

**UNIVERSIDAD AUTÓNOMA METROPOLITANA
UNIDAD AZCAPOTZALCO**

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA

Licenciatura en Ingeniería en Computación

Reporte de Proyecto Terminal:
“SISTEMA GESTOR DE NUTRICION”

Alumno:

Nombre: Rivera Torres Israel
Matrícula: 204205028

Trimestre: 12-P

Fecha de entrega: 25/07/12

Objetivos Generales:

- Diseñar e implementar un sistema de información vía web que permita hacer el registro de pacientes, guardar el historial del seguimiento de los padecimientos y dietas propuestas para estos, además de realizar estadísticas.

Objetivos Particulares:

- Diseñar la base de datos relacional para almacenar la información necesaria para el sistema: **“SISTEMA GESTOR DE NUTRICION”**.
- Diseñar e implementar un módulo de autenticación para los usuarios del sistema a través de perfiles, considerando: administrador, paciente y nutriólogo.
- Diseñar e implementar un módulo que permita a los usuarios entrar como invitados para realizar una consulta, sin que se guarde un historial.
- Diseñar e implementar un módulo que permita a los pacientes, registrarse, consultar su historial y actualizar sus datos para la generación de una nueva dieta.
- Diseñar e implementar un módulo que genere reportes de las estadísticas de los usuarios, considerando la edad, los padecimientos y tipos de dietas.
- Diseñar e implementar un módulo que genere reportes de la información histórica, en formato pdf para su posible impresión en caso que se requiera.
- Diseñar e implementar un módulo que permita la actualización, alta y baja de tipos de dietas.
- Diseñar e implementar un módulo que permita la actualización, alta y baja de tipos de enfermedades crónico degenerativas.
- Diseñar e implementar un módulo que permita la actualización, alta y baja de la pirámide nutricional.
- Diseñar e implementar un módulo que permita consultar en un catálogo, los tipos de alimentos y sus propiedades nutricionales.

Justificación:

La nutrición humana en el campo de la medicina, precisa de un conocimiento interdisciplinario que ayude a la prevención y tratamiento de la enfermedad. Una mala nutrición ocasiona diversos problemas a los individuos y aún más, complica la estabilidad del paciente con enfermedades crónicas u obesidad.

El objetivo de este proyecto es proporcionar a los pacientes una dieta personal tomando en cuenta una valoración global (antecedentes clínicos, peso corporal, etc), los requerimientos nutricionales, esto es, la cantidad de alimentos necesarios para mantener un estado de equilibrio en el adulto y asegurar un crecimiento adecuado en el niño, los requerimientos energéticos y los grupos de alimentos según los criterios variables y la pirámide nutricional (Figura 1).

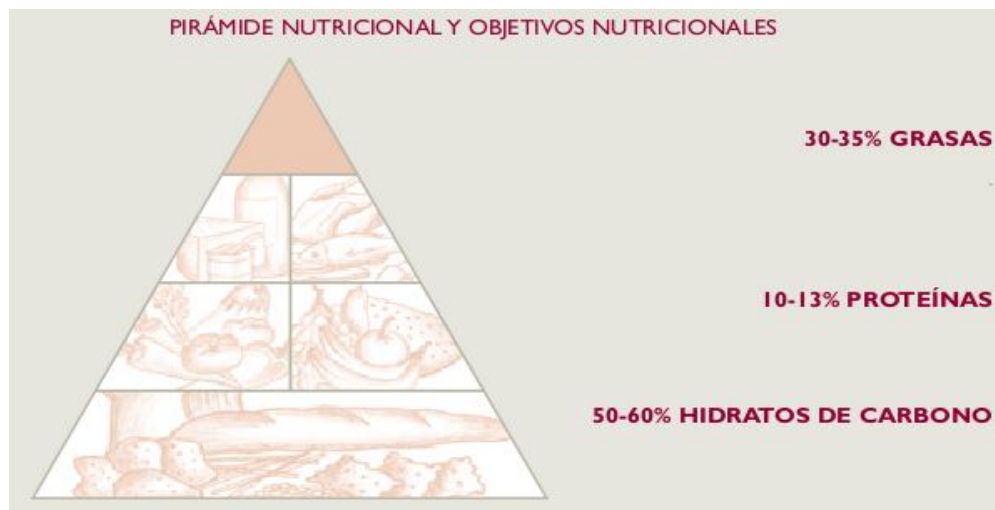


Figura 1: Pirámide nutricional

Descripción Técnica:

El proyecto consiste en un sistema de información que permite a los pacientes generar dietas en base a sus padecimientos y sus problemas de nutrición, consultar y modificar el historial de padecimientos.

En la figura 2 se muestra el diagrama general de casos de uso, que describe los roles que se manejarán en el sistema, asociados a algunas de las tareas principales.

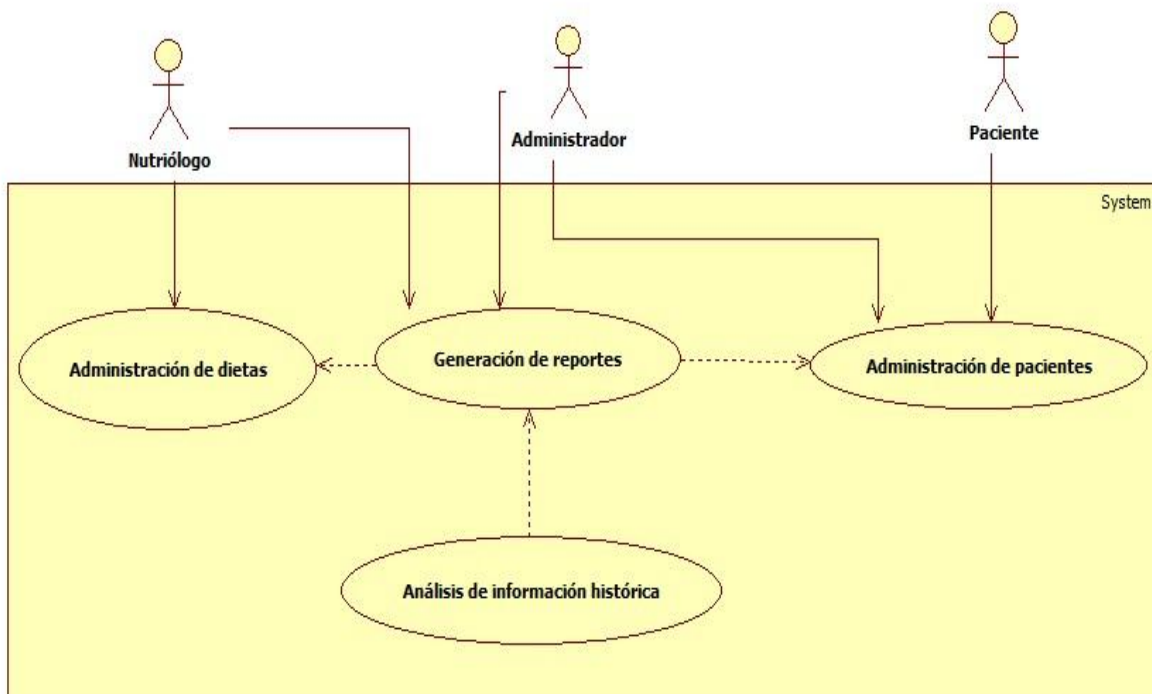


Figura 2: Diagrama General de Casos de Uso del Sistema

Diseño e implementación de siguientes módulos:

- **Módulo 1.** Diseño e implementación de la autenticación de usuarios del sistema a través de perfiles. El sistema considera los perfiles: administrador, paciente y nutriólogo. El administrador tendrá acceso a todos los módulos que integran el sistema. El paciente podrá acceder su información, modificarla y verificar el historial de dietas asignadas. El nutriólogo podrá actualizar la información de la pirámide nutricional, de los tipos de alimentos y sus propiedades nutricionales, adicionalmente podrá incluir las dietas tipo dentro del sistema.

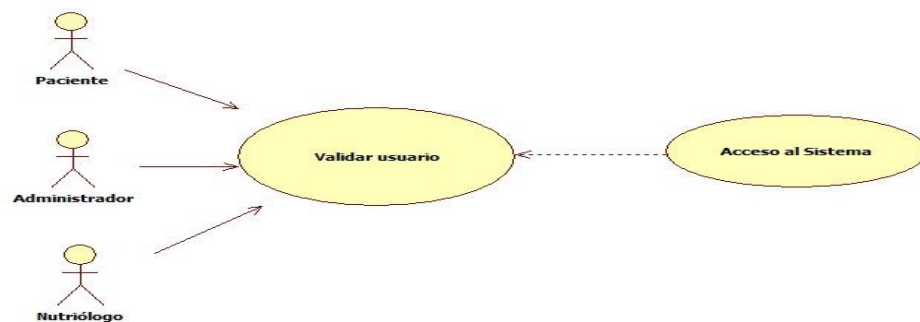


Figura 3: Diagrama de Casos de Uso del módulo 1

- **Módulo 2.** Diseño e implementación del acceso de usuarios invitados. En éste módulo los usuarios que no estén interesados en almacenar el historial de accesos, podrán ingresar a través de ésta opción, proporcionando sus datos y generando su dieta.

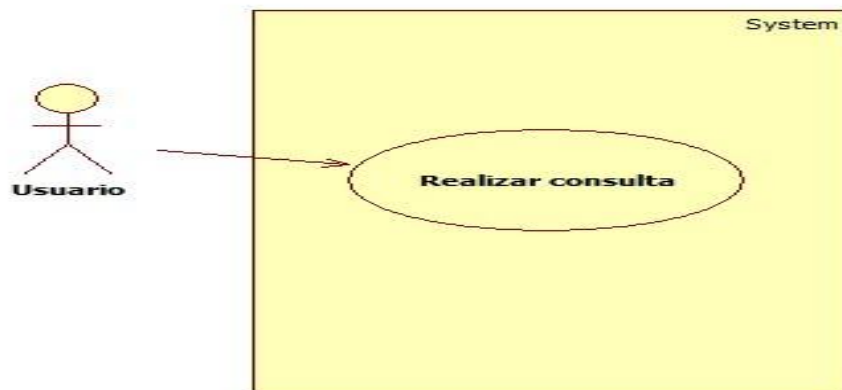


Figura 4: Diagrama de Casos de Uso del módulo 2

- **Módulo 6.** Diseño e implementación de la gestión de enfermedades crónico degenerativas. En este módulo se podrán dar de alta, realizar modificaciones y dar de baja las principales enfermedades crónicas degenerativas como: hipertensión, diabetes, entre otros.

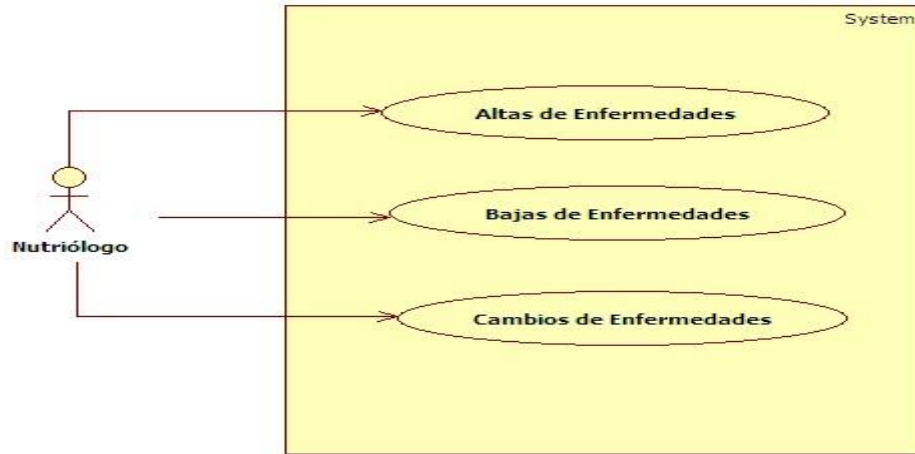


Figura 5: Diagrama de Casos de Uso del módulo 6

Requerimientos funcionales:

- Interfaz de Usuario:** Esta interfaz permitirá a un cliente de Internet basado en HTML tener acceso a toda la funcionalidad del sistema.
- Base de Datos Relacional:** Usada como repositorio de la información estructurada.
- Autenticación de los Usuarios del Sistema a Través de Perfiles:** El sistema contempla 3 perfiles: administrador, paciente y nutriólogo. El administrador tendrá acceso a todos los módulos del sistema. El paciente podrá modificar y consultar su información, una vez que se registró en el sistema. El nutriólogo podrá generar reportes y administrar la información de las dietas..
- Administración de Enfermedades:** Este módulo permite la actualización, alta y baja de tipos de enfermedades. El nutriólogo y administrador podrán tener acceso a este módulo.

Especificación Técnica:

El sistema se desarrollará en una plataforma abierta, utilizando como IDE¹: Eclipse² sobre sistema operativo Linux. Para garantizar la portabilidad del sistema, se desarrollará sobre plataforma Java, utilizando una interfaz web.

El sistema integra servidores de aplicaciones abiertos como Tomcat³, así como bases de datos relacionales abiertas como Postgres⁴.

Para agilizar el desarrollo se utilizarán *frameworks*⁵ como Spring⁶, Hibernate⁷ y Struts⁸.

Se plantea utilizar el estándar J2EE 1.4 como base, con el fin de poder ser montado a futuro sobre diferentes servidores de aplicaciones.

¹ **IDE.**- (Integrated Development Environment - Entorno integrado de desarrollo). Aplicación compuesta por un conjunto de herramientas útiles para un programador.

² **Eclipse.**- Es una plataforma de desarrollo integrado que puede ser usada para crear diversas aplicaciones como sitios de internet.

³ **Tomcat.**- Es un servidor web con soporte de servlets y JSPs.

⁴ **Postgres.**- Sistema de gestión de base de datos relacional orientada a objetos y libre.

⁵ **Framework.**- Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado, típicamente puede incluir soporte de programas, bibliotecas y un lenguaje.

⁶ **Spring.**- Es un framework de código abierto de desarrollo de aplicaciones para la plataforma Java.

⁷ **Hibernate.**- Es una herramienta de Mapeo objeto-relacional para la plataforma Java.

⁸ **Struts.**- Es una herramienta de soporte para el desarrollo de aplicaciones Web bajo el patrón MVC bajo la plataforma J2EE.

La figura 6 muestra el diagrama a bloques de las tecnologías que se utilizarán para el desarrollo y ejecución del sistema.

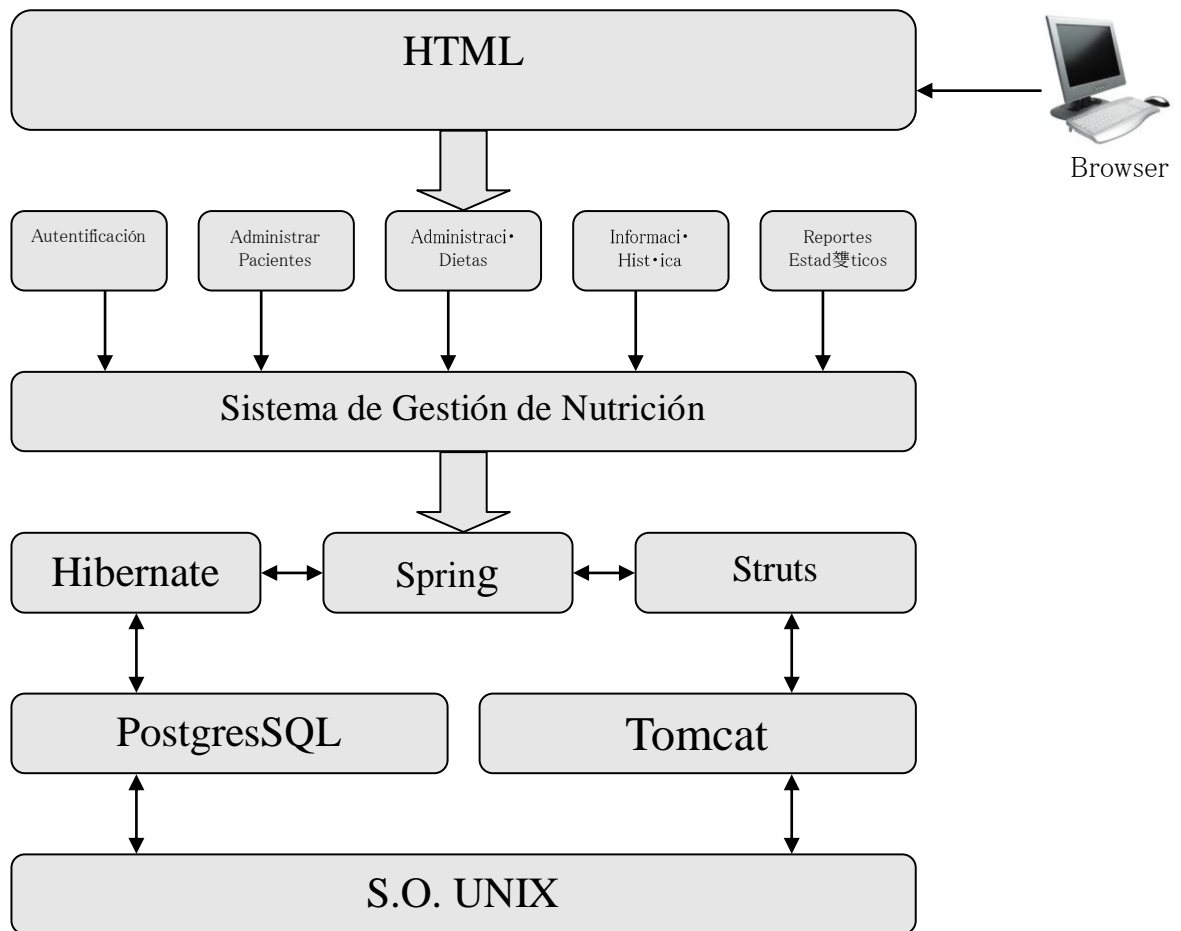


Figura 6: Diagrama de Bloques del Proyecto

La figura 6 muestra las partes generales que componen el sistema: El usuario se conecta al sistema mediante un navegador web utilizando HTML. Hace uso de los módulos ya mencionados para consultar y generar su información, el sistema realiza las peticiones dadas.

Tras la programación básica, será necesaria la implementación en *struts* para la plataforma web y el servidor Tomcat.

Utilizaremos Spring para la parte de servidores e Hibernate como intermediario en la parte de programación con la base de datos, la cual estará administrada con PostgreSQL, todo sobre un sistema operativo basado en Unix.

Licencia de Software Libre-Creative Commons.



- Reconocimiento (Attribution): En cualquier explotación de la obra autorizada por la licencia hará falta reconocer la autoría.
- Compartir Igual (Share alike): La explotación autorizada incluye la creación de obras derivadas siempre que mantengan la misma licencia al ser divulgadas.
- Reconocimiento - Compartir-Igual (by-sa): Se permite la distribución de las cuales se debe hacer una licencia igual a la que regula la obra original.

3.-Diccionario de Datos.

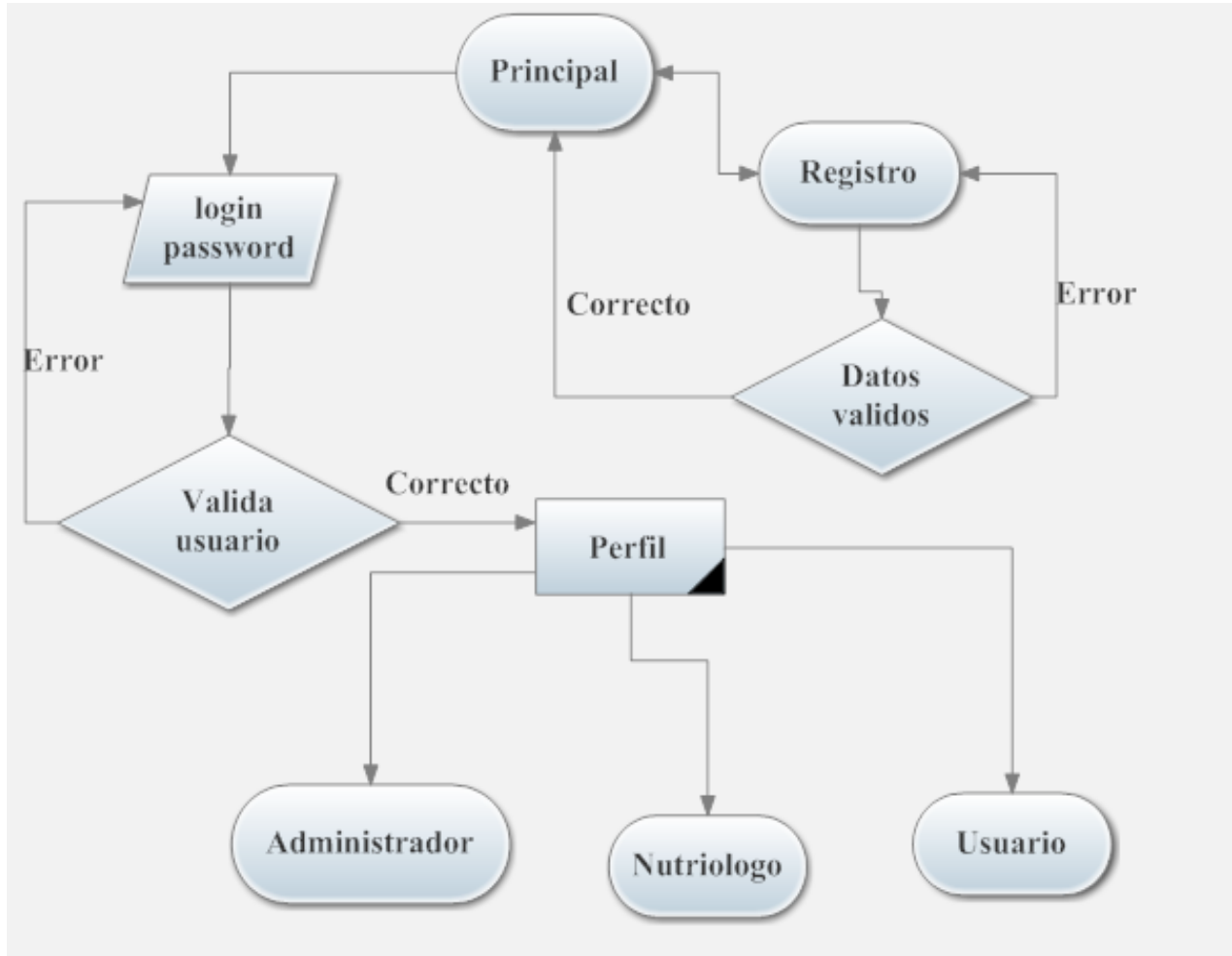
Tablas: perfiles, usuarios, enfermedades, síntomas, sugerencias.

Columna	Tipo	Llave	Descripción
id_perfil	Entero	primaria	Llave primaria de la tabla perfil, tipo de dato entero.
nombre	Cadena de caracteres		Cadena de caracteres para el nombre o descripción del perfil
permiso	Entero		Permiso para acceso al sistema, y determina el perfil del usuario.
id_usuario	Entero	Primaria	Llave primaria de la tabla usuarios, tipo de dato entero.
perfiles_id_perfil	Entero	Secundaria	Llave secundaria de la tabla perfil, tipo de dato entero.
usuario	Cadena de caracteres		Cadena de caracteres para el nombre del usuario dentro del sistema.
contrasenia	Cadena de caracteres		Cadena de caracteres para la contraseña del usuario dentro del sistema.
status	Cadena de caracteres		Cadena de caracteres para el control de acceso del usuario dentro del sistema.
correo	Cadena de caracteres		Cadena de caracteres para el correo del usuario dentro del sistema.
id_enfermedad	Entero	Primaria	Llave primaria de la tabla enfermedades, tipo de dato entero.
nombre	Cadena de		Cadena de caracteres para el

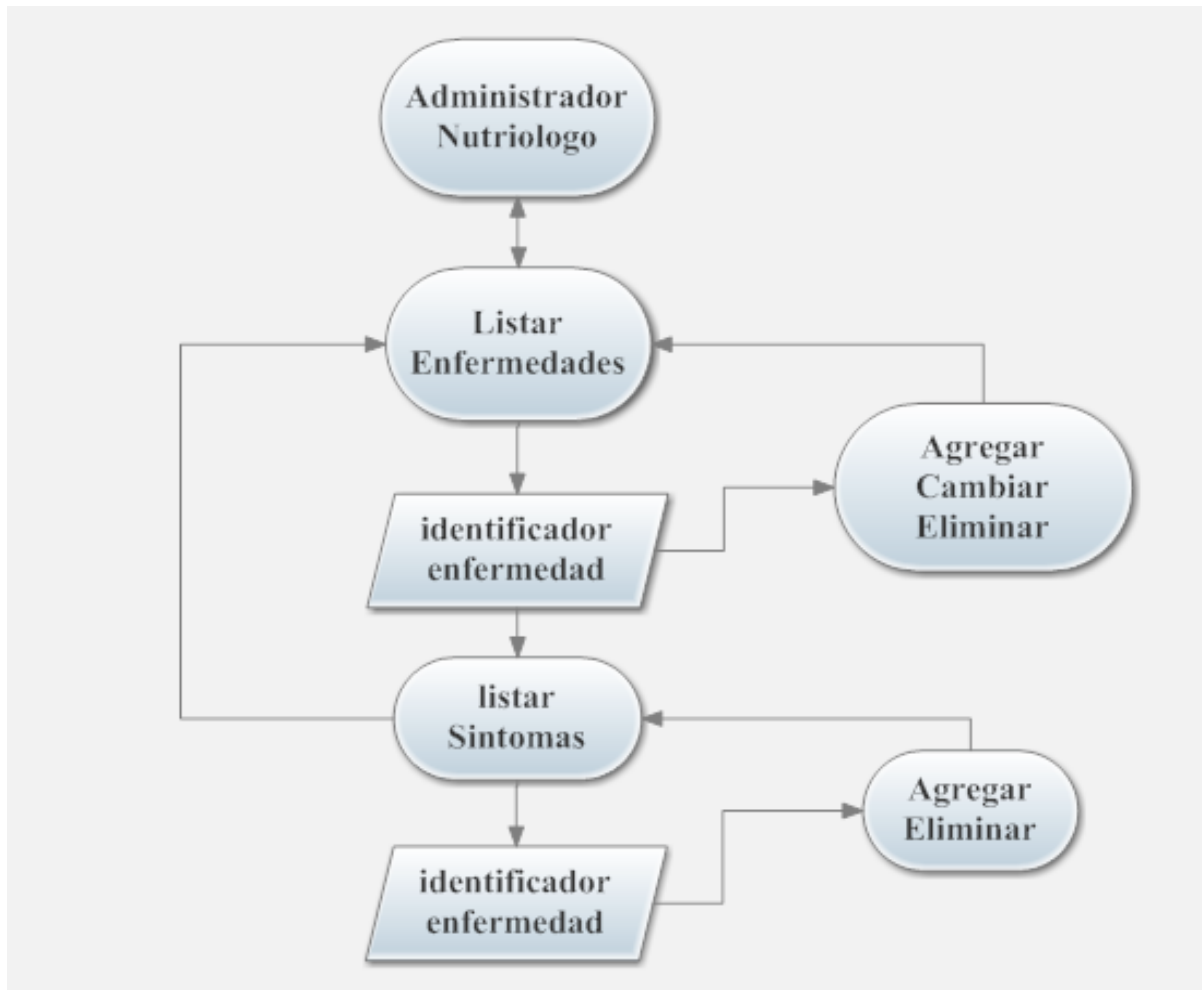
	caracteres		nombre de las enfermedades.
descripcion	Cadena de caracteres		Cadena de caracteres para la descripción de las enfermedades.
id_sintoma	Entero	Primaria	Llave primaria de la tabla síntomas, tipo de dato entero.
enfermedades_id_enfermedad	Entero	Secundaria	Llave secundaria de la tabla enfermedades tipo de dato entero.
nombre	Cadena de caracteres		Cadena de caracteres para el nombre de los síntomas.
observaciones	Cadena de caracteres		Cadena de caracteres para las observaciones de los síntomas.
id_sugerencia	Entero		Llave primaria de la tabla sugerencias, tipo de dato entero.
nusuario	Cadena de caracteres		Cadena de caracteres para el nombre del usuario. Que manda la sugerencia
sugerencia	Cadena de caracteres		Cadena de caracteres para la descripción de una sugerencia.
revisado	booleano		Marcador si la sugerencia ya esta revisada o no.

4.-Diagrama de Navegación.

Modulo 1 (Acceso al Sistema):

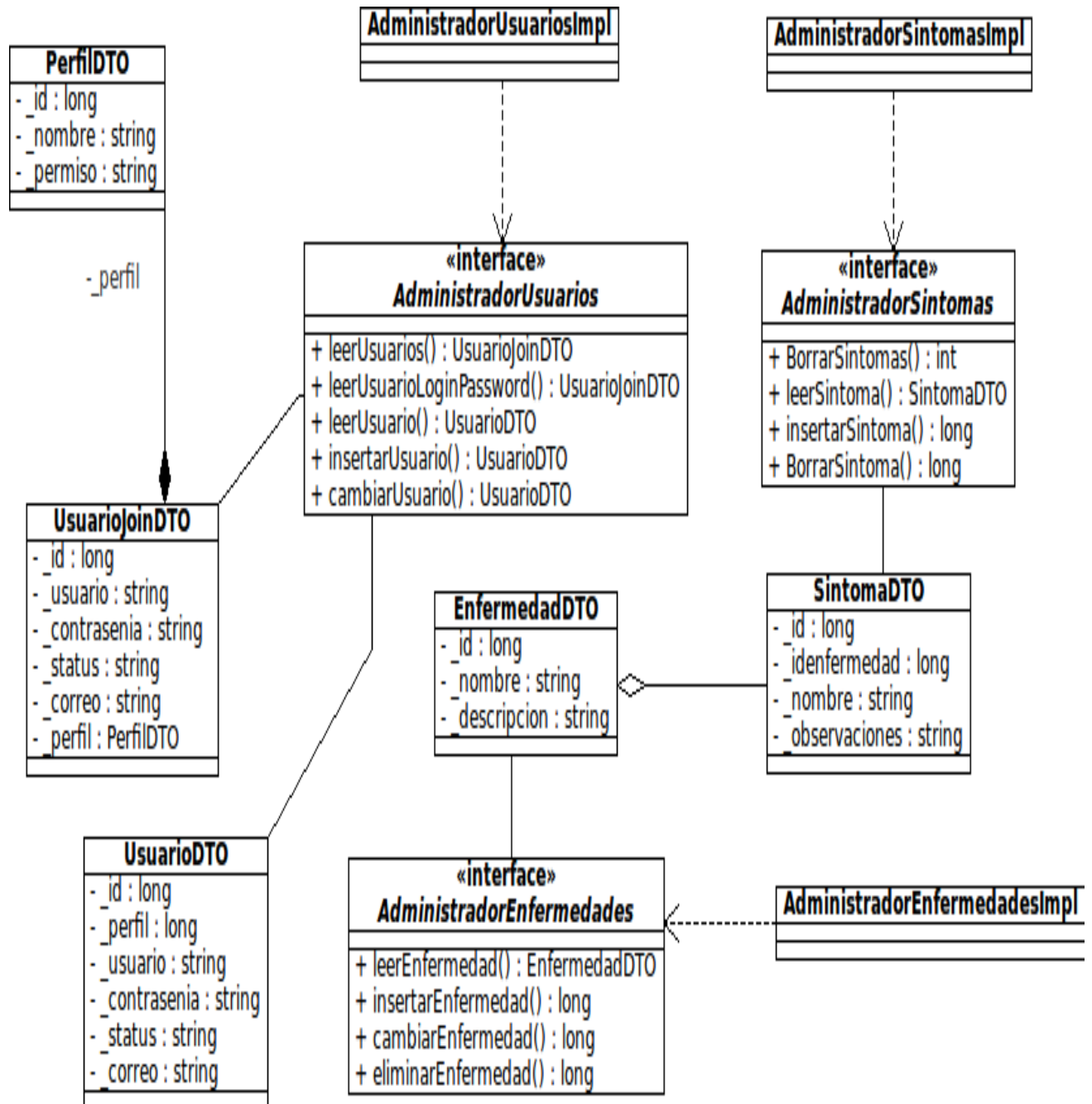


Modulo 6 (Administración de Enfermedades):



5.-Diagrama de clases.

Anexo en el archivo diagramaDeClases.png



6.-Manual de usuario.

Acceso al sistema:

Formas de acceso al sistema a través de perfiles:

1. Administrador: tiene acceso total a la administración del sistema en este caso se podrá:
 1. Solicitar dieta
 2. Ver historial
 3. Generar estadísticas
 4. Administrador dietas.
 5. Administrar Enfermedades.
 6. Administrar usuarios.
 7. Administrar Alimentos.
2. Nutriólogo: tiene acceso limitado a la administración del sistema en este caso se podrá
 1. Solicitar dieta
 2. ver historial
 3. generar estadísticas
 4. Administrador dietas.
 5. Administrar Enfermedades.
 6. Administrar Alimentos.
3. Usuario:
 1. Solicitar dieta
 2. ver historial
 3. generar estadísticas

6 a) *Acceso al sistema:*

1. Ingresar usuario y contraseña:

1. En caso que exista el usuario y la contraseña sean valida el sistema dará acceso de acuerdo a al perfil mostrando información personal y posibles acciones.
2. En caso que no exista el usuario o la contraseña no sea válida no dará acceso al sistema.
3. En caso que no se tenga usuario o registro el usuario interesado se podrá registrar pulsando el enlace REGISTRARME.

1. Se mostrara un formulario para proporcionar los datos del registro.

2. Se pedirá 2 veces la contraseña para confirmarla.

1. En caso que sean diferentes se mostrará un mensaje y para hacer coincidir las contraseñas proporcionadas.

3. Se pedirá un correo electrónico y debe cumplir con la sintaxis de un correo de la siguiente forma:

<correo>@<dominio del servidor>.<com>

Ejemplo: Nutricion@hotmail.com

4. En caso que no se proporcione un dato obligatorio se mostrara un mensaje para ingresarlo.

5. Si el usuario ya está registrado. Se mostrara un mensaje para cambiar la información. En este caso se deberá cambiar el usuario.

6. Cuando los datos sean validos y el usuario no esta registrado, el sistema da de alta el registro y re direccionara a la página principal. Y poder entrar al sistema como un usuario nuevo.

6 b) Administración de Enfermedades:

1. Para administrar enfermedades

1. Ingresar al sistema como administrador o nutriólogo.

2. Dar clic en Administración de enfermedades

1. Aparecerá la lista de las enfermedades registradas con las opciones siguientes:

1. Listar enfermedades: listara las enfermedades registradas asi como sus datos.

2. Agregar enfermedades: muestra la opción para capturar los datos de una enfermedad y al presionar el botón aceptar se guardara y re direcciona a la lista de enfermedades.

3. Cambiar enfermedad: al presionar esta opción se mostrara el formulario con la información de la enfermedad y opción para cambiar datos. Al presionar aceptar se redirección a la lista de enfermedades.

4. Eliminar enfermedad: Al dar clic en la opción eliminar se elimina la enfermedad asi como sus datos adicionales y muestra la lista de las enfermedades restantes.

5. Mostrar datos: al dar clic en esta opción muestra una lista de datos (síntomas) asociados a la enfermedad y permite administrar dichos síntomas con las opciones agregar y eliminar.

7.- Documentación de referencia: manual de instalación y configuración.

Para la instalación de el proyecto se requiere tener la instalación y configuración de los siguientes paquetes software:

1. Máquina virtual de java JRE
2. Eclipse índigo:
 1. Descargar el paquete software de la página oficial
 2. mover el paquete descargado al directorio deseado
 3. Descomprimir el paquete usando el comando:

```
sudo tar xvf eclipse-jee-indigo-linux-gtk.tar.gz
```

4. Dar todos permisos de ejecución con el comando

```
chmod -R 0777 *
```

3. Apache tomcat 6.4

1. Descargar el paquete software de la pagina oficial
2. Crear el directorio /usr/share/tomcat/ usando el comando:

```
sudo mkdir /usr/share/tomcat/
```

3. Mover el paquete descargado al directorio /usr/share/tomcat/ usando el comando :

```
sudo mv "apache-tomcat-6.0.33.tar.gz" /usr/share/tomcat/
```

4. Descomprimir el paquete usando el comando:

```
sudo tar xvf /usr/share/tomcat/apache-tomcat-6.0.33.tar.gz
```

5. Dar todos permisos de ejecución con el comando:

```
sudo chmod -R 0777 /usr/share/tomcat/*
```

4. PostgreSQL

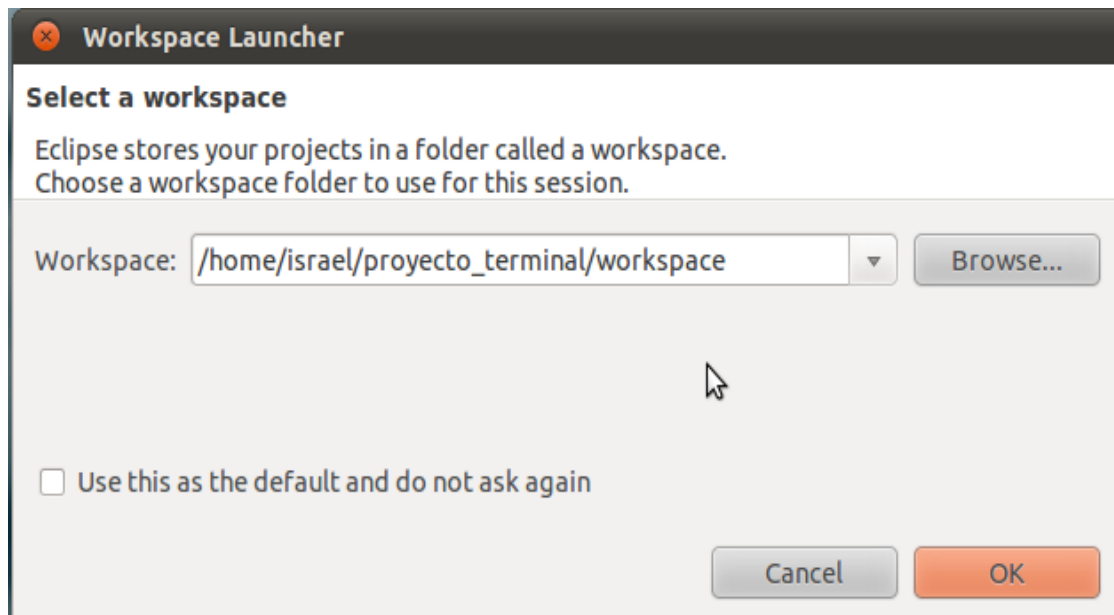
1. Instalar PostgreSQL desde repositorios:
 1. verificar puerto 5432
 2. asignar password.

8.-Configuración del Entorno de Desarrollo Y servidor de Aplicaciones.

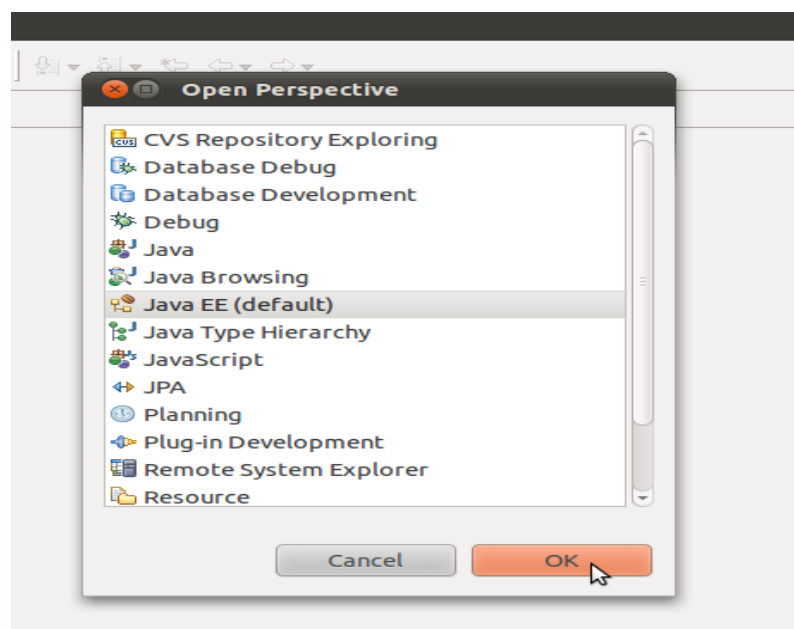
Esta opción es usada para desarrollo.

Poner en marcha eclipse:

1. Asignar espacio de trabajo workspace.

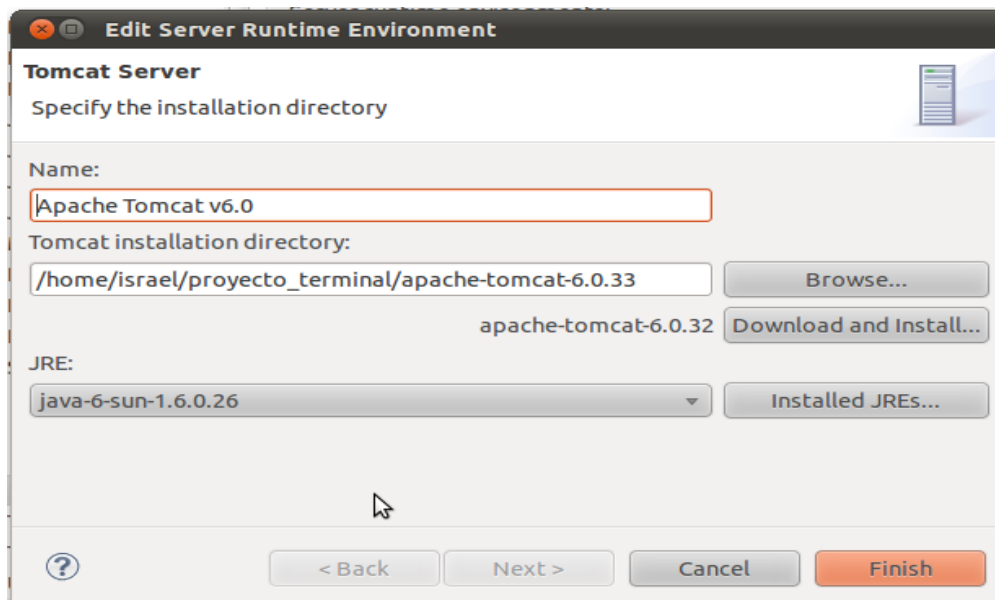


2. Cambiar perspectiva (vista) a JEE.



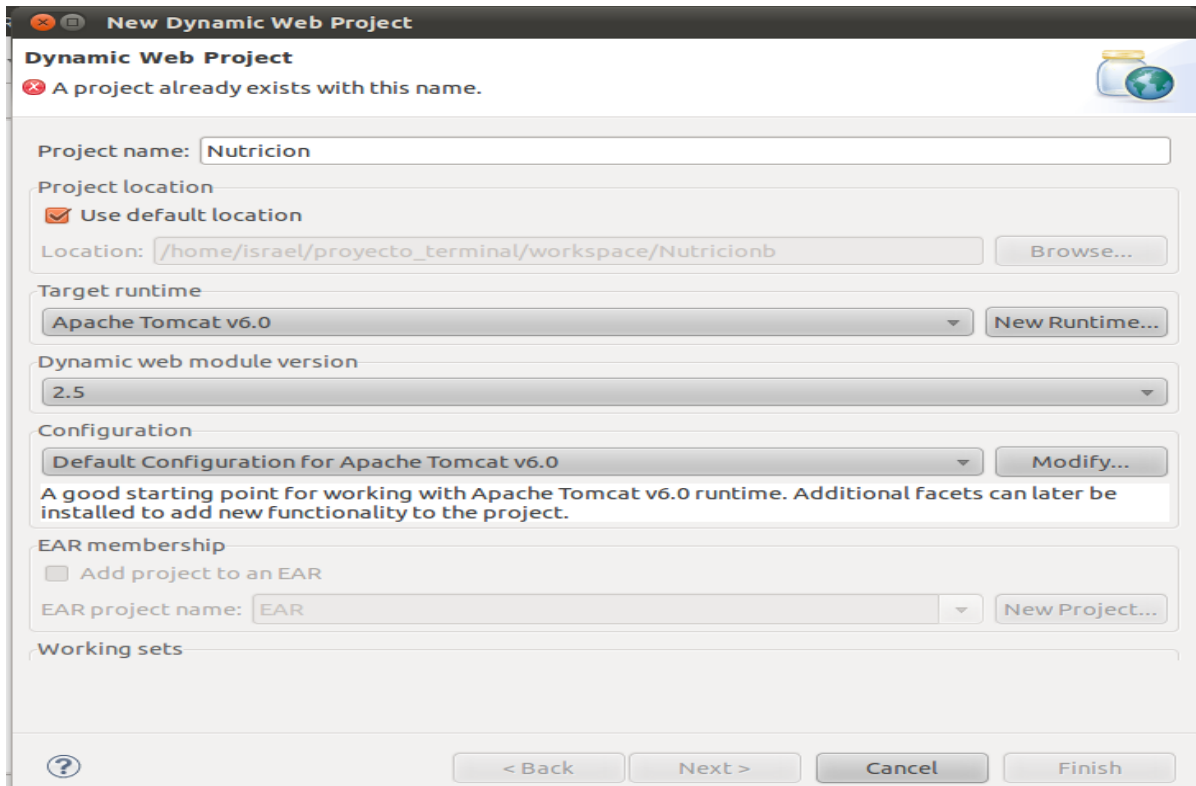
3. Crear servidor (server)

1. clic en server
2. clic secundario en parte sombreada, clici en server, clic en new.
3. Elegir Tomcat v6.0 Server
4. Configurar tomcat de la siguiente forma:



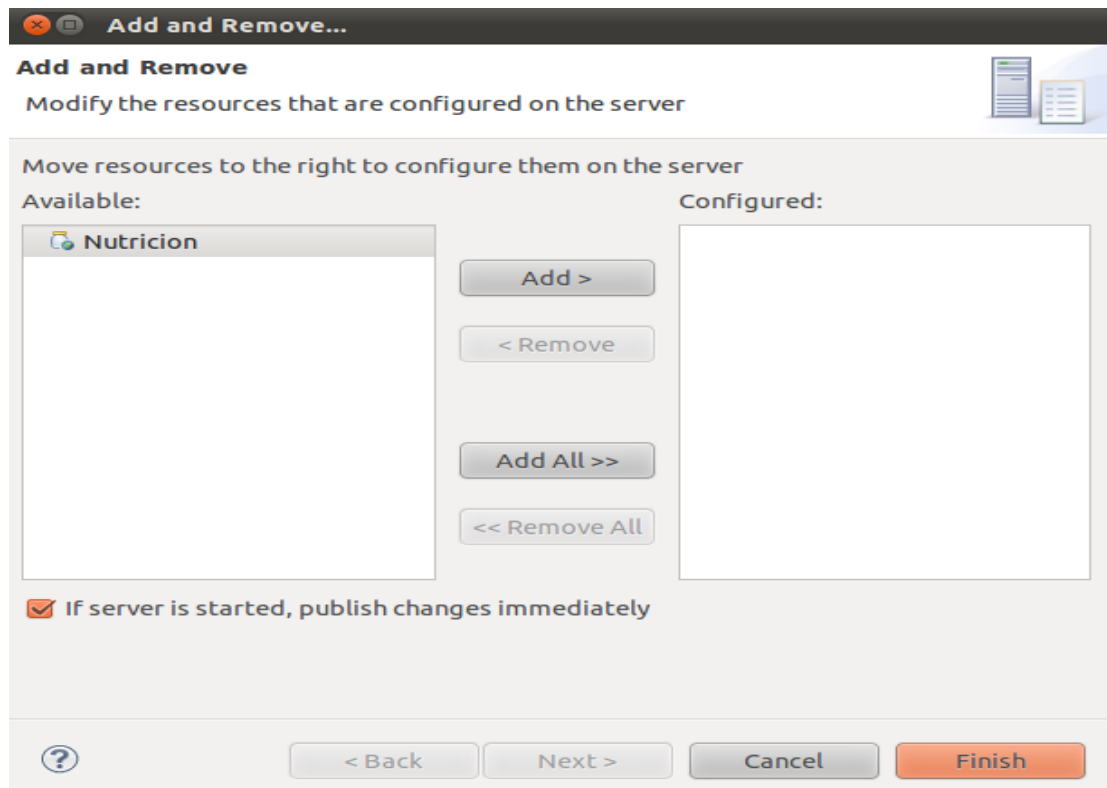
4. Crear proyecto:

1. Clic secundario en la ventana Project .
2. Clic en new.
3. Clic Dynamic Web Project.
4. Dar nombre al proyecto en este caso Nutrición.
5. Configurar el proyecto de la siguiente forma:



5.- Agregar proyecto al servidor de aplicaciones.

1. Clic secundario sobre el servidor
2. Clic en Add and Remove.



3. Dar Clic en ADD
4. Dar Clic en Finish

Con esta configuración se debe poder iniciar la edición del código fuente de la aplicación. Así como hacer pruebas usando el servidor.

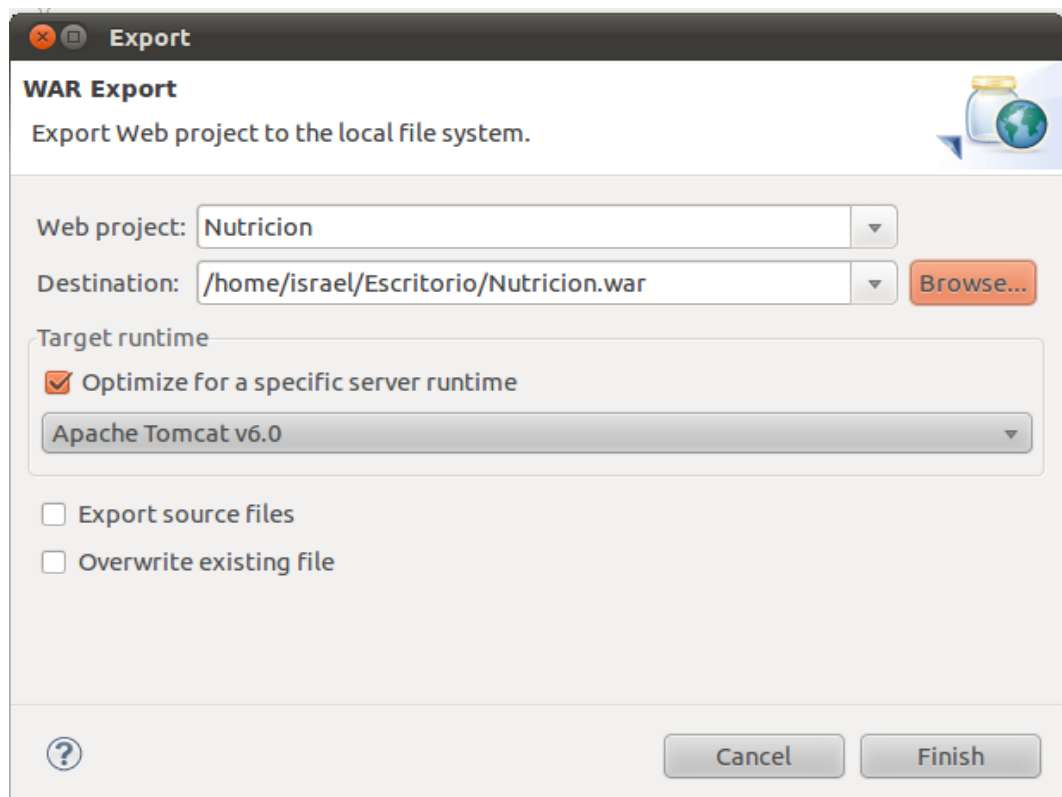
9.- Instalación y configuración en el servidor indicado por el asesor.

Esta opción: es usada para subir la aplicación al servidor sin necesidad de usar entorno de desarrollo.

Es decir cuando la aplicación esta lista se agrega al servidor para poder hacer uso de ella.

Instalación mediante un archivo WAR:

1. Crear el archivo WAR usando eclipse
 1. Clic sobre el Nombre del proyecto
 2. Clic sobre export
 3. Clic WAR file
 4. Elegir directorio destino.



En este caso el directorio destino es: */home/israel/Escritorio/Nutricion.war*

2. Si se desea cerrar eclipse
3. Abrir terminar.
4. Asegurarse que el servidor de aplicaciones este apagado:

1. Para Apagar el servidor ejecutar el comando:

sudo /etc/init.d/apache2 stop

2. Para cerrar el puerto 8080 ejecutar el comando:

sudo fuser -k 8080/tcp

5. Mover el archivo Nutricion.war al directorio */usr/share/tomcat/webapps* usando el comando:

sudo mv Nutricion.war "/usr/share/tomcat/webapps"

6. levantar el servidor tomcat usando el comando:

sudo sh /usr/share/tomcat/bin/startup.sh

Con esto el servidor descomprime el archivo *Nutricion.war* y lo optimiza para usarlo con en el servidor, en este momento se puede tener acceso a toda la funcionalidad de el sistema.

7. Para probar el funcionamiento de el sistema.
8. Abrir navegador y teclear la URL:
http://localhost:8080/Nutricion/login_forma_login.action
 1. observación: en este caso se usa localhost ya que el servidor esta en el mismo host, si se desea cambiar el host servidor solo se cambia localhost por el ip del host.
9. En este caso debe mostrar la página inicial del sistema.

Recursos:

Para el diseño de este proyecto se cuenta con el siguiente software:

- PostgreSQL. Manejador de Base de datos de software libre.
- JDK 6u13. Kit de desarrollo Java.
- Apache-tomcat-6.0.18. Servidor web de licencia libre.
- Eclipse 3.4.2.- Entorno de desarrollo integrado de código abierto multiplataforma.

El software que se utilizará en la elaboración del proyecto es de licencia libre, por lo tanto no se requiere la compra de las licencias de dicho software.

Los recursos de hardware que se utilizarán durante la elaboración del proyecto son los siguientes:

- **Laptop TOSHIBA SATELITE** con las siguientes características:
 - Procesador AMD 2.2 GHz
 - Memoria RAM 2 GB
 - 250 GB de HD
 - Sistemas Operativos: Windows Seven (profecional) y Linux Ubuntu 10.4, dual boot.

Bibliografía

- [1] “Manual básico de nutrición clínica y dietética” /A. Mesejo Arizmendi/ Generalitat Valenciana, pág. 36
- [2] <http://eclipse.org>
- [3] <http://tomcat.apache.org/>
- [4] <http://www.postgresql.org/>
- [5] <http://www.springframework.org/>
- [6] <http://www.hibernate.org/>
Consultada: 21/OCT/2010
- [7] <http://struts.apache.org/>