4. Para observar los resultados debemos hacer clic en “mostrar resultado de la búsqueda”, para poder observar los resultados, así:

Sistema de Control de Asistencia

Dirección Provincial del Ambiente de Imbabura

Consulta de Asistencia

Ingrese parametros

Num. Reg:

inicio fin Código

Cerrar Entradas Salidas

Fecha Ini.	Fecha Fin.
01/01/2011	31/01/2011

Num.	Código	Empleado	Area	Fecha	Hora Ent.	Hora Sal Alm.	Hora Ent Alm.	Hora Sal.	Min. atraso	Min. extras	Horas Trab.	Permi/Mov/Vac
1	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	03/01/2011	8:3	0:0	14:14	17:14		14min. (0h.14min)	494min. (8h.14min.)	permiso
2	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	04/01/2011	8:7	13:23	14:3	15:57	63min. (1h.3min)		417min. (6h.57min.)	
3	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	05/01/2011	8:12	13:12	13:59	17:18		16min. (0h.16min)	496min. (8h.16min.)	
4	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	06/01/2011	8:12	13:15	13:55	17:40		38min. (0h.38min)	518min. (8h.38min.)	
5	101	F FK	ADMINISTRATIVO FINANCIERO SECRETARIA	07/01/2011	8:9	13:10	14:4	17:25		25min. (0h.25min)	505min. (8h.25min.)	Movilización

Total Atrazo	Total Extras	Horas Trab.	Total Horas Trabajadas	Num Dias
63min. (1h.3min)	93min. (1h.33min)	(40h.)	2430min. (40h.30min)	5

GRAFICO X HORAS

% H. Trabajadas

Mostrar Resultado de la búsqueda

22 Detalles - página 1/5

Aquí se detalla los ingresos y salidas que ha tenido el empleado consultado en determinado intervalo de tiempo, así como los atrasos en minutos y horas y los extras que ha cumplido, además genera el número de horas trabajadas, y por último comprueba si en estas fechas el empleado ha tenido permisos, movilizaciones o vacaciones para ser tomados en cuenta.

Además verifica si el funcionario ha timbrado las 4 salidas, caso contrario genera en atrasos y extras un mensaje de que se ha timbrado mal.

Nos permite ver el gráfico estadístico de la tabla que se encuentra en pantalla haciendo fácil el control de la asistencia de los empleados.

Y la paginación si existen más registros que los predeterminados, 24 registros, para poder observarlos se navega con los controles y se vuelve a hacer clic en mostrar resultados de la búsqueda.

Ir a la página inicio de los registros

Ir a la página final de los registros

Ir a la página anterior

Número de páginas

Ir a la página siguiente

CERRAR SESIÓN

La sesión dura 10 minutos y luego automáticamente pasa a pedir nuevamente el usuario y contraseña para ingresar al sistema.

Si se desea cerrar la sesión solo se cierra el navegador por completo es decir todas las páginas que se encuentren abiertas.

ANEXO NUM 7: Construcción del Cable.

La idea es conectar el reloj biométrico al computador mediante el puerto rs-232, para comenzar es necesario ver una imagen del cable:



2. veamos los implementos para construirla.



4 conectores rs-232 hembras.

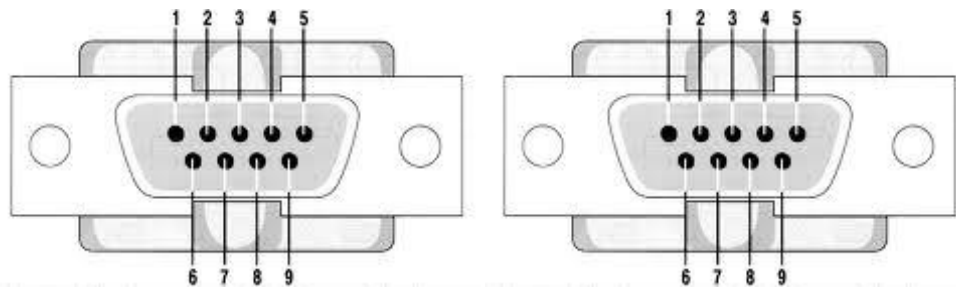


30 metros de cable de 4 hilos.

1 implemento para soldar.



3. Vista de la forma que va distribuido la conexión



Reloj biométrico

PC

2--- Recibe Datos

2—Recibe Datos

3—Trasmite datos

3—Transmite Datos

5--(GND) masa

5--(GND) masa

4. Comprobación de continuidad con un testeado para estar seguros de su eficiencia.



ANEXO NUM 8: Actas de Trabajo

ACTA DE TRABAJO No. 1		
Proyecto: Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.		
Tema a tratar: Levantamiento informal de requerimientos funcionales del sistema a implantar.		
Fecha: 06/agosto/2010		
Participantes:		
Nombre	Cargo	Firma
Ing. Segundo Fuentes Cáceres	Director	
Lcda. Mónica Maza	Sistemas	
Lcda. Mónica Maza	Secretaria	
Observaciones:		
<p>Se ha establecido la implementación de un sistema WEB, que permita mejorar los procesos de control y manejo de la asistencia de personal de la Dirección Provincial del Ambiente de Imbabura.</p> <p>Se tienen varios requerimientos que deberá cumplir este nuevo sistema:</p> <ul style="list-style-type: none">• Debe integrarse los datos contenidos en los relojes biométricos de Ibarra y Cotacachi en la base de datos del Sistema.• La descarga de los datos se realizará por medio de cable con conectores rs232, desde las secretarías de cada cantón.• La administración del Sistema se lo realizará vía Web.• Es necesario dar seguridad media a estos datos (Usuarios).• El sistema realizará los reportes según lo acordado con las autoridades de la institución.• Los reportes se podrán exportar a archivos de Excel.• Los reportes en pantalla deberán tener un grafico estadístico para su visualización.• Cada año deberá guardar un histórico del año anterior.		
Compromisos adquiridos:		
- El Departamento de Sistemas de la Dirección Provincial del Ambiente de Imbabura. debe realizar el levantamiento del documento de Visión para este proyecto de software.		

ACTA DE TRABAJO No. 2

Proyecto: Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.

Módulo 1: Parámetros

Tema a tratar: Levantamiento informal de requerimientos funcionales del módulo a implantar.

Fecha: 20/agosto/2010

Participantes:

Nombre	Cargo	Firma
Ing. Segundo Fuentes Cáceres	Director	
Lcda. Mónica Maza	Contabilidad	
Lcda. Mónica Maza	Secretaria	

Observaciones:

Se procede a explicar la estructura del módulo de parámetros, para la configuración del Sistema

- Esta es la presentación de la estructura y datos a contener en este módulo.

ESQUEMAS EN DOCUMENTO ORIGINAL

Compromisos adquiridos:

- La Institución se compromete a:
 - Brindar la información de los empleados y la instrucción para llenar la base de datos con datos reales.
 - También los datos que genera el reloj biométrico para realizar pruebas de cálculo y llenado de la base de datos en la tabla respectiva

ANEXO NUM 9: Documentación de la Etapa de Pruebas.

Especificación de Casos de Prueba

Base de datos de prueba

Departamentos

Coddepartamentos	Descripcion	Observaciones
1	ADMINISTRATIVO FINANCIERO	
4	CALIDAD AMBIENTAL	
2	PATRIMONIO NATURAL	
6	SEGUIMIENTO Y EVALUACION	
5	UNIDAD JURÍDICA	

Sección

Código Sección	Departamentos	Sección
7	ADMINISTRATIVO FINANCIERO	ASEO
12	CALIDAD AMBIENTAL	CALIDAD AMBIENTAL
3	ADMINISTRATIVO FINANCIERO	CONTABILIDAD
10	PATRIMONIO NATURAL	FORESTAL
4	ADMINISTRATIVO FINANCIERO	GUARDALMACEN
14	UNIDAD JURÍDICA	JURÍDICO
15	PATRIMONIO NATURAL	RESPONSABLE UNIDAD
8	ADMINISTRATIVO FINANCIERO	RESPONSABLE UNIDAD
1	ADMINISTRATIVO FINANCIERO	SECRETARIA
9	ADMINISTRATIVO FINANCIERO	SECRETARIA AUX

Turno

Número	Descripcion	HoraEnt	MinEnt	HoraSal	MinutoSal	HoraEntA	MinutoEntA	HoraSal	MinutoSal	MinutosGracia
1	NORMAL	8	0	13	0	14	0	17	0	10

Cargo

Numcargo	Descripcion	Observaciones
1	SERVIDOR PÚBLICO DE APOYO 1	

Usuarios

Idusuario	Usuario	Contrasena	Editar
1	admin	admin1	ADMINISTRADOR
2	101	101	EMPLEADO
7	102	102	EMPLEADO

Empleados

Cod.	Departamento	Sección	TurnoEmplead	Cédula	Apellidos	Nombres	Teléfono	Estado
101	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		MAZA RIVADENEIRA	MONICA LUCIA		ACTIVO
102	ADMINISTRATIVO FINANCIERO	GUARDALMACEN	NORMAL 8:00 A 13:00 14:00 A 17:00		ENCALADA BUITRON	HUGO BENEDICTO		ACTIVO
103	ADMINISTRATIVO FINANCIERO	RESPONSABLE UNIDAD	NORMAL 8:00 A 13:00 14:00 A 17:00		ESTRADA AVILES	JULIO CESAR		ACTIVO
104	CALIDAD AMBIENTAL	CALIDAD AMBIENTAL	NORMAL 8:00 A 13:00 14:00 A 17:00		ALDAS	JESSIKA		ACTIVO
105	PATRIMONIO NATURAL	VIDA SILVESTRE	NORMAL 8:00 A 13:00 14:00 A 17:00		PANTOJA MONTALVO	SEGUNDO MARCELO		ACTIVO
106	ADMINISTRATIVO FINANCIERO	SECRETARIA AUX	NORMAL 8:00 A 13:00 14:00 A 17:00		RAMIREZ ALBERCA	HISABEL RICARD		ACTIVO
107	ADMINISTRATIVO FINANCIERO	TESORERIA	NORMAL 8:00 A 13:00 14:00 A 17:00		VOSMEDIANO MANOSALVAS	EDUARDO		ACTIVO
108	PATRIMONIO NATURAL	FORESTAL	NORMAL 8:00 A 13:00 14:00 A 17:00		ROSETO MIER	MANUEL		ACTIVO
109	PATRIMONIO NATURAL	VIDA SILVESTRE	NORMAL 8:00 A 13:00 14:00 A 17:00		ROSALES RODRIGUEZ	CARLOS MANUEL		ACTIVO
110	SEGUIMIENTO Y EVALUACION	SEGUIMIENTO Y EVALUACION	NORMAL 8:00 A 13:00 14:00 A 17:00		CHALA OGONAGA	JENY MARIBEL		ACTIVO

Permisos

Num	Empleado	FechaPermiso	FechaPermiso	FechaFinPermiso	Numdias	Asunto	Motivo	Numautoriza	Autoriza
6	MONICA LUCIA MAZA RIVADENEIRA	2011-01-03	2011-01-03	2011-01-04	1	Calamidad	DFGDFGDSFGDSFGDFGSDFG		
1	MONICA LUCIA MAZA RIVADENEIRA	2011-01-07	2011-01-07	2011-01-10	3	Otros	ASD		
2	MONICA LUCIA MAZA RIVADENEIRA	2011-01-11	2011-01-11	2011-01-12	1	Calamidad	ZXCV		
3	MONICA LUCIA MAZA RIVADENEIRA	2011-01-19	2011-03-21	2011-03-22	1	Asunto de Trabajo	GDFGD		
5	MONICA LUCIA MAZA RIVADENEIRA	2011-03-31	2011-03-31	2011-04-02	2	Asunto Personal	GSDFGDSFGDFG		
7	HUGO BENEDICTO ENCALADA BUITRON	2011-04-18	2011-04-18	2011-04-20	2	Calamidad	DSAFASDFSDAFSAD FS DF SDA FASDFSDAFSDFSDAF		

Movilizaciones

Nro.	Empleado	FechaPed.	FechaSal.	FechaEnt.	Horas	Lugar	Asunto	Transp.	Autoriza
1	MONICA LUCIA MAZA RIVADENEIRA	2011-01-07	2011-01-07 8:00	2011-01-10 17:00	42	DSFSD	SDF		
3	HUGO BENEDICTO ENCALADA BUITRON	2011-01-18	2011-01-18 8:00	2011-01-19 8:00	8	Tulcán	GDFGSDGDFGDSFGDFGSDFGDFGDFGDFKLNDFSLNKGNDFGDFSGDSKFLGKDNFLSKGDSFKGSDNLFKGNLKDSDKADWRWERQWERWERQWERQWERQWERQWERQ	Terrestre	
2	MONICA LUCIA MAZA RIVADENEIRA	2011-03-31	2011-03-31 8:00	2011-04-01 8:00	8	Antonio Ante	FGHJFGHJFGG	Terrestre	

Vacaciones

Num	Empleado	FechaSalida	FechaIngreso	DiasV	Numautoriza	Autoriza
1	MONICA LUCIA MAZA RIVADENEIRA	2011-01-25	2011-03-28	63	4	DSF

Caso de prueba de Administrar Parámetros.

Historial de Revisiones.

Fecha	Versión	Descripción	Autor
8/09/2010	0.5	Versión preliminar a falta de aprobación del Stakeholder. ⁶	Sergio Benítez
14/11/2010	1.0	Versión revisada por el Stakeholder.	Sergio Benítez
09/01/2011	2.0	Versión lista para pasar el test.	Sergio Benítez
15/01/2011	2.9	Versión con evaluación de las pruebas actualizado.	Sergio Benítez
15/02/2011	3.0	Versión preparada para la segunda iteración de la fase de construcción.	Sergio Benítez

Descripción.

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Caso de prueba de Administrar Parámetros”. Las pruebas que se puede realizar son la de crear, editar, borrar y listar. El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

Comprobar que la administración de parámetros funciona correctamente.

Descripción.

Nos introducimos en el sistema como administrador del sistema, accediendo a su funcionalidad y crearemos departamentos, secciones, cargos y usuarios, luego editaremos y borrarremos y por ultimo listaremos los resultados de las pruebas que se realice. El parámetro turno solo se probará la modificación de los datos y guardar.

Condiciones de ejecución.

Las condiciones de ejecución del caso de prueba son que el usuario administrador del sistema “admin” está dado de alta en la base de datos y su clave

⁶Stakeholder: quienes pueden afectar o son afectados por las actividades de una empresa.

correspondiente. Administrar la Base de Datos de Pruebas para ver toda la especificación completa de los datos.

Entrada.

- Introducimos 'admin' en el campo usuario
- Introducimos 'admin1' en el campo contraseña
- Pulsamos entrar o el botón "ingresar" de la aplicación.
- Nos aparece la interfaz propia del administrador del sistema, donde pulsaremos la pestaña de "Parámetros".
- Seleccionamos uno de las opciones que deseamos administrar.
- Para crear un nuevo parámetro pulsamos el botón "nuevo".
- Para la edición sol hay que pulsar en la figura de edición "lápiz" de cada registro.
- Y si deseamos eliminar el la figura de borrar "X" de cada registro.
- Siempre al entrar nos muestra el listado de todos los registros que se encuentran creados en el parámetro.
- Para salir solo cerramos el navegador.

Resultado esperado.

El sistema nos muestra una interfaz en donde se lista todos los registros de los parámetros y en donde se puede administrar los registros.

Evaluación de la Prueba.

Prueba superada con éxito.

Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones.

Historial de Revisiones.

Fecha	Versión	Descripción	Autor
8/09/2010	0.5	Versión preliminar a falta de aprobación del Stakeholder.	Sergio Benítez
14/11/2010	1.0	Versión revisada por el Stakeholder.	Sergio Benítez
09/01/2011	2.0	Versión lista para pasar el test.	Sergio Benítez
15/01/2011	2.9	Versión con evaluación de las pruebas actualizado.	Sergio Benítez
15/02/2011	3.0	Versión preparada para la segunda iteración de la fase de construcción.	Sergio Benítez

Descripción.

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso "Caso de prueba de Crear Empleados y asignar Permisos, Movilizaciones y Vacaciones". Las pruebas que se puede realizar son la de crear, editar, borrar y listar. El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

Comprobar que la administración de asignación funciona correctamente.

Descripción.

Nos introducimos en el sistema como administrador del sistema, accediendo a su funcionalidad y crearemos empleados, permisos, movilizaciones y vacaciones del empleado, luego editaremos y borrarémos y por último listaremos los resultados de las pruebas que se realice. El parámetro turno solo se probará la modificación de los datos y guardar.

Condiciones de ejecución.

Las condiciones de ejecución del caso de prueba son que el usuario administrador del sistema "admin" está dado de alta en la base de datos y su clave

correspondiente. Administrar la Base de Datos de Pruebas para ver toda la especificación completa de los datos.

Entrada.

- Introducimos 'admin' en el campo usuario
- Introducimos 'admin1' en el campo contraseña
- Pulsamos entrar o el botón "ingresar" de la aplicación.
- Nos aparece la interfaz propia del administrador del sistema, donde pulsaremos la pestaña de "ASIGNACIÓN".
- Seleccionamos uno de las opciones que deseamos administrar.
- Para crear un nuevo parámetro pulsamos el botón "nuevo".
- Para la edición sol hay que pulsar en la figura de edición "lápiz" de cada registro.
- Y si deseamos eliminar el la figura de borrar "X" de cada registro.
- Siempre al entrar nos muestra el listado de todos los registros que se encuentran creados en el parámetro.
- Para salir solo cerramos el navegador.

Resultado esperado.

El sistema nos muestra una interfaz en donde se lista todos los registros de los parámetros y en donde se puede administrar los registros.

Evaluación de la Prueba.

Prueba superada con éxito.

Cargar y calcular Datos generados del reloj biométrico hacia el sistema.

Historial de Revisiones.

Fecha	Versión	Descripción	Autor
28/09/2010	0.5	Versión preliminar a falta de aprobación del Stakeholder.	Sergio Benítez
20/01/2010	1.0	Versión revisada por el Stakeholder.	Sergio Benítez
09/02/2011	2.0	Versión lista para pasar el test.	Sergio Benítez
15/02/2011	2.2	Versión con diversas modificaciones no revisadas por el Stakeholder.	Sergio Benítez
20/02/2011	2.5	Versión con detalles y ajustes a la aplicación en desarrollo.	Sergio Benítez
3/03/2011	2.9	Versión con evaluación de las pruebas actualizado.	Sergio Benítez
15/03/2011	3.0	Versión preparada para la segunda iteración de la fase de construcción.	Sergio Benítez

Descripción.

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Cargar y calcular Datos generados del reloj biométrico hacia el sistema”. Las pruebas a realizar son cargar datos del fichero de texto a la base de datos y calcular los datos ingresados para generar los minutos de atrasos extras horas trabajadas y verificación de permisos movilizaciones y vacaciones de los empleados. El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

1) Comprobar que la carga de datos al sistema funciona correctamente.


Descripción.

Nos introducimos en el sistema como administrador del sistema, accediendo a su funcionalidad y cargaremos el archivo de texto a la base de datos.

Condiciones de ejecución.

Las condiciones de ejecución del caso de prueba son que el usuario administrador del sistema "admin" está dado de alta en la base de datos y su clave correspondiente. Administrar la Base de Datos de Pruebas para ver toda la especificación completa de los datos.

Entrada.

- Introducimos 'admin' en el campo usuario
- Introducimos 'admin1' en el campo contraseña
- Pulsamos entrar o el botón "ingresar" de la aplicación.
- Nos aparece la interfaz propia del administrador del sistema, donde pulsaremos la pestaña de "CARGAR".
- Seleccionamos "carga archivo".
- Buscamos la ubicación del archivo de texto a cargar y lo seleccionamos con un clic.
- Hacemos clic en "abrir".
- Hacemos clic en el botón procesar -> 
- Y esperamos el mensaje de "Datos ingresados correctamente".
- Luego al hacer clic en aceptar nos re direcciona a "calcular datos".

Resultado esperado.

El sistema nos muestra un mensaje de "Datos ingresados correctamente".

Evaluación de la Prueba.

Mensaje de Texto que genera el mensaje esperado.

Prueba superada con éxito.

2) Comprobar que la opción de cálculos de datos funciona correctamente.

Descripción.

Nos introducimos en el sistema como administrador del sistema, accediendo a su funcionalidad y calcularemos con la información cargada mediante un rango de fechas.

Condiciones de ejecución.

Las condiciones de ejecución del caso de prueba son que el usuario administrador del sistema "admin" está dado de alta en la base de datos y su clave correspondiente. Administrar la Base de Datos de Pruebas para ver toda la especificación completa de los datos.

También es necesario que se haya realizado la carga de los datos del rango seleccionado.

Entrada.

- Introducimos 'admin' en el campo usuario
- Introducimos 'admin1' en el campo contraseña
- Pulsamos entrar o el botón "ingresar" de la aplicación.
- Nos aparece la interfaz propia del administrador del sistema, donde pulsaremos la pestaña de "Calcular atrasos".
- Seleccionamos un rango de fechas inicial y final para realizar el cálculo.
- Hacemos clic en "calcular".
- Luego al hacer clic en aceptar nos re direcciona a "consultas y reportes".

Resultado esperado.

El sistema nos muestra un mensaje de "Datos calculados correctamente".

Evaluación de la Prueba.

Mensaje de Texto que genera el mensaje esperado.

Prueba superada con éxito.

Consultar y Reportar la asistencia del personal.

Historial de Revisiones.

Fecha	Versión	Descripción	Autor
28/09/2010	0.5	Versión preliminar a falta de aprobación del Stakeholder.	Sergio Benítez
20/01/2010	1.0	Versión revisada por el Stakeholder.	Sergio Benítez
09/02/2011	2.0	Versión lista para pasar el test.	Sergio Benítez
30/02/2011	2.2	Versión con diversas modificaciones no revisadas por el Stakeholder.	Sergio Benítez
15/03/2011	2.5	Versión con detalles y ajustes a la aplicación en desarrollo.	Sergio Benítez
20/03/2011	2.9	Versión con evaluación de las pruebas actualizado.	Sergio Benítez
15/04/2011	3.0	Versión preparada para la segunda iteración de la fase de construcción.	Sergio Benítez

Descripción.

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Consultar y Reportar la asistencia del personal”. Las pruebas a realizar:

- Consultar por medio de un rango de fechas y de nombres de empleados las entradas y salidas, minutos de atrasos, extras, trabajadas y si existen permisos, movilizaciones y vacaciones de los empleados.
- Graficar en diagrama de barras los datos obtenidos.
- Exportar a pdf los resultados de la consulta.
- Exportar a pdf los resultados de los permisos, vacaciones y movilizaciones con sus detalles.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

1) Comprobar que la consulta y despliegue de los datos funciona correctamente.

Descripción.

Nos introducimos en el sistema como administrador del sistema, accediendo a su

funcionalidad e ingresaremos a realizar la consulta mediante un rango de fechas y de empleados para que se despliegue la información requerida.

Condiciones de ejecución.

Las condiciones de ejecución del caso de prueba son que el usuario administrador del sistema "admin" está dado de alta en la base de datos y su clave correspondiente. Administrar la Base de Datos de Pruebas para ver toda la especificación completa de los datos.

También es necesario que se haya realizado la carga de los datos del rango seleccionado.

Entrada.

- Introducimos 'admin' en el campo usuario
- Introducimos 'admin1' en el campo contraseña
- Pulsamos entrar o el botón "ingresar" de la aplicación.
- Nos aparece la interfaz propia del administrador del sistema, donde pulsaremos la pestaña de "Consulta y reportes".
- Seleccionamos la opción "entradas y salidas".
- Seleccionamos un rango de fechas inicial y final y de empleados para realizar la consulta.
- Si es necesario asignamos el número de registros a presentar en pantalla
- Hacemos clic en "Buscar".
- Por último seleccionamos "Mostrar Resultado de la búsqueda", para desplegar el resultado, en la parte inferior se encuentra el gráfico estadístico.
- Para salir solo cerramos el navegador.

Resultado esperado.

El sistema nos muestra una lista con:

Nombre del empleado, Departamento y sección:

Número, Fecha, Hora entrada, HoraSalAlm, HoraEntAlm, Hora Salida, Atrasos(Min.), Extras(Min.), Horas Trabajadas(Hora), Observaciones si hay permisos vacaciones o movilizaciones.

Gráfico estadístico de la consulta realizada, que contiene total minutos atrasos, total minutos extras, total horas trabajadas y porcentaje de trabajo con referencia a lo normal.

Evaluación de la Prueba.

La consulta se realizada con éxito.

Los gráficos se realizan con éxito.

Los totales se realizan con éxito.

Prueba superada con éxito.

2) Comprobar que el reporte en pdf de los datos consultados funciona correctamente

Descripción.

Nos introducimos en el sistema como administrador del sistema, accediendo a su funcionalidad e ingresaremos a realizar la consulta mediante un rango de fechas y de empleados para que se despliegue la información requerida en formato pdf.

Condiciones de ejecución

Las condiciones de ejecución del caso de prueba son que el usuario administrador del sistema "admin" está dado de alta en la base de datos y su clave correspondiente. Administrar la Base de Datos de Pruebas para ver toda la especificación completa de los datos.

También es necesario que se haya realizado la carga de los datos del rango seleccionado.

Y que se haya realizado la consulta con los rangos seleccionados de fecha y empleado

Entrada.

- Introducimos 'admin' en el campo usuario
- Introducimos 'admin1' en el campo contraseña
- Pulsamos entrar o el botón "ingresar" de la aplicación.
- Nos aparece la interfaz propia del administrador del sistema, donde pulsaremos la pestaña de "Consulta y reportes".
- Seleccionamos la opción "entradas y salidas".
- Seleccionamos un rango de fechas inicial y final y de empleados para realizar la consulta.
- Si es necesario asignamos el número de registros a presentar en pantalla
- Hacemos clic en "Buscar".
- Hacemos clic en "Entradas y salidas"
- O hacemos clic en "Perm. Mov. Y Vac".
- Para salir solo cerramos el navegador.

Resultado esperado.

El sistema nos muestra una lista en pdf con:

Entradas y salidas:

Nombre del empleado, Departamento y sección:

Número, Fecha, Hora entrada, HoraSalAlm, HoraEntAlm, Hora Salida, Atrasos(Min.), Extras(Min.), Horas Trabajadas(Hora), Observaciones si hay permisos vacaciones o movilizaciones.

Perm. Mov. Y Vac:

Nombre del empleado, Departamento y sección:

Número, tipo, Fecha Pedido, Fecha Salida, Fecha Regreso, Lugar, Asunto, Motivo, Transporte, Autoriza Núm. Dias.

Evaluación de la Prueba.

La generación del pdf se realiza con éxito.

Los totales se realizan con éxito.

Prueba superada con éxito.

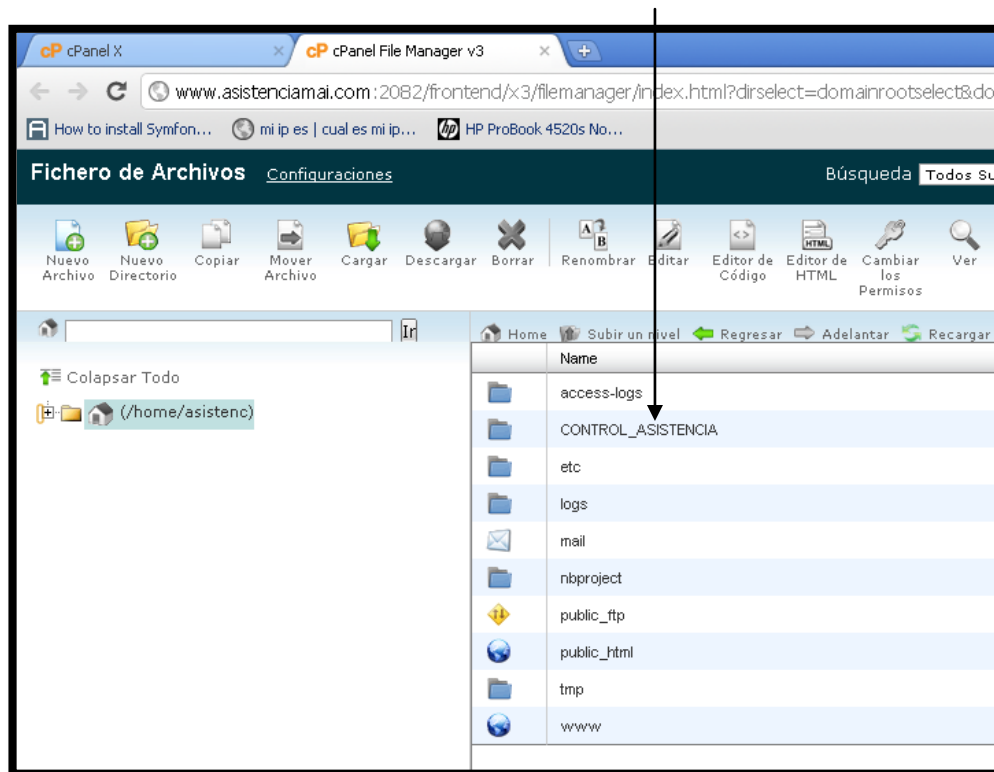
Informe de Instalación.

Migración del servidor local al hosting.

Existe una incompatibilidad con el servidor Hosting, por causa de algunas configuraciones del motor de la base de datos y el descriptor de php.

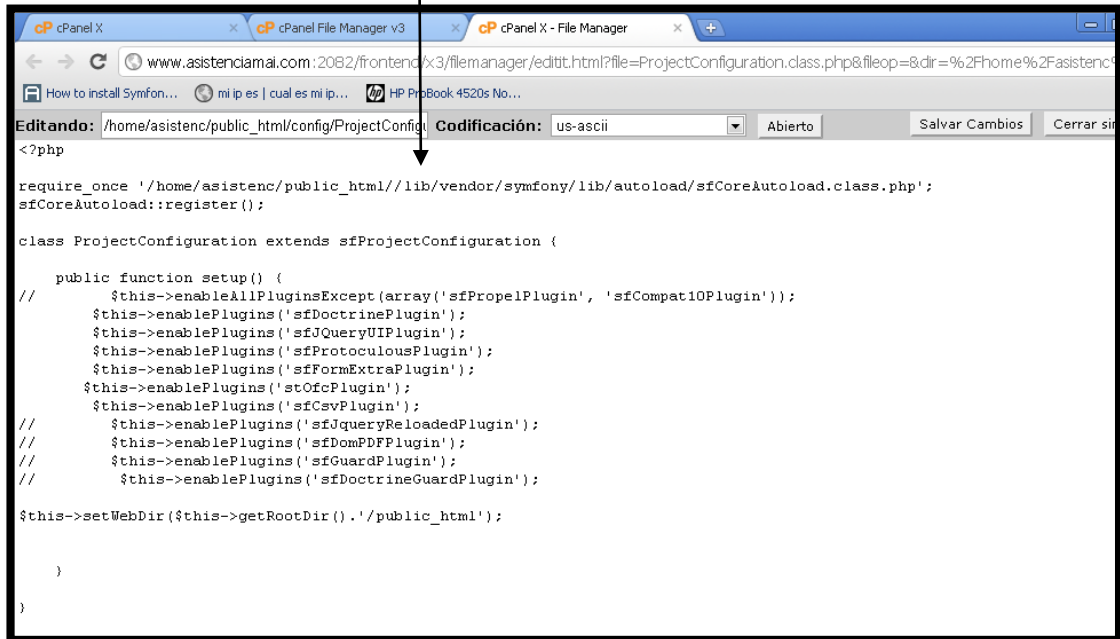
Se procede a realizar el cambio de servidor a uno ubicado en Canadá que ofrece la configuración que es compatible, y sirve para hacer las pruebas.

1. Para subir los ficheros se utilizó la administración del servidor de archivos del hosting.
2. La carpeta Web de la aplicación se subió a la carpeta Public_html de hosting.
3. Las demás carpetas se las subió a la raíz del espacio contratado /home/asistenc/CONTROL_ASISTENCIA
4. La carpeta Symfony se la copió en la carpeta de la aplicación lib/vendor para que se puede ejecutar sin ninguna novedad de direccionamiento.



Configuración de ficheros para re direccionamiento:

1. Se modificó el archivo config/ProjectConfig.class.php, se modificó el enrutamiento.



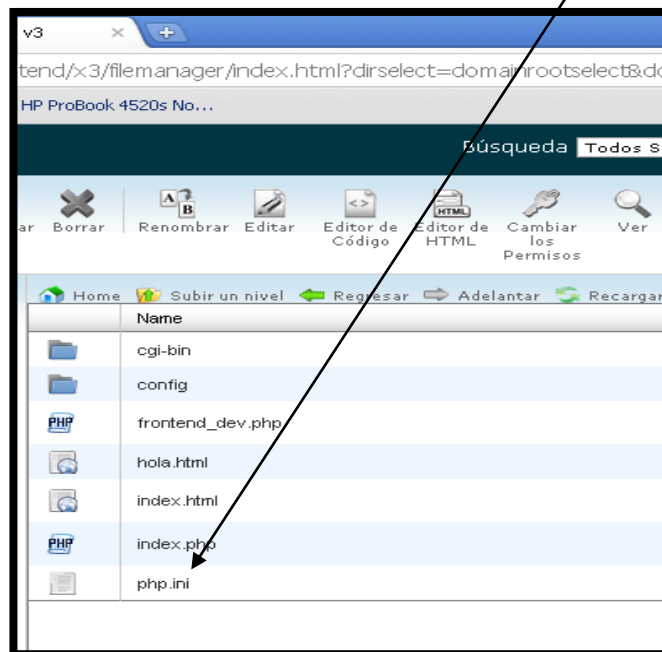
```
<?php
require_once '/home/asistenc/public_html/lib/vendor/symfony/lib/autoload/sfCoreAutoload.class.php';
sfCoreAutoload::register();

class ProjectConfiguration extends sfProjectConfiguration {

    public function setup() {
        // $this->enableAllPluginsExcept(array('sfPropelPlugin', 'sfCompat10Plugin'));
        $this->enablePlugins('sfDoctrinePlugin');
        $this->enablePlugins('sfjQueryUIPlugin');
        $this->enablePlugins('sfProtoculousPlugin');
        $this->enablePlugins('sfFormExtraPlugin');
        $this->enablePlugins('stOfcPlugin');
        $this->enablePlugins('sfCsvPlugin');
        // $this->enablePlugins('sfjQueryReloadedPlugin');
        // $this->enablePlugins('sfDomPDFPlugin');
        // $this->enablePlugins('sfGuardPlugin');
        // $this->enablePlugins('sfDoctrineGuardPlugin');

        $this->setWebDir($this->getRootDir().'public_html');
    }
}
```

2. Se añadió en el public_html el archivo php.ini, para que funcione las páginas php.



El resultado de la instalación se realizó con éxito.

Carta de aceptación del sistema

ACTA DE ENTREGA / RECEPCIÓN		
Proyecto: Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.		
Tema a tratar: Entrega del Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico.		
Fecha: 17 de junio del 2011		
<p>Luego de cumplidas todas las etapas contempladas en el plan de proyecto de implementación del Sistema Web con software libre para el Control de Asistencia de Personal de la Dirección Provincial del Ambiente de Imbabura, mediante un colector de datos biométrico, incluyendo la etapa de capacitación, validación funcional y validación de datos con la Lcda. Mónica Maza Rivadeneira, secretaria de la institución y encargada de la administración del sistema, se acuerda lo siguiente:</p> <ul style="list-style-type: none"> • El Ing. Marco Andrés Morales Vizcaíno, Técnico de Sistemas de la Dirección Provincial Ambiental de Imbabura, luego de haber recibido la capacitación técnica necesaria, será el encargado de realizar el mantenimiento técnico y brindar soporte a los usuarios interesados, así como del desarrollo de nuevos requerimientos que serán previamente aprobados por el Director de la Institución. • El nuevo sistema de control asistencia de personal entrará en funcionamiento desde el día 17 de agosto de 2011. • Se verifica con los presentes el funcionamiento de Instalación realizada en la institución del cable con conectores rs232, que van del Reloj Biométrico a PC, de Ibarra y Cotacachi. • Se observa con los presentes el funcionamiento de la descarga de datos del reloj biométrico por medio del software suministrado, en las PC de la secretaria de Ibarra y Cotacachi. • Resumen de los datos ingresados al nuevo sistema. 		
Departamentos		
Codepartamentos	Descripcion	Observaciones
1	ADMINISTRATIVO FINANCIERO	
4	CALIDAD AMBIENTAL	
2	PATRIMONIO NATURAL	
6	SEGUIMIENTO Y EVALUACION	
5	UNIDAD JURIDICA	
Sección		
Código Sección	Departamentos	Sección
7	ADMINISTRATIVO FINANCIERO	ASEO
12	CALIDAD AMBIENTAL	CALIDAD AMBIENTAL
3	ADMINISTRATIVO FINANCIERO	CONTABILIDAD
10	PATRIMONIO NATURAL	FORESTAL
4	ADMINISTRATIVO FINANCIERO	GUARDALMACEN
14	UNIDAD JURIDICA	JURIDICO
15	PATRIMONIO NATURAL	RESPONSABLE UNIDAD
8	ADMINISTRATIVO FINANCIERO	RESPONSABLE UNIDAD
1	ADMINISTRATIVO FINANCIERO	SECRETARIA

9	ADMINISTRATIVO FINANCIERO	SECRETARIA AUX
---	---------------------------	----------------

Turno

Número	Descripcion	HoraEnt	MinEnt	HoraSalAlm	MinutoSalAlm	HoraEntAlm	MinutoEntAlm	HoraSal	MinutoSal	MinutosGracia
1	NORMAL	8	0	13	0	14	0	17	0	10

Cargo

Numcargo	Descripcion	Observaciones
1	SERVIDOR PUBLICO DE APOYO 1	

Usuarios

Idusuario	Usuario	Contrasena	Editar
1	admin	a123	ADMINISTRADOR
2	101	101	EMPLEADO
7	102	102	EMPLEADO

Empleados

Cod.	Departamento	Seccion	TurnoEmplead	Cédula	Apellidos	Nombres	Teléfono	Estado
101	ADMINISTRATIVO FINANCIERO	SECRETARIA	NORMAL 8:00 A 13:00 14:00 A 17:00		MAZA RIVADENEIRA	MONICA LUCIA		ACTIVO
102	ADMINISTRATIVO FINANCIERO	GUARDALMACEN	NORMAL 8:00 A 13:00 14:00 A 17:00		ENCALADA BUITRON	HUGO BENEDICTO		ACTIVO
103	ADMINISTRATIVO FINANCIERO	RESPONSABLE UNIDAD	NORMAL 8:00 A 13:00 14:00 A 17:00		ESTRADA AVILES	JULIO CESAR		ACTIVO
104	CALIDAD AMBIENTAL	CALIDAD AMBIENTAL	NORMAL 8:00 A 13:00 14:00 A 17:00		ALDAS	JESSIKA		ACTIVO
105	PATRIMONIO NATURAL	VIDA SILVESTRE	NORMAL 8:00 A 13:00 14:00 A 17:00		PANTOJA MONTALVO	SEGUNDO MARCELO		ACTIVO
106	ADMINISTRATIVO FINANCIERO	SECRETARIA AUX	NORMAL 8:00 A 13:00 14:00 A 17:00		RAMIREZ ALBERCA	HISABEL RICARD		ACTIVO
107	ADMINISTRATIVO FINANCIERO	TESORERIA	NORMAL 8:00 A 13:00 14:00 A 17:00		VOSMEDIANO MANOSALVAS	EDUARDO		ACTIVO
108	PATRIMONIO NATURAL	FORESTAL	NORMAL 8:00 A 13:00 14:00 A 17:00		ROSETO MIER	MANUEL		ACTIVO
109	PATRIMONIO NATURAL	VIDA SILVESTRE	NORMAL 8:00 A 13:00 14:00 A 17:00		ROSALES RODRIGUEZ	CARLOS MANUEL		ACTIVO
110	SEGUIMIENTO Y EVALUACION	SEGUIMIENTO Y EVALUACION	NORMAL 8:00 A 13:00 14:00 A 17:00		CHALA OGNAGA	JENY MARIBEL		ACTIVO

• **Resumen de Carga de datos y Cálculos:**

• **Asistencia:**

En la Base de datos se cargará los datos de la asistencia del personal desde el mes de enero del 2011 hasta la fecha:

• **Permisos Movilizaciones y vacaciones:**

En la Base de datos se carga los datos de permisos desde el mes de abril hasta la fecha.

• **Cálculos:**

Se realiza los cálculos de minutos de atraso, minutos extras, horas trabajadas y verificación de permisos, vacaciones y movilizaciones.

• **Reportes:**

Se genera un reporte con los cálculos descritos de forma dinámica (ingresando rango de Fechas y rango de personal), con su respectivo grafico estadístico

Se genera un pdf con el formato requerido por la institución.

Para constancia y ratificando el contenido de esta acta firman:

Nombre	Cargo	Firma
Ing. Segundo Fuentes	Director Provincial del Ambiente de Imbabura	
Ing. Marco Andrés Morales Vizcaíno	Técnico de Sistemas de la Institución	
Lcda. Mónica Maza	Secretaria de la Institución	
Sergio Benítez T.	Equipo de desarrollo	

Observaciones:

Se entrega la siguiente documentación al Ing. Marco Andrés Morales

- Carpeta de Proyecto con documentación Impresa de:
 - Artefactos generados siguiendo la Metodología RUP para el Sistema de Control de Asistencia.
 - Documento de Visión.
 - Lista de Riesgos.
 - Caso de Desarrollo.
 - Plan de Administración de Requerimientos.
 - Documento de Arquitectura del Software.
 - Plan de Desarrollo de Software.
 - Glosario.
 - Casos de Negocio.
 - Casos de Uso.
 - Actas de Trabajo.

- Cd con información digital de:
 - Código fuente de formas y librerías.
 - Documentación siguiendo Metodología RUP.
 - Diagramas Entidad Relación de Base de Datos.
 - Scripts de creación de Usuarios y Objetos de Base de Datos.
 - Diccionario de Datos.
 - Manual de Usuario.
 - Manual de Operación.
 - Documentación para publicación de Aplicación en el Servidor.

ANEXO NUM 10: BENCHMARK (Comparación Sistemas similares) y Cálculo de costo (Cocomo)

TABLA DE COMPARACION

CARACTERÍSTICAS	SISTEMA WEB MAI	FULLTIME	ATTENDANCE MANAGEMENT (ESPAÑOL)
Ingreso Seguro	X	X	X
Ingreso Parámetros	X	X	X
Ingreso P/M/V	X	X	X
Reporte Ent-Salida	X	X	X
Reporte P/MV	X	X	X
Exportación a pdf	X	X	-
Grafico Estadístico	X	X	-
Consultas Dinámicas	X	-	-
Cliente-Servidor	X	-	-
Consultas Vía Web	X	-	-
Reportes Vía Web	X	-	-
Carga de datos Vía Web	X	-	-
Integración de datos de varios dispositivos	X	X	X

Fuente: Propia

ESTIMACIONES Y CÁLCULO DE COSTOS DEL SISTEMA WEB

Utilizaremos el modelo COCOMO (CONstructive COst MOdel) desarrollado por Barry M. Boehm, se engloba en el grupo de los modelos algorítmicos que tratan de establecer una relación matemática la cual permite estimar el esfuerzo y tiempo requerido para desarrollar un producto y con estos parámetros su costo.

Necesitamos estimar líneas de código del Sistema, aproximadamente 7000 líneas de código se consideró, pero con la ayuda de el framework Symfony que genera varias tareas, la codificación real será de unos 4500 aproximadamente, esto es parámetro:

$$kldc = 4.5$$

COCOMO define tres modos de desarrollo o tipos de proyectos, "Orgánico" proyectos relativamente sencillos, menores de 50 KDLC líneas de código, es éste el que se utilizará para los cálculos.⁷

PROYECTO SOFTWARE	a	b	c	d
Orgánico	3,2	1,05	2,5	0,38

Fuente: <http://es.wikipedia.org/wiki/COCOMO>

⁷<http://es.wikipedia.org/wiki/COCOMO>: artículo COCOMO.

CONDUCTORES DE COSTE	VALORACIÓN					
	<i>Muy bajo</i>	<i>Bajo</i>	<i>Nominal</i>	<i>Alto</i>	<i>Muy alto</i>	<i>Extr. alto</i>
Fiabilidad requerida del software	0,75	0,88	1.00	1,15	1,40	-
Tamaño de la base de datos	-	0,94	1.00	1,08	1,16	-
Complejidad del producto	0,70	0,85	1.00	1,15	1,30	1,65
Restricciones del tiempo de ejecución	-	-	1.00	1,11	1,30	1,66
Restricciones del almacenamiento principal	-	-	1.00	1,06	1,21	1,56
Volatilidad de la máquina virtual	-	0,87	1.00	1,15	1,30	-
Tiempo de respuesta del ordenador	-	0,87	1.00	1,07	1,15	-
Capacidad del analista	1,46	1,19	1.00	0,86	0,71	-
Experiencia en la aplicación	1,29	1,13	1.00	0,91	0,82	-
Capacidad de los programadores	1,42	1,17	1.00	0,86	0,70	-
Experiencia en S.O. utilizado	1,21	1,10	1.00	0,90	-	-
Experiencia en el lenguaje de programación	1,14	1,07	1.00	0,95	-	-
Prácticas de programación modernas	1,24	1,10	1.00	0,91	0,82	-
Utilización de herramientas software	1,24	1,10	1.00	0,91	0,83	-
Limitaciones de planificación del proyecto	1,23	1,08	1.00	1,04	1,10	-

Fuente: <http://es.wikipedia.org/wiki/COCOMO>

Otro parámetro que necesitamos es encontrar el factor $m(x)$, el cual se obtiene de la evaluación de 15 conductores de coste, lo que se explica en la siguiente tabla:

$$M(x) = 1.15 * 0.94 * 1.00 * 1.11 * 1.00 * 1.00 * 1.07 * 1.00 * 0.82 * 0.86 * 0.95 * 0.91 * 0.91 * 1.08$$

$$m(x) = 0,769263097$$

Ahora reemplazamos en la fórmula para medir el Esfuerzo:

$$E = a(Kldc)^b m(x)^8$$

Entonces:

$$E = 3.2 * (4.5)^{1.05} * (0,769263097)$$

⁸Fuente: <http://es.wikipedia.org/wiki/COCOMO>

$$E = (3.2 * 4,851467) * (0,769263097)$$

$$\mathbf{E = 11,94 HOMBRES /MES}$$

En las formulas del modelo intermedio podemos calcular el tiempo:

$$\text{Tiempo} = T = c * E^d$$

$$T = 2.5 * (11.94)^{0.38}$$

$$\mathbf{T = 6,42 MESES}$$

$$\text{Personas} = P = E/T$$

$$P = 11.94/6.42$$

$$\mathbf{P = 1.86 PERSONAS}$$

$$Ldc/E = 4500/11.94$$

$$\mathbf{Ldc/E = 377 LINEAS HOMBRE/MES}$$

Entonces de los resultados se puede estimar un costo de mano de obra:

Entonces:

$$\text{Costo} = \text{Personas} * \text{Salario} * \text{Tiempo}$$

$$\text{Costo} = (2 * 263.00) * 6.42$$

$$\mathbf{COSTO = \$ 3.376.92 APROXIMADAMENTE}$$