

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS COMPUTACIONALES**

MANUAL TÉCNICO

TEMA:

**SISTEMATIZACIÓN DEL CENTRO DE ENTRENAMIENTO PARA EL
ALTO RENDIMIENTO DE CARPUELA USANDO HERRAMIENTAS DE
SOFTWARE LIBRE CON LOS MÓDULOS (ADMINISTRATIVO Y
CONTROL DE HOSPEDAJE).**

AUTOR:

ALVARO DAVID ARIAS CORONADO

DIRECTOR:

Ing. PEDRO GRANDA

Ibarra – Ecuador

2015

INDICE DE CONTENIDO

1.- Introducción.....	3
2.- Materiales y Métodos.....	4
2.1.- Procesos:.....	4
2.2.- Procedimientos.....	4
3.- Procesos a ser implementados.....	5
3.1 Seguimiento del deportista.....	5
3.2.- Hospedaje del deportista.....	5
4.- Software.....	6
4.1.- Base de Datos Postgres.....	6
4.2.- Herramientas de Sistema Web.....	6
4.3 Servidor.....	7
5.- Desarrollo.....	7

INDICE DE ILUSTRACIONES

Ilustración 1: Arquitectura.....	7
Ilustración 2: Seguimiento del Deportista.....	8
Ilustración 3: Hospedaje del Deportista.....	8
Ilustración 4: Base de Datos CEAR.....	9

INDICE DE TABLAS

Tabla 1: Deportista.....	11
Tabla 2:Habitaciones.....	12
Tabla 3:Ingresos Visitantes.....	13
Tabla 4: Inventario.....	14
Tabla 5:Logros Obtenidos.....	15
Tabla 6: Medicación.....	15
Tabla 7: Medicación General.....	16
Tabla 8: Nombre del Deporte.....	17
Tabla 9: Personal.....	17
Tabla 10: Psicología Deportiva.....	19
Tabla 11: Tipo deportista.....	19
Tabla 12: Traumatología.....	20
Tabla 13:Usuario.....	21
Tabla 14: Fisiatria.....	22
Tabla 15: Enfermeria.....	22
Tabla 16: Área de Trabajo.....	24
Tabla 17: Alimentación.....	24
Tabla 18: Alimentos Diarios.....	25
Tabla 19: Tipo de Usuario.....	25
Tabla 20: Manager DAO.....	26

MANUAL TECNICO

1.- Introducción.

En la actualidad el Centro de Entrenamiento Para el Alto Rendimiento ubicado en Carpuela tiene muchos retos y que se han convertido en limitantes para el buen desarrollo de actividades dentro de la institución en esto hablamos de lo que es la parte administrativa y la atención al deportista dentro del mismo podemos citar como ejemplo los siguientes procesos: seguimiento del deportistas y de visitantes para la institución y también como lo es el control de artículos y demás activos que se encuentran dentro de la institución y por lo que se necesita urgentemente la optimización de estos procesos para poder optimizarlos y agilizarlos.

Actualmente el Centro de Entrenamiento no cuenta con ningún sistema que permita guiarnos para el desarrollo por lo que hemos escogido como herramientas la utilización de herramientas de software libre para la realización de toda la sistematización de los procesos de nuestro nuevo sistema. Algo que debemos tomar muy en cuenta es que actualmente todos los procesos se los lleva manualmente por lo que se hace un poco dañino para el medio ambiente y por lo que los procesos son un poco más lentos y mucho mas deficientes en lo que es la organización de la empresa.

Para la toma de decisiones por la parte administrativa es un punto muy importante para mejorar lo que es mantener al deportista en un ambiente relajado y para un buen desempeño en las prácticas y desarrollo del deporte. Este proyecto tiene como finalidad mejorar la gestión y acelerar los procesos que existen dentro de la institución y los que nos hemos mantenido fijados como punto de alcance los siguientes procesos: Seguimiento del Deportista y Hospedaje del Deportista con lo cual se procederá a la elaboración de los mismos.

Con todo lo anterior dicho los beneficiarios serán los deportistas del país y a su vez el personal administrativo del Centro de Entrenamiento para el Alto Rendimiento de Carpuela ya que será el personal que se encuentre involucrado dentro de lo que es la sistematización de información.

También se debe tomar muy en cuenta que al momento que nosotros sistematizamos los procesos dichos se tendrá información inmediata capaz de acceder a la información de cualquier deportista en cuestión de segundos.

Las herramientas que utilizaremos para el desarrollo del sistema serán de software libre en nuestro caso usaremos herramientas como:

- ECLIPSE
- JSF
- POSTGRES
- PRIMEFACES
- CSS
- JAVASCRIPT
- ITEXTPDF

Los cuales nos ayudaran a lo que es el desarrollo del sistema y los cuales nos ayudaran ya que son herramientas robustas capaces de soportar una gran cantidad de información sin colapsar.

2.- Materiales y Métodos

El método con el cual fue desarrollado este proyecto de desarrollo de un sistema web fue utilizar procesos y procedimientos los cuales vamos a detallar a continuación:

2.1.- Procesos: Proceso es el conjunto de actividades o tareas, mutuamente relacionadas entre sí que admite elementos de entrada durante su desarrollo ya sea al inicio o a lo largo del mismo, los cuales se administran, regulan o autorregulan bajo modelos de gestión particulares para obtener elementos de salida o resultados esperados.

2.2.- Procedimientos Los procedimientos se los puede definir como módulos homogéneos que son capaces de especificar y detallar un proceso, los cuales forman un conjunto ordenado de actividades u operaciones determinadas secuencialmente y tienen relación directa con los responsables de la ejecución, como parte fundamental de los procedimientos

es que deben cumplir con políticas, normas establecidas, la duración del procedimiento y el flujo de documentos a seguir para su correcto desarrollo.

Una característica muy importante de los procedimientos es que deben ser documentados en todo el desarrollo de estos paso a paso para tener un control de ejecución. El objetivo primordial de los procedimientos es el identificar y señalar para qué?, por qué?, quién?, donde?, cómo?, y cuando de cada una de las actividades que integran los procedimientos de los diferentes procesos y procedimientos del CENTRO DE ENTRENAMIENTO PARA EL ALTO RENDIMIENTO CARPUELA en cuanto a todo el proceso administrativo.

3.- Procesos a ser implementados

3.1 Seguimiento del deportista.

Dentro de este llevaremos el seguimiento del deportista dentro de la institución y tomaremos en cuenta la información que sea ingresada por las diferentes áreas de los departamentos tales como:

- Área Médica.
- Área de Seguridad.
- Área de Servicios.
- Área Administrativa.
- Área de Cocina.

Por medio de todos estos departamentos se podrá obtener información veraz e inmediata por la cual se podrá tomar decisiones por la parte administrativa para el mejoramiento o adecuaciones que necesiten cada uno de cada departamento dentro de la institución.

3.2.- Hospedaje del deportista.

Dentro de lo que es el hospedaje del deportista se procederá a la utilización de este proceso únicamente por el área de servicios la cual es la encargada de conocer y asignar una habitación a un nuevo deportista que ingrese a la institución a ocupar los espacios que ofrece el centro de entrenamiento.

4.- Software

El software que se ha utilizado son los siguientes:

4.1.- Base de Datos Postgres

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

- ✓ La versión de la base de datos que usaremos es la 9.3 por mantener una gran potencia y robustecida para la gran cantidad de información que mantendremos de los deportistas y de los visitantes

4.2.- Herramientas de Sistema Web

✓ ECLIPSE

Es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java.

Es la herramienta escogida para el desarrollo del sistema web en el cual usaremos JSF, JPA para la elaboración del sistema.

✓ JSF

Con este tipo de tecnologías y frameworks que están basadas en Web que ayudan a facilitar el desarrollo de interfaces de usuario en lo que corresponde en aplicaciones Java EE¹

✓ PRIMEFACES

Usaremos la versión de primefaces 5.0 la cual nos facilitara las vistas enriquecidas de nuestro sistema.

¹ Fuente: http://es.wikipedia.org/wiki/Java_EE

La funcionalidad que tiene las herramientas mencionadas anteriormente nos proporcionaran el siguiente diagrama.

Arquitectura

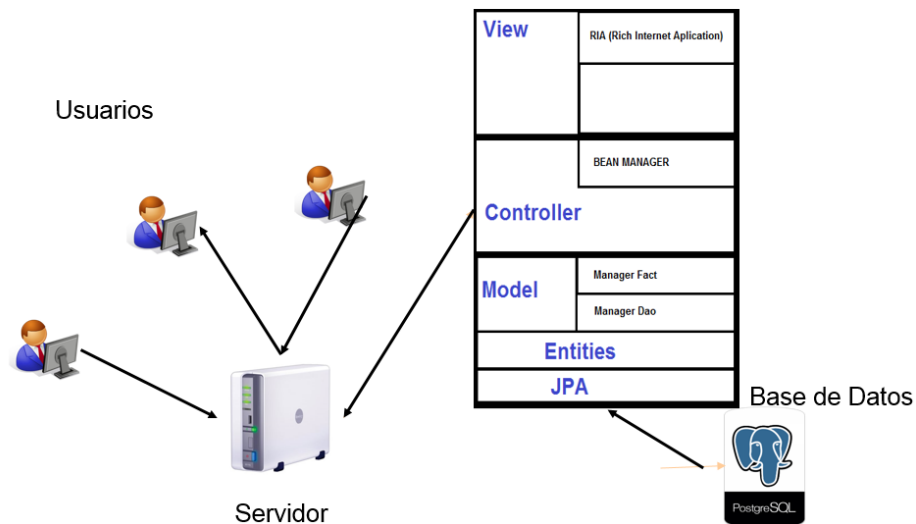


Ilustración 1: Arquitectura

Fuente: Propia

Para los reportes usaremos itextpdf el cual nos proporcionara pdf para obtener reportes necesarios para mantener los reportes necesarios para la parte administrativa.

4.3 Servidor

Apache Tomcat es el encargado de mantener nuestro sistema en línea y el cual usaremos la versión 7.0.52.

5.- Desarrollo

Procesos: Seguimiento del deportista.

El proceso que se mantiene y el cual ha sido sistematizado es el siguiente:

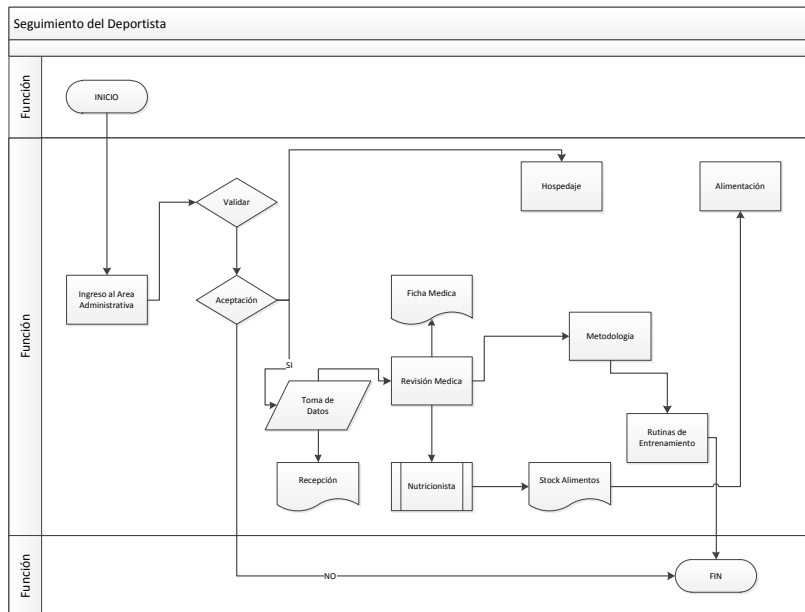


Ilustración 2: Seguimiento del Deportista

Fuente: Propia

Proceso: Hospedaje del Deportista

El siguiente proceso se ha levantado para el hospedaje deportista el cual va a ser sistematizado en el nuevo CEAR.

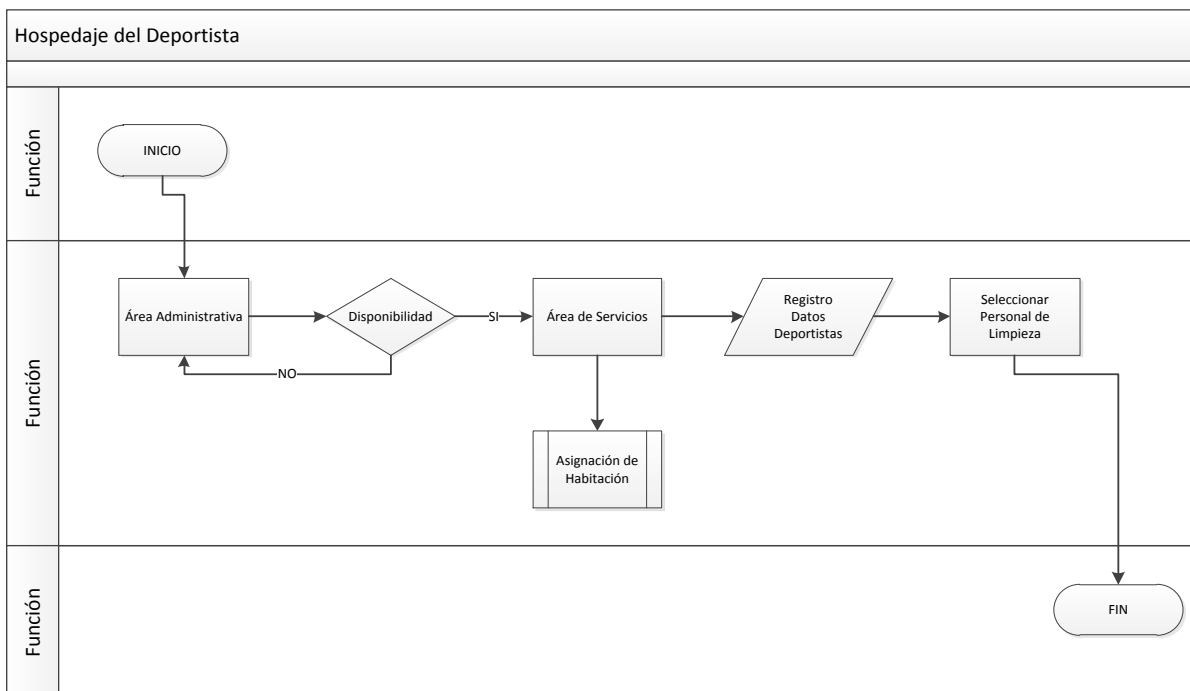


Ilustración 3: Hospedaje del Deportista

Fuente: Propia

El diseño de la base Datos es la siguiente:

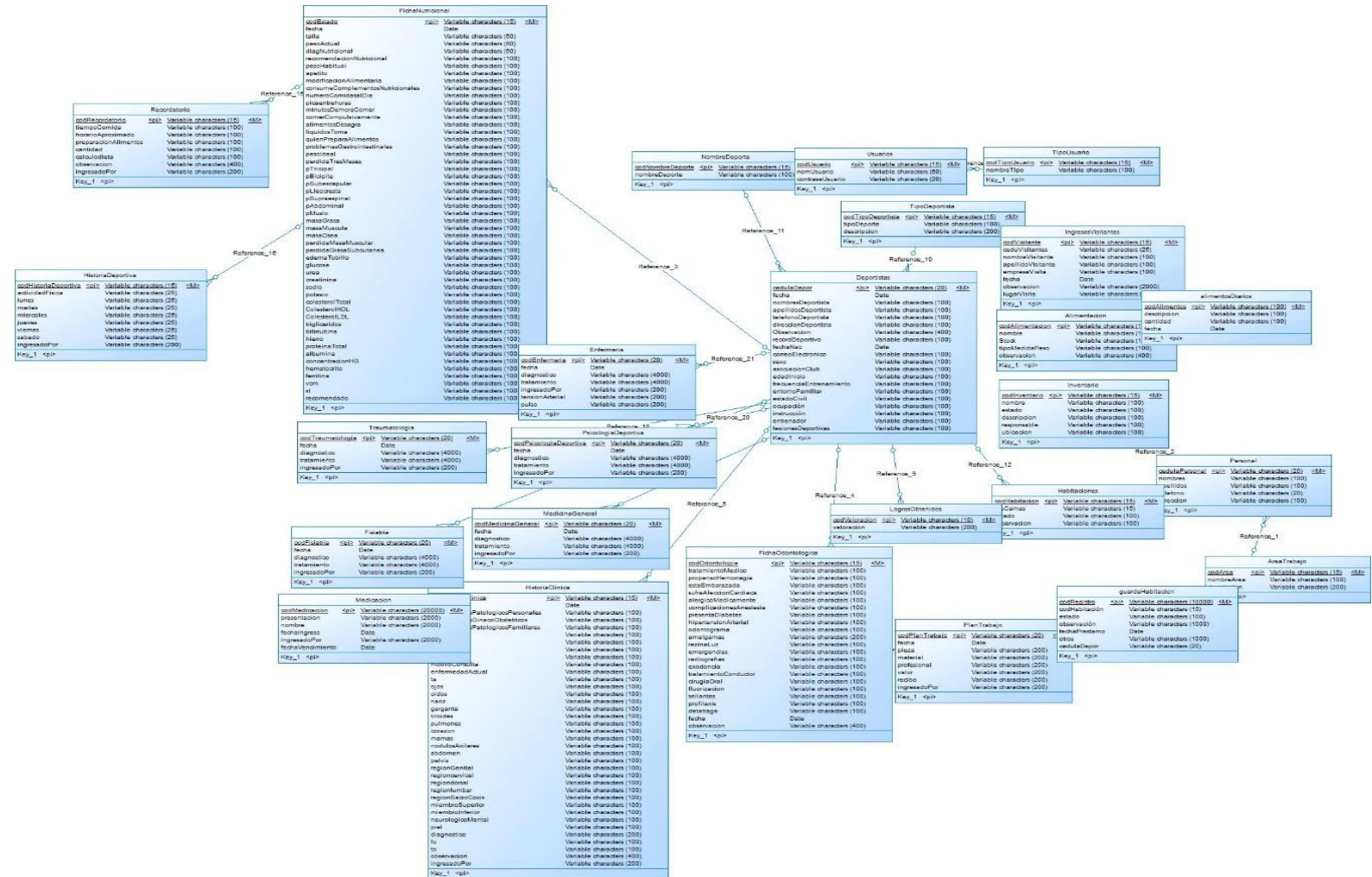


Ilustración 4: Base de Datos CEAR

Fuente: Propia

El diseño de la base de datos la cual detallaremos las tablas que hemos utilizado en la base de datos en total tenemos 26:

Alimentación
Alimentosdiarios
Areatrabajo
Deportistas
Enfermería
Fichanutricional
Fichaodontologica
Fisiatría
Guardahabitacion
Habitaciones
Historiaclinica
Historiadeportiva
Ingresosvisitantes
Inventario
Logrosobtenidos
Medicación
Medicinageneral
Nombredeporte
Personal
Plantrabajo
Psicologiadeportiva
Tipodeportista
Tipousuario
Traumatología
Usuarios

Las tablas usadas para el hospedaje del deportista son las siguientes: deportistas, personal, usuarios, rol.

En la tabla deportista tenemos la siguiente información la cual es la mas importante para poder asignarle una habitación:

Fuente: Propia

Tabla 1: Deportista

Deportista	
Enfermeria	addEnfermeria (Enfermeria enfermeria)
Fichanutricional l	addFichanutricional (Fichanutricional fichanutricional)
Fichaodontologica ca	addFichaodontologica (Fichaodontologica fichaodontolog ica)
Fisiatria	addFisiatria (Fisiatria fisiatria)
Habitacione	addHabitacione (Habitacione habitacione)
Historiaclinica	addHistoriaclinica (Historiaclinica historiaclinica)
Logrosobtenido	addLogrosobtenido (Logrosobtenido logrosobtenido)
Medicinageneral	addMedicinageneral (Medicinageneral medicinageneral)
Psicologiadepor tiva	addPsicologiadeportiva (Psicologiadeportiva psicologia deportiva)
Traumatologia	addTraumatologia (Traumatologia traumatologia)
java.lang.String	getApellidosdeportista ()
java.lang.String	getAsociacionclub ()
java.lang.String	getCeduladepor ()
java.lang.String	getCorreoelectronico ()
java.lang.String	getDirecciondeportista ()
java.lang.String	getEdadinicio ()

Dentro de lo que es el asignamiento de habitación tendremos las habitaciones que son las siguientes:

Fuente: Propia

Tabla 2:Habitaciones

Habitaciones	
java.lang.String	<u>getCodhabitacion</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()
java.lang.String	<u>getEstado</u> ()
java.lang.String	<u>getNrocamas</u> ()
java.lang.String	<u>getObservacion</u> ()
<u>Personal</u>	<u>getPersonal</u> ()
void	<u>setCodhabitacion</u> (java.lang.String codhabitacion)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setEstado</u> (java.lang.String estado)
void	<u>setNrocamas</u> (java.lang.String nrocamas)
void	<u>setObservacion</u> (java.lang.String observacion)
void	<u>setPersonal</u> (<u>Personal</u> personal)

Por medio de las mismas se procederá a realizar el asignamiento de un deportista a una habitación.

Para el proceso Seguimiento del Deportista usaremos las siguientes tablas: seguridad, área servicios, medica, administrativa. La tabla más principal es la del deportista que ya la hemos visualizado anteriormente pero aquí obtendremos toda la información necesaria de todos

los departamentos para realizar reportes y demás información que se necesita para la toma de decisiones.

A continuación detallaremos las tablas que involucran al segundo proceso el cual es el seguimiento del deportista dentro de la institución.

Tabla Ingresos de Visitantes del área de seguridad.

Fuente: Propia

Tabla 3:Ingresos Visitantes

Ingresos Visitantes	
java.lang.String	<u>getApellidovisitante</u> ()
java.lang.String	<u>getCeduvisitantes</u> ()
java.lang.String	<u>getCodvisitante</u> ()
java.lang.String	<u>getEmpresavisita</u> ()
java.util.Date	<u>getFecha</u> ()
java.lang.String	<u>getLugarvisita</u> ()
java.lang.String	<u>getNombrevisitante</u> ()
java.lang.String	<u>getObservacion</u> ()
void	<u>setApellidovisitante</u> (java.lang.String apellidovisitante)
void	<u>setCeduvisitantes</u> (java.lang.String ceduvisitantes)
void	<u>setCodvisitante</u> (java.lang.String codvisitante)
void	<u>setEmpresavisita</u> (java.lang.String empresavisita)
void	<u>setFecha</u> (java.util.Date fecha)

void	<u>setLugarvisita</u> (java.lang.String lugarvisita)
void	<u>setNombrevisitante</u> (java.lang.String nombrevisitante)
void	<u>setObservacion</u> (java.lang.String observacion)

Dentro de este se procede a obtener los registros de ingreso de visitantes al Centro de entrenamiento para sus respectivas estadísticas y reportes para la toma de decisiones.

La tabla Inventario:

Fuente: Propia

Tabla 4: Inventario

Inventario	
java.lang.String	<u>getCodinventario</u> ()
java.lang.String	<u>getDescripcion</u> ()
java.lang.String	<u>getEstado</u> ()
java.lang.String	<u>getNombre</u> ()
<u>Personal</u>	<u>getPersonal</u> ()
java.lang.String	<u>getResponsable</u> ()
java.lang.String	<u>getUbicacion</u> ()
void	<u>setCodinventario</u> (java.lang.String codinventario)
void	<u>setDescripcion</u> (java.lang.String descripcion)
void	<u>setEstado</u> (java.lang.String estado)
void	<u>setNombre</u> (java.lang.String nombre)
void	<u>setPersonal</u> (<u>Personal</u> personal)

void	<u>setResponsable</u> (java.lang.String responsable)
void	<u>setUbicacion</u> (java.lang.String ubicacion)

A continuación detallaremos la tabla de Logros Obtenidos:

Fuente: Propia

Tabla 5: Logros Obtenidos

Logros Obtenidos	
java.lang.String	<u>getCodvaloracion</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()
java.lang.String	<u>getValoracion</u> ()
void	<u>setCodvaloracion</u> (java.lang.String codvaloracion)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setValoracion</u> (java.lang.String valoracion)

A continuación detallaremos la tabla de Medicación:

Fuente: Propia

Tabla 6: Medicación

Medicación	
java.lang.String	<u>getCodmedicacion</u> ()
java.util.Date	<u>getFechaingreso</u> ()
java.util.Date	<u>getFechavencimiento</u> ()
java.lang.String	<u>getIngresadopor</u> ()
java.lang.String	<u>getNombre</u> ()

java.lang.String	<u>getPresentacion</u> ()
void	<u>setCodmedicacion</u> (java.lang.String codmedicacion)
void	<u>setFechaingreso</u> (java.util.Date fechaingreso)
void	<u>setFechavencimiento</u> (java.util.Date fechavencimiento)
void	<u>setIngresadopor</u> (java.lang.String ingresadopor)
void	<u>setNombre</u> (java.lang.String nombre)
void	<u>setPresentacion</u> (java.lang.String presentacion)

A continuación detallaremos la tabla Medicina General:

Fuente: Propia

Tabla 7: Medicación General

Medicación General	
java.lang.String	<u>getCodmedicinageneral</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()
java.lang.String	<u>getDiagnostico</u> ()
java.util.Date	<u>getFecha</u> ()
java.lang.String	<u>getIngresadopor</u> ()
java.lang.String	<u>getTratamiento</u> ()
void	<u>setCodmedicinageneral</u> (java.lang.String codmedicinageneral)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setDiagnostico</u> (java.lang.String diagnostico)

void	setFecha (java.util.Date fecha)
void	setIngresadopor (java.lang.String ingresadopor)
void	setTratamiento (java.lang.String tratamiento)

A continuación la tabla Nombre del Deporte:

Fuente: Propia

Tabla 8: Nombre del Deporte

Nombre del Deporte	
Deportista	addDeportista (Deportista deportista)
java.lang.String	getCodnombredeporte ()
java.util.List< Deportista >	getDeportistas ()
java.lang.String	getNombredeporte ()
Deportista	removeDeportista (Deportista deportista)
void	setCodnombredeporte (java.lang.String codnombredeporte)
void	setDeportistas (java.util.List< Deportista > deportistas)
void	setNombredeporte (java.lang.String nombredeporte)

A continuación se describe la tabla Personal:

Fuente: Propia

Tabla 9: Personal

Personal	
Habitacione	addHabitacione (Habitacione habitacione)
Inventario	addInventario (Inventario inventario)

java.lang.String	<u>getApellidos</u> ()
<u>Areatrabajo</u>	<u>getAreatrabajo</u> ()
java.lang.String	<u>getCedulapersonal</u> ()
java.lang.String	<u>getDireccion</u> ()
java.util.List< <u>Habitacione</u> >	<u>getHabitaciones</u> ()
java.util.List< <u>Inventario</u> >	<u>getInventarios</u> ()
java.lang.String	<u>getNombres</u> ()
java.lang.String	<u>getTelefono</u> ()
<u>Habitacione</u>	<u>removeHabitacione</u> (<u>Habitacione</u> habitacione)
<u>Inventario</u>	<u>removeInventario</u> (<u>Inventario</u> inventario)
void	<u>setApellidos</u> (java.lang.String apellidos)
void	<u>setAreatrabajo</u> (<u>Areatrabajo</u> areatrabajo)
void	<u>setCedulapersonal</u> (java.lang.String cedulapersonal)
void	<u>setDireccion</u> (java.lang.String direccion)
void	<u>setHabitaciones</u> (java.util.List< <u>Habitacione</u> > habitaciones)
void	<u>setInventarios</u> (java.util.List< <u>Inventario</u> > inventarios)
void	<u>setNombres</u> (java.lang.String nombres)
void	<u>setTelefono</u> (java.lang.String telefono)

A continuación describimos Psicología Deportiva:

Fuente: Propia
Tabla 10: Psicología Deportiva

Psicología Deportiva	
java.lang.String	<u>getCodpsicologiadeportiva</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()
java.lang.String	<u>getDiagnostico</u> ()
java.util.Date	<u>getFecha</u> ()
java.lang.String	<u>getIngresadopor</u> ()
java.lang.String	<u>getTratamiento</u> ()
void	<u>setCodpsicologiadeportiva</u> (java.lang.String codpsicologiadeportiva)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setDiagnostico</u> (java.lang.String diagnostico)
void	<u>setFecha</u> (java.util.Date fecha)
void	<u>setIngresadopor</u> (java.lang.String ingresadopor)
void	<u>setTratamiento</u> (java.lang.String tratamiento)

A continuación la tabla Tipo Deportista:

Fuente: Propia
Tabla 11: Tipo deportista

Tipo Deportista	
<u>Deportista</u>	<u>addDeportista</u> (<u>Deportista</u> deportista)

java.lang.String	<u>getCodtipodeportista</u> ()
java.util.List< <u>Deportista</u> >	<u>getDeportistas</u> ()
java.lang.String	<u>getDescripcion</u> ()
java.lang.String	<u>getTipodeporte</u> ()
<u>Deportista</u>	<u>removeDeportista</u> (<u>Deportista</u> deportista)
void	<u>setCodtipodeportista</u> (java.lang.String codtipodeportista)
void	<u>setDeportistas</u> (java.util.List< <u>Deportista</u> > deportistas)
void	<u>setDescripcion</u> (java.lang.String descripcion)
void	<u>setTipodeporte</u> (java.lang.String tipodeporte)

A continuación la tabla Traumatología:

Fuente: Propia

Tabla 12: Traumatología

Traumatología	
java.lang.String	<u>getCodtraumatologia</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()
java.lang.String	<u>getDiagnostico</u> ()
java.util.Date	<u>getFecha</u> ()
java.lang.String	<u>getIngresadopor</u> ()
java.lang.String	<u>getTratamiento</u> ()

void	<u>setCodtraumatologia</u> (java.lang.String codtraumatologia)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setDiagnostico</u> (java.lang.String diagnostico)
void	<u>setFecha</u> (java.util.Date fecha)
void	<u>setIngresadopor</u> (java.lang.String ingresadopor)
void	<u>setTratamiento</u> (java.lang.String tratamiento)

A continuación la tabla Usuario:

Fuente: Propia

Tabla 13:Usuario

Usuario	
java.lang.String	<u>getCodusuario</u> ()
java.lang.String	<u>getContraseusuario</u> ()
java.lang.String	<u>getNomusuario</u> ()
<u>Tipousuario</u>	<u>getTipousuario</u> ()
void	<u>setCodusuario</u> (java.lang.String codusuario)
void	<u>setContraseusuario</u> (java.lang.String contraseusuario)
void	<u>setNomusuario</u> (java.lang.String nomusuario)
void	<u>setTipousuario</u> (<u>Tipousuario</u> tipousuario)

A continuación la tabla Fisiatria:

Fuente: Propia

Tabla 14: Fisiatria

Fisiatría	
java.lang.String	<u>getCodfisiatria</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()
java.lang.String	<u>getDiagnostico</u> ()
java.util.Date	<u>getFecha</u> ()
java.lang.String	<u>getIngresadopor</u> ()
java.lang.String	<u>getTratamiento</u> ()
void	<u>setCodfisiatria</u> (java.lang.String codfisiatria)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setDiagnostico</u> (java.lang.String diagnostico)
void	<u>setFecha</u> (java.util.Date fecha)
void	<u>setIngresadopor</u> (java.lang.String ingresadopor)
void	<u>setTratamiento</u> (java.lang.String tratamiento)

A continuación la tabla Enfermería:

Fuente: Propia

Tabla 15: Enfermería

Enfermería	
java.lang.String	<u>getCodenfermeria</u> ()
<u>Deportista</u>	<u>getDeportista</u> ()

java.lang.String	<u>getDiagnostico</u> ()
java.util.Date	<u>getFecha</u> ()
java.lang.String	<u>getIngresadopor</u> ()
java.lang.String	<u>getPulso</u> ()
java.lang.String	<u>getTemperatura</u> ()
java.lang.String	<u>getTensionarterial</u> ()
java.lang.String	<u>getTratamiento</u> ()
void	<u>setCodenfermeria</u> (java.lang.String codenfermeria)
void	<u>setDeportista</u> (<u>Deportista</u> deportista)
void	<u>setDiagnostico</u> (java.lang.String diagnostico)
void	<u>setFecha</u> (java.util.Date fecha)
void	<u>setIngresadopor</u> (java.lang.String ingresadopor)
void	<u>setPulso</u> (java.lang.String pulso)
void	<u>setTemperatura</u> (java.lang.String temperatura)
void	<u>setTensionarterial</u> (java.lang.String tensionarterial)
void	<u>setTratamiento</u> (java.lang.String tratamiento)

A continuación la tabla Área de Trabajo:

Fuente: Propia

Tabla 16: Área de Trabajo

Área de Trabajo	
<u>Personal</u>	<u>addPersonal</u> (<u>Personal</u> personal)
java.lang.String	<u>getCodarea</u> ()
java.lang.String	<u>getDescripcion</u> ()
java.lang.String	<u>getNombrearea</u> ()
java.util.List< <u>Persona</u> <u>l</u> >	<u>getPersonals</u> ()
<u>Personal</u>	<u>removePersonal</u> (<u>Personal</u> personal)
void	<u>setCodarea</u> (java.lang.String codarea)
void	<u>setDescripcion</u> (java.lang.String descripcion)
void	<u>setNombrearea</u> (java.lang.String nombrearea)
void	<u>setPersonals</u> (java.util.List< <u>Personal</u> > personals)

A continuación la tabla Alimentación:

Fuente: Propia

Tabla 17: Alimentación

Alimentación	
java.lang.String	<u>getCodalimentacion</u> ()
java.lang.String	<u>getNombre</u> ()
java.lang.String	<u>getObservacion</u> ()
java.lang.String	<u>getStock</u> ()

java.lang.String	<u>getTipomedidapeso</u> ()
Void	<u>setCodalimentacion</u> (java.lang.String codalimentacion)
Void	<u>setNombre</u> (java.lang.String nombre)
Void	<u>setObservacion</u> (java.lang.String observacion)
Void	<u>setStock</u> (java.lang.String stock)
Void	<u>setTipomedidapeso</u> (java.lang.String tipomedidapeso)

A continuación la tabla Alimentos Diarios:

Fuente: Propia

Tabla 18: Alimentos Diarios

Alimentos Diarios	
java.lang.String	<u>getCantidad</u> ()
java.lang.String	<u>getCodalimentos</u> ()
java.lang.String	<u>getDescripcion</u> ()
java.util.Date	<u>getFecha</u> ()
void	<u>setCantidad</u> (java.lang.String cantidad)
void	<u>setCodalimentos</u> (java.lang.String codalimentos)
void	<u>setDescripcion</u> (java.lang.String descripcion)
void	<u>setFecha</u> (java.util.Date fecha)

A continuación la tabla Tipo de Usuario:

Fuente: Propia

Tabla 19: Tipo de Usuario

Tipo de Usuario

Usuario	addUsuario (Usuario usuario)
java.lang.String	getCodtipousuario ()
java.lang.String	getNombretipo ()
java.util.List< Usuario >	getUsuarios ()
Usuario	removeUsuario (Usuario usuario)
void	setCodtipousuario (java.lang.String codtipousuario)
void	setNombretipo (java.lang.String nombretipo)
void	setUsuarios (java.util.List< Usuario > usuarios)

Manager DAO

Es la parte donde se manejan los datos para la creación de los crud de nuestro sistema y a continuación se presenta sus métodos que van a ser utilizados por parte del sistema.

Fuente: Propia

Tabla 20: Manager DAO

Manager DAO	
void	actualizar (java.lang.Object pObjeto) Actualiza la informacion de un objeto en la persistencia.
java.lang.String	count (java.lang.Class clase) CUENTA LA CANTIDAD DE REGISTROS INGRESADOS EN NUESTRAS TABLAS.
void	eliminar (java.lang.Class clase, java.lang.Object pID) Elimina un objeto de la persistencia.

java.util.List	<u>findAll</u> (java.lang.Class clase) finder Generico que devuelve todos las entidades de una tabla.
java.util.List	<u>findAll</u> (java.lang.Class clase, java.lang.String orderBy) finder Generico que devuelve todos las entidades de una tabla.
java.util.List	<u>findAllT</u> (java.lang.Class clase, java.lang.String orderBy) finder Generico que devuelve todos las entidades de una tabla.
java.lang.Object	<u>findById</u> (java.lang.Class clase, java.lang.Object pID) Finder generico para buscar un objeto especifico.
java.util.List	<u>findJPQLT</u> (java.lang.String pClausula JPQL) Finder generico que permite ejecutar cualquier sentencia JPQL.
java.util.List	<u>findWhere</u> (java.lang.Class clase, java.lang.String pClausulaWhere, java.lang.String pOrderBy) Finder generico que permite aplicar clausulas where y order by.
java.util.List	<u>findWhereT</u> (java.lang.Class clase, java.lang.String pClausulaWhere, java.lang.String pOrderBy) Finder generico que permite aplicar clausulas where y order by.
static javax.persistence.EntityManager	<u>getEntityManager</u> ()
void	<u>insertar</u> (java.lang.Object pObjeto) Almacena un objeto en la persistencia.
void	<u>mostrarLog</u> (java.lang.String nombreMe todo, java.lang.String mensaje) Metodo basico para mostrar mensajes de depuracion.