

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

INSTITUTO DE INGENIERÍA

MAESTRÍA EN CIENCIAS EN INGENIERÍA DE SISTEMAS



“DESARROLLO DE UN SISTEMA DE INFORMACIÓN INTERACTIVO
PARA LA SELECCIÓN Y DISEÑO DE PRÓTESIS DE CADERA”

T E S I S

QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS EN INGENIERÍA DE SISTEMAS

PRESENTA

SALVADOR FLORES FRIAS

MEXICALI, B.C. SEPTIEMBRE DEL 2006

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

INSTITUTO DE INGENIERÍA

MAESTRÍA EN CIENCIAS EN INGENIERÍA DE SISTEMAS



“DESARROLLO DE UN SISTEMA DE INFORMACIÓN INTERACTIVO
PARA LA SELECCIÓN Y DISEÑO DE PRÓTESIS DE CADERA”

T E S I S
QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS EN INGENIERÍA DE SISTEMAS

PRESENTA
SALVADOR FLORES FRIAS
DIRECTOR DE TESIS
DR. BENJAMÍN VALDEZ SALAS

MEXICALI, B.C. SEPTIEMBRE DEL 2006

Agradecimientos

Existen infinidad de personas que durante el transcurso de mi vida la han marcado, algunos en ocasiones de no muy buena manera, pero gracias a esas personas e crecido, sin embargo mi intención nunca será el del reclamo si no el del agradecimiento.

Quisiera tener el tiempo y la suficiente humildad para darles las gracias personalmente a todas esas personas que siempre creyeron en mí, quisiera empezar por los dos seres más maravillosos que he conocido en este mundo.

A mis padres, que con su paciencia, apoyo y consejo supieron hacer de mi el hombre que ahora soy, ya que ellos siempre me mostraron su amor incondicional sobre todo en los momentos mas difíciles de mi vida, quisiera no solo agradecerles si no decirles que los amo y que cada día que dios me permite tenerlos a mi lado es un día de total felicidad, que dios los bendiga y me permita seguir disfrutando de su compañía.

Quisiera agradecer a toda mi familia, en especial a mi hermana Maria que por su incondicional apoyo y amor.

Y por su puesto a mi esposa Verónica que ha sido un apoyo muy importante durante todo este tiempo, soportando mi mal humor y frustraciones, demostrándome un amor difícil de encontrar en estos tiempos.

Quisiera dar las gracias a la Universidad Autónoma de Baja California por su apoyo para la realización de la maestría, a todos los maestros buenos o malos, todos dejaron una enseñanza en mí, a las secretarias Dolores y Normita por su tremendo apoyo.

A todos mis sínodos que con su apoyo para la realización de este trabajo y en especial al Dr. Benjamín Valdez Salas por creer en mi y agradezco su confianza para la realización de este trabajo.

Gracias a todos los mencionados y los olvidados, Gracias.

DESARROLLO DE UN SISTEMA DE INFORMACIÓN INTERACTIVO PARA LA SELECCIÓN Y DISEÑO DE PRÓTESIS DE CADERA

CONTENIDO

Agradecimientos	i
-----------------------	---

Capítulo I Introducción

1.1 Introducción	1
1.2 Planteamiento del problema	11
1.3 Objetivo General	14
1.4 Metas	14

Capítulo II Metodología

2.1 Metodología	15
2.2 BioM-BD: base de datos biomateriales	15
2.3 BioCH-BD: base de datos de biocompatibilidad del Humano	15
2.4 RP-M3D: representación de las prótesis en su modelo tridimensional	16
2.5 SIM-PTC: sistema de Integración de Modelado de Prótesis de total cadera.....	16
2.6 I-WEB: Interfase de portal de Internet	16

Capítulo III Desarrollo experimental

3.1 Desarrollo experimental	18
3.2 BioM-BD: base de datos biomateriales	18
3.3 BioCH-BD: base de datos de biocompatibilidad del Humano	26
3.4 RP-M3D: representación de las prótesis en su modelo tridimensional	29
3.5 SIM-PTC: sistema de Integración de Modelado de Prótesis de total cadera.....	37
3.6 I-WEB: Interfase de portal de Internet	47

Capítulo IV Resultados

4.1 Resultados	48
----------------------	----

Conclusiones.....	56
-------------------	----

Recomendaciones	57
-----------------------	----

Referencias bibliográficas	58
----------------------------------	----

Anexo

Código fuente desarrollado en Lingo	60
---	----

Lista de figuras

Figura 1-1. Comparación de Prótesis de Cadera Sana y Cadera con Artritis	11
Figura 3-1. Cadera Omnifit HA	29
Figura 3-2. Cótulo osteonics	30
Figura 3-3. Cadera Omnifit-C	31
Figura 3-4. Liner constreñido	32
Figura 3-5. Componentes generados con la herramienta CAD, correspondiente al modelo <i>Cadera Omnifit C</i>	33
Figura 3-6. Representación en milímetros de las piezas que componen la PTC, modelo <i>Cadera Omnifit C</i>	34
Figura 3-7. Componentes generados con la herramienta CAD, correspondiente al modelo <i>Cadera Omnifit HA</i>	35
Figura 3-8. Representación en milímetros de las piezas que componen la PTC, modelo <i>Cadera Omnifit HA</i>	36
Figura 3-9. Representación tridimensional de la prótesis <i>Cadera Omnifit C</i> , dentro del ambiente del lenguaje de programación Lingo.	40
Figura 3-10. Representación tridimensional de la prótesis <i>Cadera Omnifit HA</i> , dentro del ambiente del lenguaje de programación Lingo.	41
Figura 3-11 Posición de cámaras	43
Figura 3-12 Vista de las tres cámaras	44
Figura 3-13 Barra de Navegación	46
Figura 3-14 Unión de los módulos	47
Figura 4-1 Resultado final de los procesos desarrollados en el módulo I-WEB ...	55

Lista de diagramas

Diagrama 1. Representación de los módulos que compone la herramienta I-WEB	17
Diagrama 2. Representación de las clases utilizadas en módulo BioM-BD	19
Diagrama 3. Clases y relaciones entre las clases	28
Diagrama 4. Muestra las clases y sus relaciones con los modelos tridimensionales	38
Diagrama 5. Procesos del módulo I-WEB	49

Lista de tablas

Tabla 3-1. Características del paciente para la selección de prótesis de cadera total.	26
--	----

Capítulo I

1.1 Introducción

Una prótesis es un dispositivo que sirve como extensión de alguna extremidad o parte del cuerpo, para reemplazar un hueso o articulación que ha sido dañado o con el objeto de establecer o reestablecer alguna función.

¿Por qué es necesaria una prótesis total de cadera?

La razón más frecuente para colocar una prótesis total de cadera es la de eliminar el dolor y la incapacidad que causa una destrucción severa de la articulación. Las superficies de la articulación pueden estar dañadas por la artrosis, proceso en el que se produce el desgaste del cartílago articular. La articulación también puede resultar dañada por la artritis reumatoide. En esta enfermedad, la membrana sinovial produce unas sustancias químicas que atacan la superficie articular y destruyen el cartílago articular.

Debido al dolor y a la rigidez, el paciente evita la utilización de la articulación, los músculos que rodean la articulación se debilitan y esto contribuye a dificultar la movilidad de la cadera. Cuando la destrucción articular es severa, una prótesis total de cadera puede permitir al paciente volver a desempeñar sus actividades cotidianas.

El diagnóstico de la artrosis o de la artritis reumatoide consiste en anotar los síntomas del paciente, realizar una exploración física, pruebas de laboratorio y radiografías que muestre la extensión de la destrucción articular.

La posibilidad de colocar una prótesis se considera después de que el tratamiento no quirúrgico u otras alternativas hayan fallado sin conseguir eliminar el dolor y la incapacidad.

Existen prótesis diseñadas para diferentes partes del cuerpo: ojos, brazos, manos, piernas, cadera, rodillas, dientes o pene. Su uso, ha permitido a miles de personas mejorar su calidad de vida y reintegrarse a sus actividades de forma normal. La cadera está integrada principalmente por los huesos de la pelvis y la cabeza del fémur y es una parte del cuerpo muy importante para su sostén, movimiento y equilibrio. Pero hay varias razones por las que puede lastimarse o dañarse y dejar de funcionar, lo que además es acompañado por dolores muy fuertes, incapacidades y un fuerte deterioro en la calidad de vida.

La era moderna de los reemplazos articulares se inicia en los años 60, con la divulgación de la artroplastía de cadera de Charnley, quien publicó en la revista Lancet su experiencia con una prótesis de vástago femoral metálico y una copa acetabular de polietileno de alta densidad, fijando ambos componentes al hueso con metilmetacrilato (cemento ortopédico). Más tarde en los años 70 se

desarrolló el reemplazo articular de rodilla y posteriormente aparecieron los reemplazos protésicos de hombro, codo, muñeca, tobillo y articulaciones de los dedos [1].

Las prótesis normalmente son fabricados en acero inoxidable o titanio, debido a que no generan rechazo por el cuerpo humano, sin embargo se exige a su vez en los aceros inoxidables contenidos máximos de carbono, manganeso, fósforo, azufre, molibdeno y cobre, para reducir riesgos, lo que a generado la aparición de aceros inoxidables de aplicación específica a la biomecánica [2].

En la actualidad las prótesis para la reconstrucción de partes dañadas del cuerpo humano, lejos de parecer ciencia ficción es una realidad que aplica a partes del cuerpo humano.

Más de cincuenta millones de personas en todo el mundo tienen implantado algún tipo de prótesis. Según los demógrafos de Naciones Unidas, dentro de unos cinco años es muy probable que haya mas personas mayores de 60 años que niños de menos de 15. Las prótesis están permitiendo una mejor calidad de vida que se refleja en la longevidad de las personas que usan alguna de ellas [3].

El diseño de prótesis esta relacionado estrechamente con:

- **Fabricación de Bio-materiales:**

Combinación de las palabras biología y materiales, estos bio-materiales deben de cumplir con las condiciones de ser biocompatibles y asegurar una determinada vida media. A su vez, tienen que aportar las características necesarias que sean requeridas para la aplicación que vayan a ser destinadas.

- **Diseño asistido por herramientas CAD-CAM, CAE, CNC:**

- o **CAD:** *Computer-Aided Design* (Diseño asistido por computadora) es utilizado para generar modelos tridimensionales de las prótesis.
- o **CAM:** *Computer-Aided Manufacturing* (Fabricación asistida por computadora) en realidad CAD-CAM es un matrimonio entre numerosas disciplinas de ingeniería y fabricación.
- o **CAE:** son las siglas de *Computer-Aided Engineering* (Ingeniería asistida por computadora). Sus funciones incluyen diseño, análisis de simulación y diseño de procesos y herramientas.
- o **CNC:** *Computer Numerically Controlled* (Control numérico por computadora) es una técnica que incluye una computadora o microcontrolador dedicado al control de una máquina especializada para la fabricación de diversas piezas.

- **Elemento finito para el análisis de esfuerzos en las prótesis:**

Es un método numérico para resolver ecuaciones diferenciales que describan el comportamiento al ejercer esfuerzos mecánicos en las prótesis, permitiéndonos detectar posibles deficiencias (fatiga) en los materiales y/o diseño de los componentes.

- **Biomecánica:**

Combinación de las palabras biología y mecánica, tiene que ver con los principios y métodos de la mecánica aplicados al estudio de las fuerzas, estructuras y funcionalidad de sistemas biológicos.

La combinación de todos estos campos permite en la actualidad generar estudios profundos sobre la fabricación de prótesis, compañías como *Va Lakeside Medical Center (Valmc)* Y *Northwestern University Prosthetics Research Laboratory (Nuprl)* de Estados Unidos de América, trabajan en el desarrollo de programas computacionales para el diseño de métodos de CAD-CAM para la fabricación de prótesis probando medidas experimentales y la generación de modelos de elemento finito, modelos CNC [4], que permitan un ahorro significativo en tiempo y dinero, durante la manufactura de los distintos dispositivos.

La creación de máquinas que reproduzcan con precisión la inserción de prótesis de articulación de cadera en el fémur para pruebas preclínicas *del Departamento de Ingeniería en Mecánica y Manufactura, del Colegio de Trinity en Dublín Irlanda*, permite analizar el comportamiento de esfuerzos de una prótesis insertada en el fémur, para conocer el grado de adhesión que las piezas tienen con el fémur [5].

La compañía PolyWare™ ha desarrollado un sistema computacional que determina con precisión el diseño de prótesis, a través de radiografías del paciente haciendo un análisis de materiales y el diseño a utilizar para la implementación de la prótesis [6].

El diseño de prótesis en la actualidad se basan en Diseño asistido por computadora “*Computer attented design*” (CAD por su siglas en ingles), permitiendo generar dos tipos de representación:

- Bidimensional: Dos planos, permitiendo mostrar dimensiones y ángulos planos carentes de interacción con el usuario.
- Tridimensional: Representación de los objetos en los tres planos, permitiendo mostrar tanto dimensiones de longitud como de profundidad logrando una mejor visualización de los objetos.

Existen estudios relacionados con la generación de prótesis de cadera como el proyecto Arom que consiste en la manufacturación de prótesis totales cementadas de cadera usando técnicas CAD/CAM y el objetivo de este proyecto de investigación es proponer un nuevo diseño para prótesis cementadas de cadera [7].

Una representación tridimensional del fémur con una prótesis cementada de cadera Charnley fue utilizada como geometría inicial para el análisis, donde el comportamiento asociado con cambios en parámetros geométricos en combinación con diferentes superaleaciones es analizado por el método de elemento finito.

El proyecto WEBD (Web based and training in the field of the biomedical and design engineering) por sus siglas en ingles, tiene módulos de aprendizaje de biomecánica y de ingeniería de diseño a través de Internet donde los gráficos tridimensionales e interactivos resultan el aspecto mas innovador y atractivos. El objetivo de este proyecto es demostrar el beneficio de la utilización de estas tecnologías en la enseñanza.

El proyecto WEBD surge de la colaboración de diferentes universidades europeas para la elaboración de material docente en los campos de la biomecánica y de la ingeniería de diseño donde existe una gran demanda de especialistas en el entorno europeo. Este proyecto está cofinanciado por el Programa Leonardo Da Vinci I-00-B-F-PP-120758 que apoya experiencias piloto en el campo de la formación profesional. La gran novedad que aporta este proyecto

es la utilización de realidad virtual de manera extensa en la elaboración de los cursos. La elaboración del contenido tridimensional interactivo y de gran parte del material docente elaborado de los módulos de ingeniería de diseño, así como la gestión del sitio Web ha sido realizado por el Grupo de Ingeniería Gráfica del Departamento de Ingeniería Mecánica y Fabricación de la Universidad Politécnica de Madrid [8].

En la actualidad, para todas aquellas aplicaciones clínicas que requieran soportar carga se están utilizando implantes, en su mayoría metálicos, lo que origina problemas importantes debido a:

- la gran diferencia de propiedades mecánicas entre el implante artificial y el hueso natural, lo que da lugar a rupturas.
- la presencia de iones que, procediendo del implante artificial, pueden ser tóxicos o perjudiciales y son causa de dolor.
- la imposibilidad de regenerar el hueso natural.

Una alternativa, para combatir en parte estos problemas mientras no se logre un material similar al hueso es recubrir el implante metálico con cerámicas. Esto se está realizando tanto en implantes dentales como en prótesis de cadera, aunque el camino que hay que recorrer es todavía muy largo para mejorar estos productos. El proceso de recubrimiento de un metal por una cerámica es complejo y existen muchos métodos para realizarlo, de él depende, en gran parte, el éxito clínico, ya que la calidad y la duración de la fijación en la interfaz dependen en gran medida de la pureza, tamaño de partícula, composición

química del recubrimiento, espesor de la capa y características superficiales del sustrato. Otra de las ventajas que se obtienen al recubrir un implante metálico por una cerámica es la reducción de la liberación de iones procedentes de la aleación metálica.

La cerámica constituye una verdadera y eficaz barrera que disminuye la cinética de difusión de iones metálicos al organismo vivo. Por esta razón, se está utilizando la hidroxiapatita, para mejorar la fijación de las prótesis articulares de cadera, dadas sus excelentes propiedades biológicas [9].

La hidroxiapatita: Efecto piezoeléctrico de los cristales óseos

Los cristales minerales del hueso están encajados en una matriz orgánica, los cuales constituyen aproximadamente las 2/3 partes del peso del hueso. La estructura de los cristales óseos se parece mucho a la estructura de la fluorapatita, un mineral que se encuentra en las rocas. En el cristal del hueso las posiciones de los átomos de flúor en la fluorapatita están ocupadas por grupos hidroxil (OH) por lo que recibe el nombre de hidroxiapatita.[9]

Son Llätzer (Hospital, Mallorca. España) utiliza plantillas digitales para planificar intervenciones ortopédicas. El programa está dotado de una serie de plantillas digitales que se superponen a la imagen radiológica del paciente, lo que permite realizar una valoración del tamaño y colocación de las prótesis de una manera más precisa que con el método tradicional. Esta herramienta calcula

automáticamente los ángulos, las inclinaciones, las medidas, los puntos de corte y otros parámetros necesarios para planificar una intervención quirúrgica.

Se han desarrollado modelos tridimensionales del método del elemento finito del sistema hueso-prótesis no convencional. Este implante se emplea para tratar tumores del tercio proximal del fémur, siendo la inestabilidad su principal desventaja. La geometría del fémur se estableció con sesenta cortes tomográficos, empleando el software Ansys versión 5.7. Posteriormente, se integró un modelo hueso-prótesis con 19,895 elementos y 32,154 nodos. Se estudiaron cuatro casos, en los primeros tres se consideró sólo la reacción de la cadera y se ubicó el perno proximal a 50, 25 y 75 mm del sitio de la osteotomía. Para todos los casos, el perno distal se colocó a 25 mm del proximal. Un cuarto caso consideró adicionalmente la acción del abductor mayor, con el perno proximal a 50 mm. Se encontró que entre más cerca se encuentren los pernos, los esfuerzos se incrementan. El caso 4 presentó los mejores resultados, seguido de 1, 2 y 3. Como la práctica actual no permite unir el abductor mayor al implante, el caso 1 es la mejor opción, no obstante, es promisorio dirigir esfuerzos de investigación para restablecer la palanca abductora y mejorar aún más la estabilidad del implante [10], este caso de estudio es uno de tantas investigaciones que se han hecho sobre las prótesis de cadera.

1.2 Planteamiento del Problema

La articulación de la cadera es comúnmente conocida como una cabeza y un recipiente articular y se forma donde la cabeza redondeada del hueso del muslo se junta con la pelvis. La articulación está rodeada y revestida por cartílago, músculos y tendones. Estos tejidos proporcionan soporte, estabilidad y facilitar el movimiento.

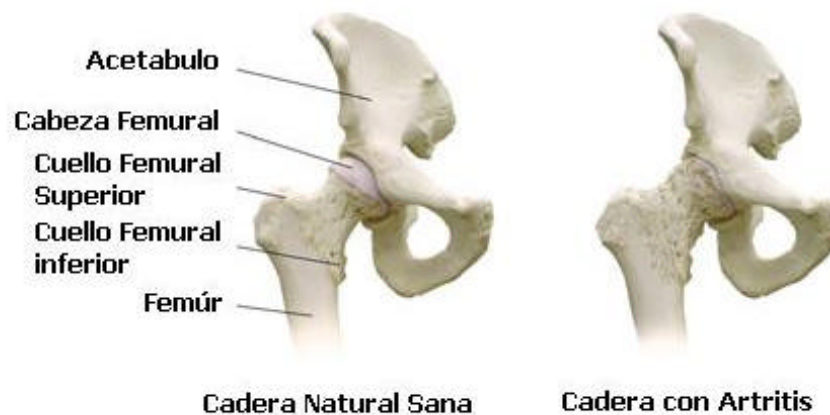


Figura 1-1. Comparación de Prótesis de Cadera Sana y Cadera con Artritis

La prótesis de cadera actual puede variar en tamaño y materiales de construcción debido a la complexión corporal del paciente así como sus características biológicas. El presente trabajo pretende desarrollar una herramienta que permita interactuar con los diferentes tipos de prótesis total de cadera de la compañía *Striker*. Las prótesis seleccionadas son: **Cadera Omnifit HA** y **Cadera Omnifit C**; la herramienta será capaz de manipular los modelos tridimensionales de las prótesis así como permitirle la total interacción

con las mismas al usuario, también podrá ser capaz de mostrar tanto el tipo de biomateriales y las características particulares de las cuales están compuestas los diferentes prótesis al igual que las características necesarias que debe cumplir un paciente para ser candidato a una operación de prótesis total de cadera.

Para poder cumplir con un programa modular con el siguiente contenido:

a) **BioM-BD (base de datos biomateriales):**

Base de datos con información de todos los tipos de materiales para la construcción de las prótesis *Omnifit HA* y *Omnifit C*, diseño y características las cuales incluya las dimensiones de las prótesis.

b) **BioCH-BD(base de datos de biocompatibilidad del Humano):**

Caracterización de los biomateriales usados en la fabricación de las prótesis para su correlación con las características biológicas del ser humano, basado en estudios existentes de aplicación de prótesis total de cadera exitosos. La caracterización de las medidas de la articulación cadera sana (pelvis y fémur) de determinada población. Las medidas pueden variar dependiendo del país en donde sea hecho el estudio, en este caso se hacen en la población mexicana.

c) **RP-M3D (representación de las prótesis en su modelo tridimensional):**

Representación de los modelos tridimensionales de las prótesis seleccionadas para estudio, para su manipulación y representación de su armado.

d) **SIM-PCT (Sistema de Integración de Modelado de Prótesis de cadera total):**

Integración de los módulos “BioM-BD, RP-M3D, BioCH-BD” que permitirán al sistema proponer posibles prótesis que cumplan con las necesidades del paciente tanto en aspectos de biocompatibilidad como en cuestiones dimensionales.

e) **I-WEB (Interfase de portal de Internet):**

La plataforma de trabajo será a través de un navegador de páginas de Internet, la cual permita acceso desde cualquier computadora en cualquier parte del mundo, las plataformas de desarrollo propuestas para este módulo son PHP combinado con Macromedia Lingo Shockwave utilizando JavaScript y HTML soportado por un servidor APACHE con Base de datos MYSQL en el sistema operativo Linux.

1.3 Objetivo General

Desarrollar herramienta (software) interactiva para la selección y diseño de la prótesis de cadera adecuada a un determinado paciente, con respecto a materiales biocompatibles de uso común y las dimensiones del dispositivo.

1.4 Metas

Compilación detallada de los materiales que se utilizan para la fabricación de las prótesis *Omnifit HA* y *Omnifit C* y caracterización de información biológica necesarias para su uso. Con esta información se diseñará y generará una base de datos.

- Compilación detallada de la correlación entre materiales de fabricación de las prótesis de cadera con las características biológicas del paciente para las prótesis *Omnifit HA* y *Omnifit C*.
- Generación de los modelos tridimensionales interactivo de las prótesis existentes en el mercado.
- Elaboración de un portal de Internet donde se pongan los resultados del estudio de este trabajo así como la herramienta desarrollada.

Capítulo II

2.1 Metodología

Para la realización de este trabajo se utilizan diferentes plataformas de desarrollo y métodos de estudio para el diseño y construcción de los diferentes módulos:

2.2 BioM-BD: Base de datos biomateriales

Para la recolección de información se utilizaron dos métodos: la entrevista personalizada con doctores especializados en el área en México en los implantes de prótesis de cadera, experiencias y preferencias sobre las mismas, generación de fichas bibliografías que contengan la información de biomateriales para la construcción de las prótesis más usadas en México. Después de generada esta información, se utilizaron para la fabricación de la base de datos que permitirá almacenar toda la información generada en la plataforma *Linux (Sistema operativo de licencia libre)* con *MYSQL (MySQL AB)* como creador/controlador de base de datos y Servidor de Páginas de Internet *Apache (Apache Software Foundation)* combinado con *PHP (Php Group)* lenguaje de programación que permitirá el manejo de la información vía Internet.

2.3 BioCH-BD: Base de datos de biocompatibilidad del Humano

Generación de una base de datos que permita correlacionar los materiales biocompatibles de las prótesis con las características biológicas del o los pacientes.

2.4 RP-M3D: Representación de las prótesis en su modelo tridimensional

Para la creación de modelos tridimensionales se utilizó el paquete *3DStudio Max 4.0 (AutoDesk inc)*. exportados a Macromedia Director Mx (Macromedia Inc), para su manipulación.

2.5 SIM-PTC: Sistema de Integración de Modelado de Prótesis de total cadera

Generación y visualización de la prótesis de cadera de acuerdo a los resultados de los módulos a, b y c. Utilizando el lenguaje Lingo de Macromedia Director Mx (Macromedia Inc) para la modelación y simulación así como el lenguaje PHP (*Php Group*) para todos los desarrollos de algoritmos necesarios.

2.6 I-WEB: Interfase de portal de Internet

La interfase del portal de Internet, es la herramienta que unifico y permite visualizar cada uno de los procesos del sistema final que se debe lograr a través de este estudio. Esta herramienta permite manipular los componentes tanto de biomateriales como la representación tridimensional de las prótesis. Para lograr estos resultados se utilizará como plataforma de desarrollo el lenguaje de programación *Lingo* que pertenece a *Macromedia Director Mx (Macromedia Inc)*.

El siguiente diagrama a bloques (Diagrama 1) muestra el orden de desarrollo de los diferentes módulos ya planteados.

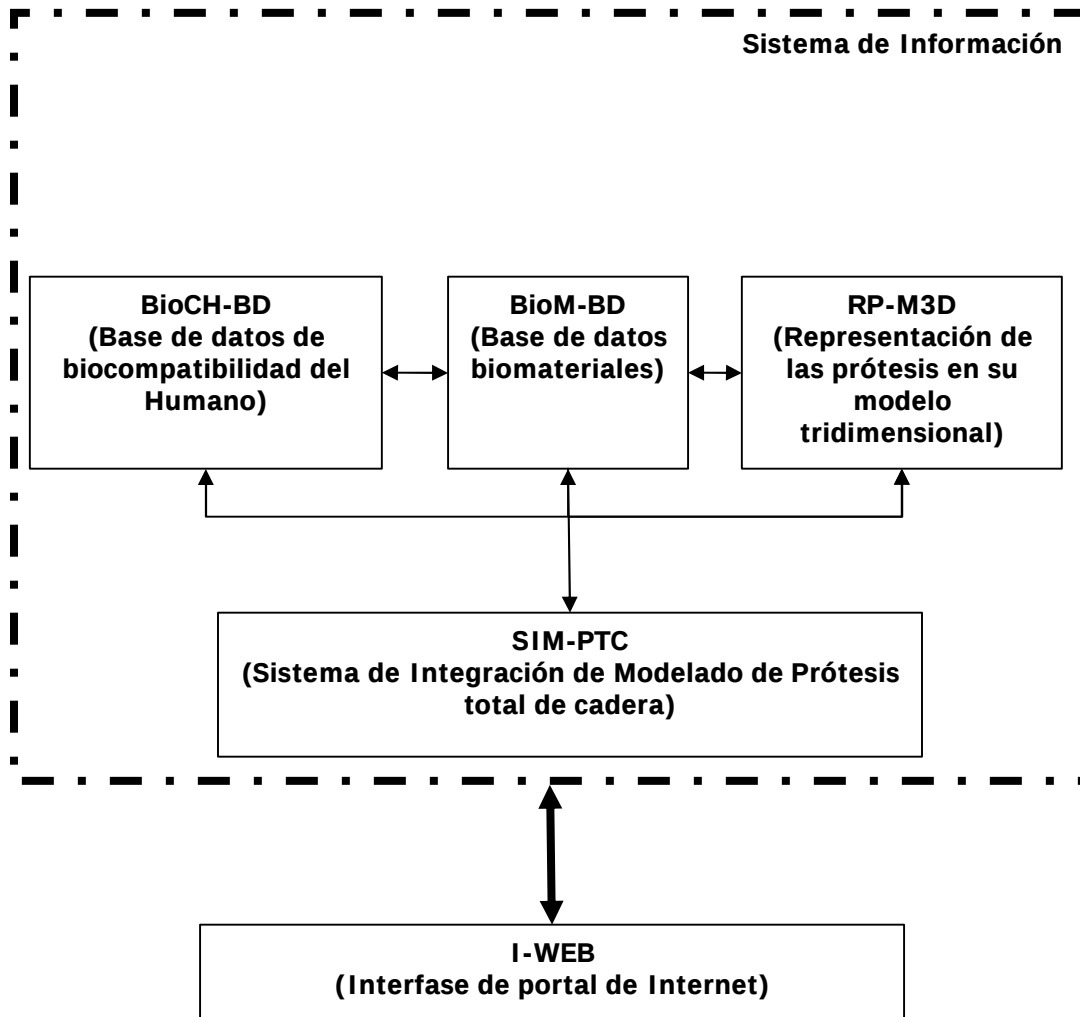


Diagrama 1. Representación de los módulos que componen la herramienta I-WEB

Capítulo III

3.1 Desarrollo experimental

El Lenguaje de Modelamiento Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

Módulos: Ver figura Diagrama 1 para referencia de los módulos que continuación se describen.

3.2 BioM-BD: Base de datos biomateriales

Para llevar un control de los biomateriales se diseñó una base de datos constituida por las siguientes características:

- Físicas : Fuerza-estrés y flexibilidad
- Físico-Químicas: Resistencia a la corrosión
- Biocompatibilidad
- Biomateriales

Las siguientes clases definen el comportamiento del módulo, diagrama 2:

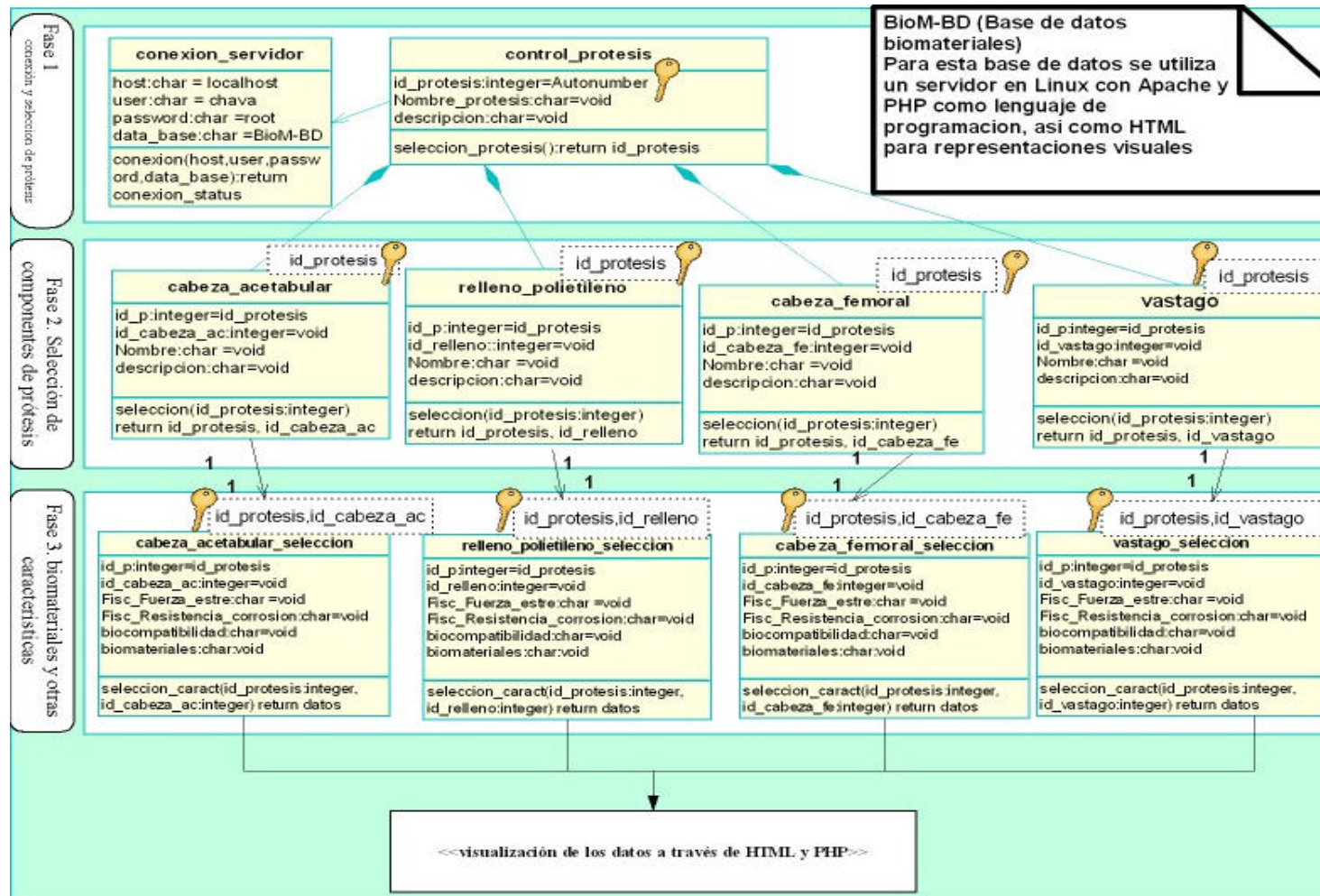


Diagrama 2: Representación de las clases utilizadas en módulo BioM-BD

Descripción de las clases mostradas en diagrama 1 en UML:

El diagrama se compone de tres fases, la primera fase *conexión y selección de prótesis*, el propósito de esta fase es únicamente generar la comunicación del módulo con el servidor a través de sus diversos atributos y sus métodos definidos, la instancia de estas clases definidas tienen una asociación de 1:1 y la clase *control_protesis* tiene una *agregación por valor* con todas las clases de la fase *Selección de componentes de prótesis*. En la fase dos ya seleccionado el modelo deseado se desglosa en los diferentes componentes de la PTC (Prótesis total de cadera), como son la cabeza acetabular, el relleno de polietileno, cabeza femoral y el vástago correspondientes al modelo seleccionado, cada uno con sus respectivos atributos.

La fase 3 contiene el grueso de este módulo ya que permite conocer las características principales de la PTC a través de sus atributos y métodos diseñados. Cada una de las clases en esta fase están parametrizadas por dos variables que le permitirán seleccionar a cada clase el componente de la prótesis que se haya seleccionado en la fase anterior, permitiendo mostrar la información al usuario a través de diferentes componentes visuales programados con lenguajes como HTML, PHP y Lingo.

Descripción de Fase 1:

Conexión y selección de prótesis:

- conexión_servidor: Genera la conexión a la base de datos para el acceso a los datos. Esta clase utiliza los siguientes atributos :
 - o Host: Dirección del servidor que almacena la base de datos (MYSQL)
 - o User: Usuario valido para la conexión con las bases de datos
 - o Password: contraseña valida para establecer una conexión validad al servidor de la base de datos.
 - o Data_base: Nombre de la base de datos en la cual se van a extraer la información necesaria para el funcionamiento de este módulo.

Métodos

- o conexión(host:char,user:char,password:char,data_base):return conexion_status: Este método realiza la conexión al servidor de base de datos utilizando los parámetros de entrada, generando un direccionamiento a la base de datos en caso de ser satisfactoria la conexión, caso contrario regresa un estado de error de conexión.
- Control_prótesis: Muestra los tipos de prótesis capturados en el sistema, permitiendo su selección, los atributos con los cuales hace la selección son los siguientes:

- o id_protesis: Este valor entero auto numérico único es el índice asignado por la base de datos para cada modelo registrado.
- o Nombre_protesis: Contiene el nombre comercial de la prótesis
- o Descripción: Contiene una breve descripción de la prótesis, este atributo permite al usuario final tener una idea mas firme sobre el tipo de prótesis.

Método

Selección_protesis: este método regresa el índice correspondiente a el modelo seleccionado.

Descripción de Fase 2

Selección de componentes de prótesis:

En este caso se utilizan clases parametrizadas ya que mantienen una relación de agregación por valor para cada una de las clases, las cuales tienen los mismos atributos; el único cambio significativo son los parámetros de entrada para permitir la *selección* de todas las partes involucradas en una PTC, entregando como resultado los índices correspondientes a cada componente que le permita a la siguiente fase seleccionar la información correspondiente a los biomateriales y los restantes atributos señalados en el diagrama.

Descripción de Fase 3

Biomateriales y otras características

En esta fase las clases siguen siendo parametrizadas ya que mantienen una relación de 1:1 con la fase 2, esto es necesario ya que los atributos de entrada para estas clases son los que le permitirán al método de *selección_caract* encontrar los datos necesarios correspondientes a cuestiones físicas, biocompatibilidad y biomateriales de la PTC.

Las clases desarrolladas en esta fase requieren como atributos los siguientes datos:

- Clase *cabeza_acetabular_seleccion*:

Id_cabeza_ac: permite la identificación de la cabeza acetabular que le corresponde a la prótesis seleccionada a través de los atributos parametrizados *id_protesis* y *id_cabeza_ac* generados por la clase *cabeza_acetabular* a través de su método *selección*.

Fisc_fuerza_estre: contiene las características físicas del componente, indicando su relación fuerza – estrés de la pieza.

Fisc_resistencia_corrosion: contiene la información de las características correspondientes a la resistencia de los materiales de fabricación sobre la corrosión.

Biocompatibilidad: contiene la información de biocompatibilidad de los materiales de fabricación con el ser humano así como sus posibles contra indicaciones.

Biomateriales: contiene la información correspondiente a los tipos de materiales usados en la fabricación de la prótesis seleccionada.

- Clase *relleno_polietileno_seleccion*:

Id_relleno: permite la identificación del relleno de polietileno que le corresponde a la prótesis seleccionada a través de los atributos parametrizados *id_protesis* y *id_relleno* generados por la clase *relleno_polietileno* a través de su método *selección*.

Los componentes de *Fisc_fuerza_estres*, *Fisc_resistencia_corrosion*, *Biocompatibilidad*, *Biomateriales*: contienen la información correspondiente al componente.

- Clase *cabeza_femoral_seleccion*:

Id_cabeza_fe: permite la identificación de la cabeza femoral que le corresponde a la prótesis seleccionada a través de los atributos parametrizados *id_protesis* y *id_cabez_fe* generados por la clase *cabeza_femoral* a través de su método *selección*.

Los componentes de Fisc_fuerza_estres, Fisc_resistencia_corrosion, Biocompatibilidad, Biomateriales: contienen la información correspondiente al componente.

- Clase *vastago_seleccion*:

Id_vastago: permite la identificación del vástago que le corresponde a la prótesis seleccionada a través de los atributos parametrizados *id_prtesis* y *id_vastago* generados por la clase *vastago* a través de su método *selección*.

Los componentes de Fisc_fuerza_estres, Fisc_resistencia_corrosion, Biocompatibilidad, Biomateriales: contienen la información correspondiente al componente.

Todas las clases contienen un método *seleccion_caract* que mantiene una relación con la clase <<visualización de los datos a través de HTML y PHP >> permitiendo interactuar con los módulos:

SIM-PCT (Sistema de Integración de Modelado de Prótesis de cadera total) y *I-WEB (Interfase de portal de Internet)*. Permitiendo el manejo de características tridimensionales asignados a los diferentes componentes de las prótesis.

3.3 BioCH-BD: Base de datos de biocompatibilidad del Humano

La siguiente base de datos controla las características necesarias que un paciente debe de cumplir para ser candidato a una prótesis total de cadera. Los datos mostrados corresponden a las características señaladas por la compañía *Striker* para los dos tipos de prótesis expuestas en el presente trabajo, la tabla 3-1 muestra las características tomadas en cuenta para este estudio:

Datos del Paciente			
Edad	Sexo	Ocupación	Tipo de problema detectado
<50 50-60	Femenino, Masculino	Deportista, Ama de casa, trabajo de escritorio, trabajo de pie, Carga / movimiento de Equipo u mobiliario	Enfermedad degenerativa no inflamatoria de las articulaciones incluyendo artrosis y necrosis avascular, Artritis reumatoide
61-70 71-80			Deformidad funcional Desunión, fractura del cuello del fémur y fracturas trocanterianas del fémur proximal que involucran la cabeza
>81			La necesidad de una artroplastia total de cadera de refuerzo
			infección, septicemia y osteomielitis
			Coxoartrosis derecha, Luxación congénita, Espondilitis
			Enfermedad de Perthes, Fractura de cadera, Necrosis, Secuela de polio, Traumatismo

Tabla 3-1. Características del paciente para la selección de prótesis de cadera total.

Los siguientes puntos representan la estructura de las clases involucradas para el control de este módulo.

Clases:

- Diccionario_datos: Crea el diccionario de datos de la base de datos, asignándole los tipos de datos correspondientes.
- Representación_visual: Utilizando el lenguaje de Marcación de Hipertexto HTML (Hypertext Markup Language) por sus siglas en ingles combinado con PHP (Programing Hypertext Preprocessor) se crea la representación visual de los elementos que permiten trabajar con los diferentes tipos de datos.
- conexion_servidor: Genera la conexión a la base de datos para el acceso a los datos.
- conexion_elementos_visual_con_diccionario: Código en PHP que genera conexión para el vaciado de la información que contienen las bases de datos.

Las clases *conexion_servidor* mantienen una relación constante con las clases ya que su método permite el acceso a los datos del servidor de base de datos, la clase *caracteristicas_protesis* mantiene una agregación por referencia con las clases *Representación_visual* y *conexion_elementos_visual* esta relación es estrictamente necesaria ya que los datos que necesitan estas clases son proporcionados por la clase de la cual tienen dependencia.

La finalidad de este módulo es indicar las posibles causas por las cuales un paciente es candidato a una prótesis total de cadera, correlacionando los datos de edad, sexo, ocupación y tipo de problema con los del módulo anterior.

El diagrama 3 en UML muestra la relación entre los diferentes clases para este módulo.

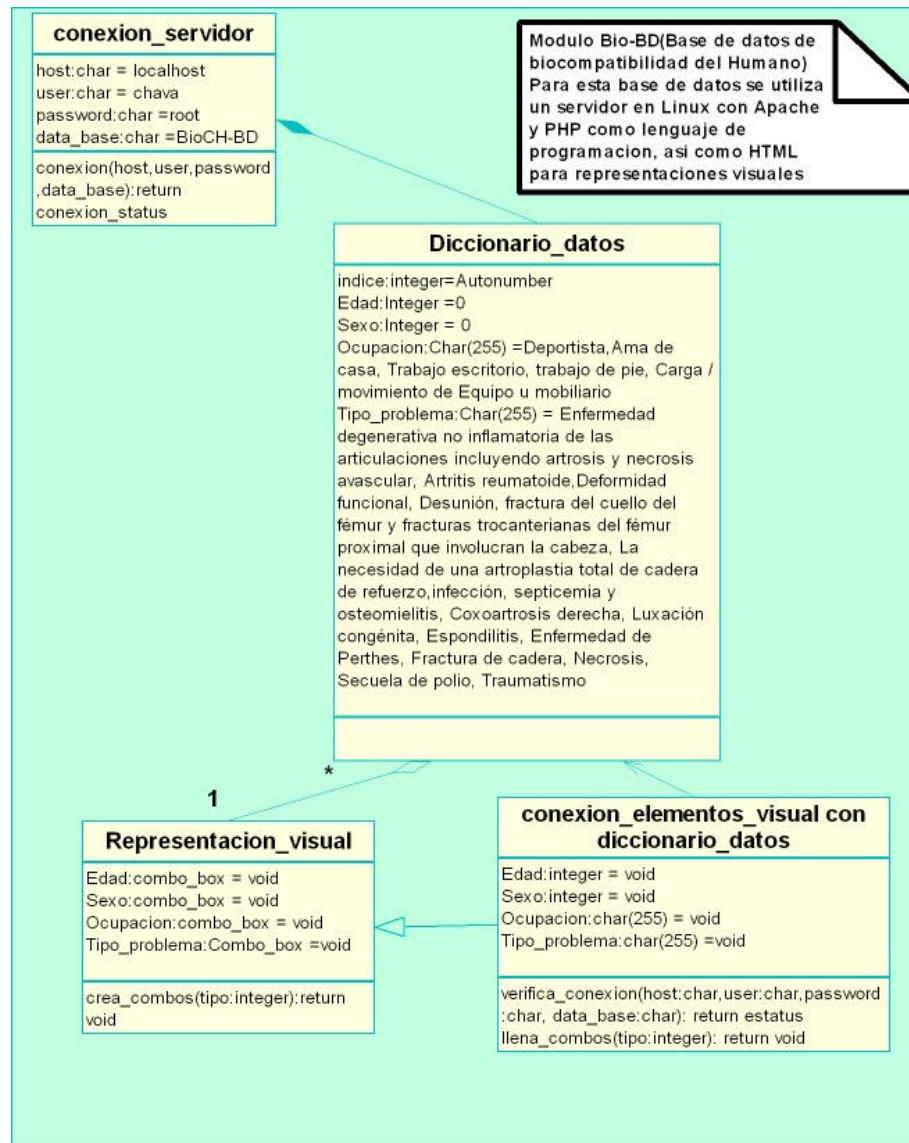


Diagrama 3 en UML. Clases y sus relaciones entre las clases.

3.4 RP-M3D: Representación de las prótesis en su modelo tridimensional

Para el desarrollo de este módulo se utilizó la herramienta CAD 3D Studio Max 7.0, partiendo de dos piezas completas de prótesis total, ambas piezas son de la compañía Estadounidense Stryker[®] las cuales son:

Cadera Omnifit HA: Tallo omnifit fabricado con aleación de titanio recubierto en su parte proximal con Hidroxiapatita, ver figura 3-1.

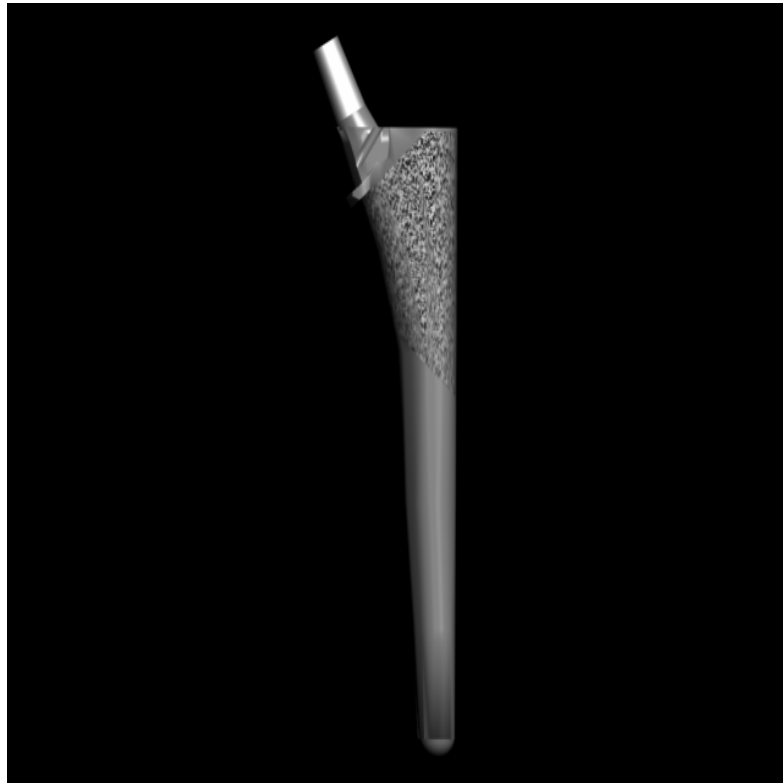


Figura 3-1. Cadera Omnifit HA

Características

- Tallo (vástago)
- Tallo con recubrimiento de hidroxiapatita, con estrías antirotacionales
- Con apoyo en calcar
- Cubierta de HA en 1/3 de su totalidad
- Cabeza en 22 y 28 mm
- Longitud de 100 a 170 mm
- Diámetro de 8 a 17 mm
- Base largo de cuello de 25 a 40 mm

Lo complementa la Cabeza acetabular “**cotilo osteonics**” que es un cotilo recubierto con plasma spray de titanio como lo muestra la figura c-2



Figura 3-2. Cotilo osteonics

Características

- No cementado en titanio, con 5 orificios laterales y tornillos de 6.5 mm
- Diámetros: desde el 40 al 72 de 2 en 2.
- Linner: En 22 y 28 mm y 10° de ceja posterior

Cadera Omnifit-C: Tallo de cadera cementado fabricado con aleación de titanio recubierto en su parte proximal superior con Hidroxiapatita con estrías antirotacionales ver figura 3-3.

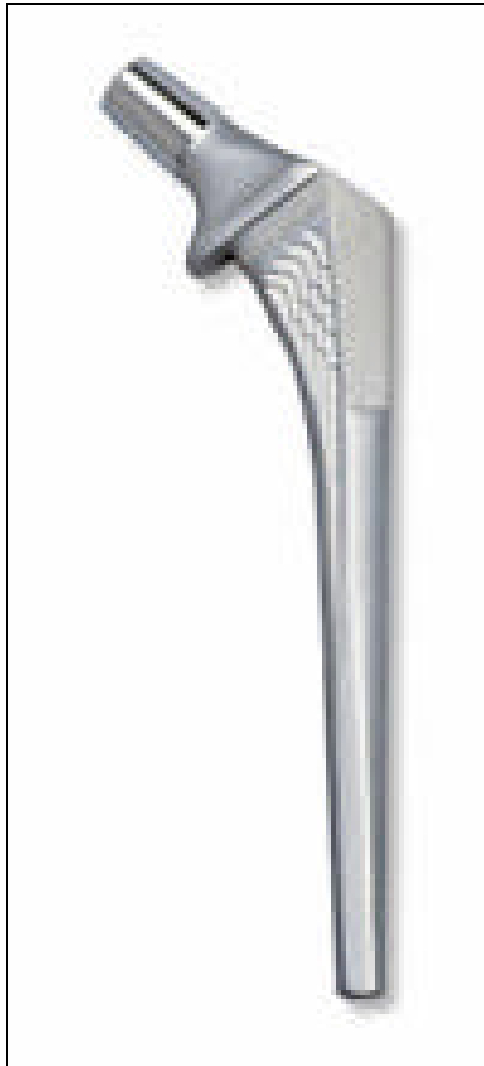


Figura 3-3. Cadera Omnifit-C

Características

- Tallo (vástago)
- Tallo con aleación de titanio con recubrimiento de hidroxiapatita
- Apoyo calcar
- Estrías antirotacionales
- Centralizador distal
- Cabeza en 22 y 28 mm
- Longitud: de 100 a 170 mm
- Diámetros: de 8 a 16 mm
- Base largo de cuello: de 25 a 40 mm

Se complementa con *Cotilo osteonics* que a su vez se complementa con *Liner*

Constreñido ver figura 3-2 y figura 3-4.



Figura 3-4. Liner constreñido

Características

- Cotilo constreñido para utilización en Cotilo Osteonics no cementado
- Pared posterior de 20°
- Diámetro: Interno 22mm: 50, 52, 54, 56, 61, 66, 70 mm
- Interno 28 mm: 58, 61, 66, 70, 74 mm
- Grado de movilidad desde 72° a 82°

Diseño de prótesis con herramienta CAD

El procedimiento a seguir para el diseño se apoyó en los siguientes pasos:

Se tomaron medidas de las piezas que componen la PTC correspondiente al modelo *Cadera Omnifit C* fabricado por la compañía *Striker*: Cabeza Acetabular, Relleno de Polietileno, Cabeza Femoral y vástago. La figura c-5 muestra los diferentes componentes generados con la herramienta CAD.

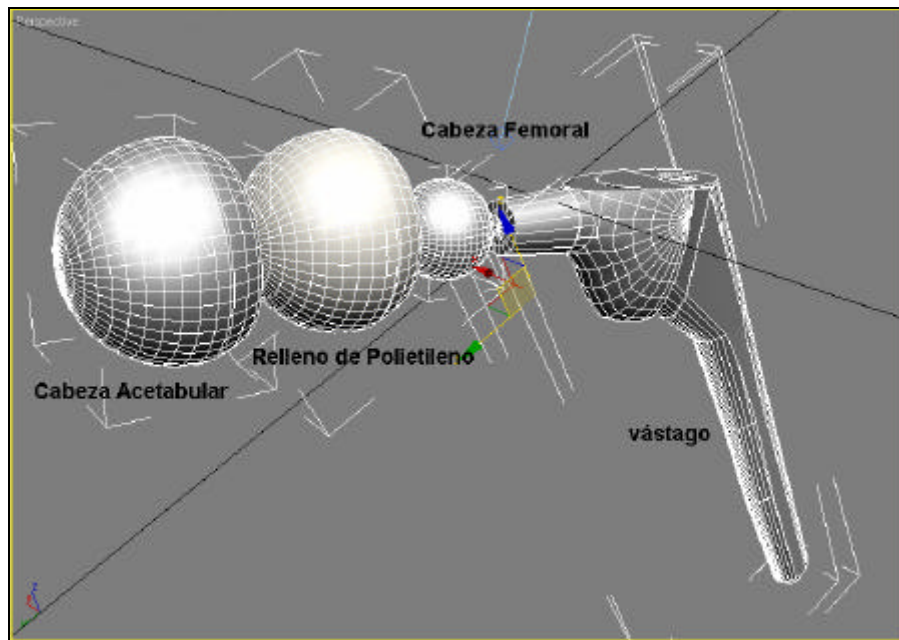


Figura 3-5. Componentes generados con la herramienta CAD, correspondiente al modelo *Cadera Omnifit C*.

La figura 3-6 muestra cómo se tomaron las medidas de los componentes de la PTC, correspondiente al modelo *Cadera Omnifit C*

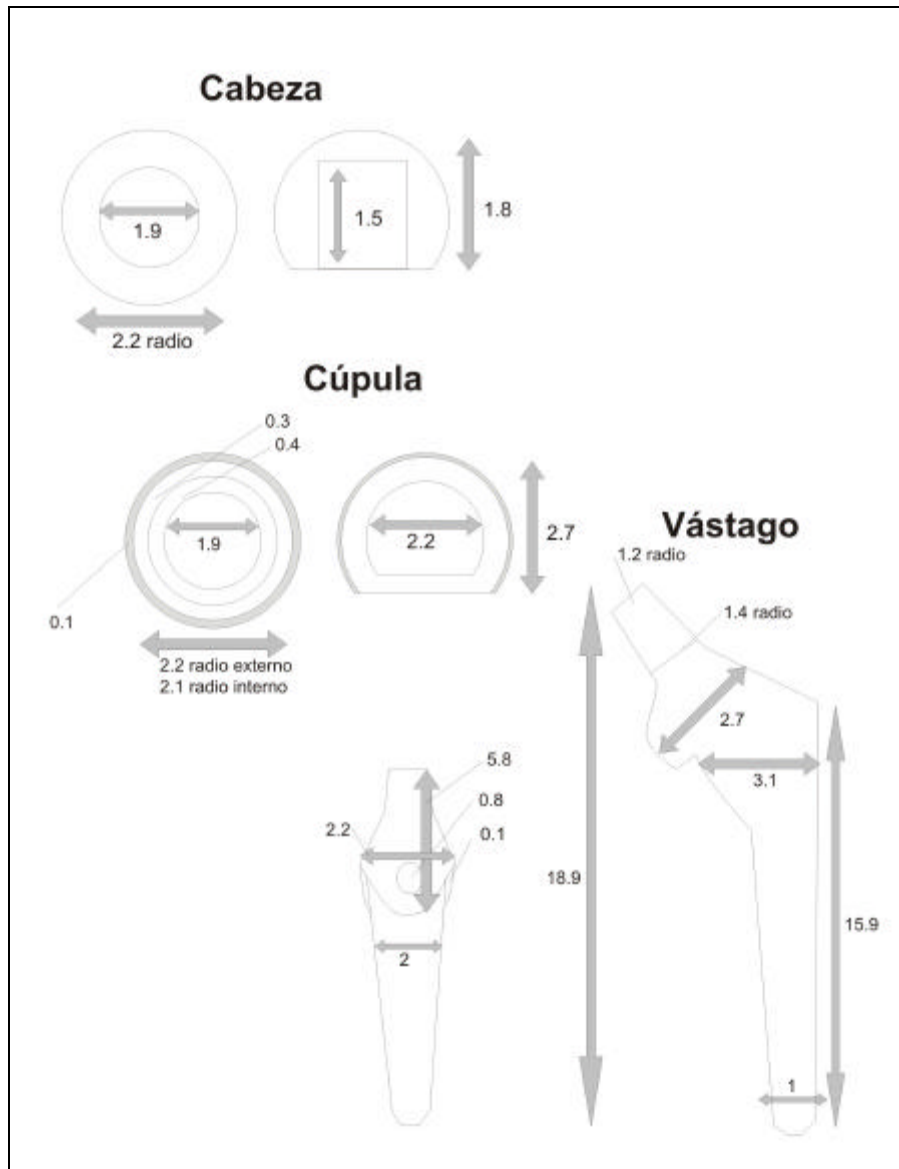


Figura 3-6. Representación en centímetros de las piezas que componen la PTC,
modelo *Cadera Omnifit C*

Para la segunda prótesis se tomaron medidas de las piezas que componen la PTC correspondiente al modelo *Cadera Omnifit HA* fabricado por la compañía *Striker*. Cabeza acetabular con Relleno de Polietileno, Cabeza Femoral y vástago. La figura 3-7 muestra los diferentes componentes generados con la herramienta CAD.

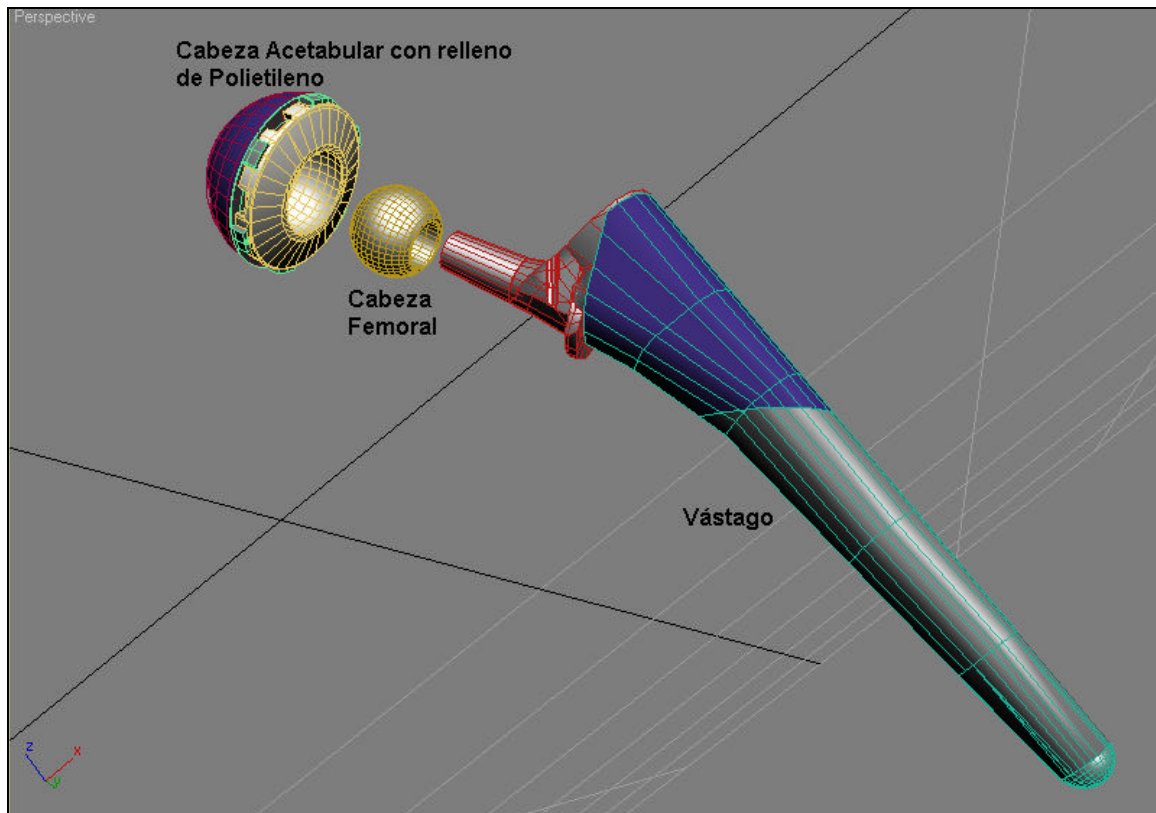


Figura 3-7. Componentes generados con la herramienta CAD, correspondiente al modelo *Cadera Omnifit C*.

La figura 3-8 muestra como se tomaron las medidas de los componentes de la PTC, correspondiente al modelo *Cadera Omnifit HA*

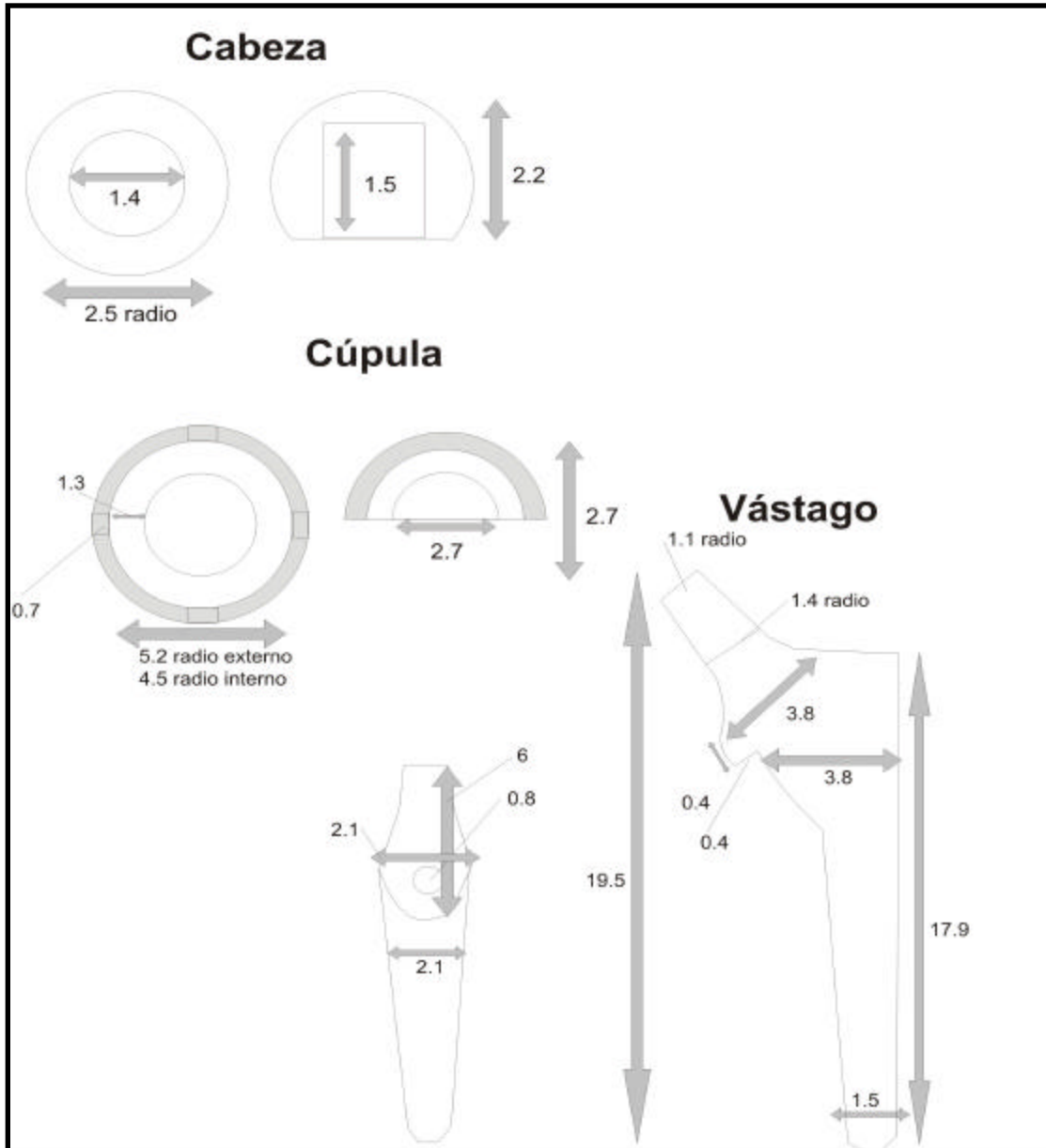


Figura 3-8. Representación en centímetros de las piezas que componen la PTC, modelo *Cadera Omnifit HA*.

Los dos modelos digitalizados le permiten a los siguientes módulos tener una representación visual tridimensional de los componentes al igual que poder tener la facultad de modificar, rotar, separar y manipular casi en su totalidad todos los componentes de las prótesis.

3.5 SIM-PTC: Sistema de Integración de Modelado de Prótesis Total de cadera

Se encarga de la generación y visualización de la prótesis de cadera de acuerdo a los resultados en el módulo 3.4. En este módulo se utiliza el lenguaje Lingo de Macromedia Director Mx (Macromedia Inc) para la modelación y simulación.

Para lograr interactuar con los modelos tridimensionales de las prótesis se diseñaron diferentes clases que permiten a través del lenguaje lingo interactuar con las piezas diseñadas con la herramienta CAD.

El siguiente diagrama 4 en UML muestra las clases y sus relaciones con los modelos tridimensionales.

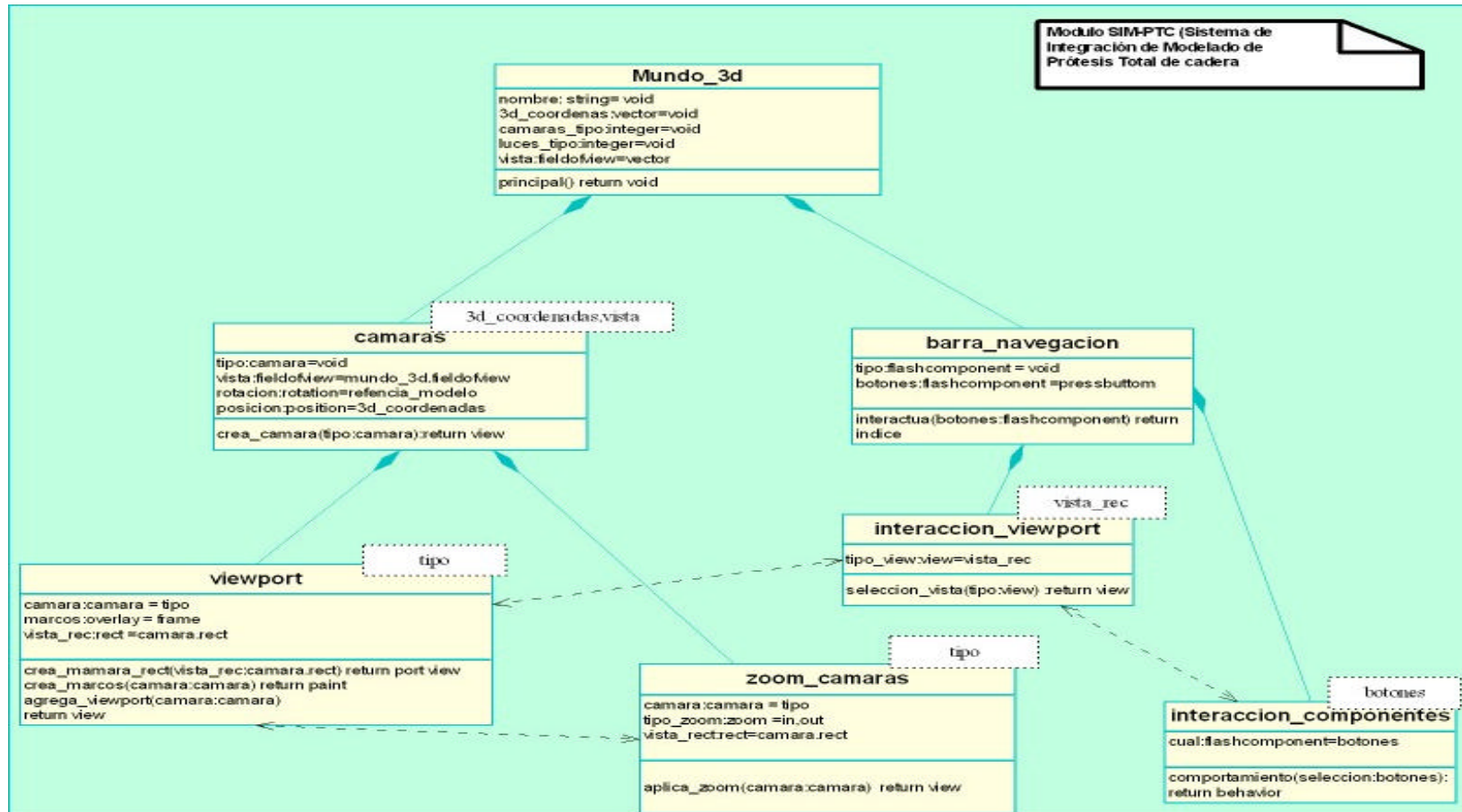


Diagrama 4 en UML muestra las clases y sus relaciones con los modelos tridimensionales.

Descripción de las clases:

Clase Mundo_3D:

Esta clase es la principal debido a que es la encargada de representar las prótesis en un ambiente tridimensional, asignándole automáticamente características espaciales como: vector de posición en el mundo virtual, ejes a cada objeto (componente de la prótesis) dotándolo de las propiedades de rotación, transformación y translación. La inicialización del mundo virtual se logra a través del módulo *principal()* que contiene los siguientes eventos para su funcionamiento:

```
on preparemovie
.
.
end preparemovie
.
.
on startmovie
end startmovie
```

El evento *preparemovie*: Asigna todas las propiedades de rotación, translación y transformación del objeto tridimensional en nuestro caso la prótesis, permitiendo a través de comandos del lenguaje Lingo interactuar con los modelos tridimensionales.

El evento *startmovie*: Inicializa las variables globales del sistema.

La figura 3-9 y 3-10 muestra la representación de las prótesis de cadera en el ambiente tridimensional.

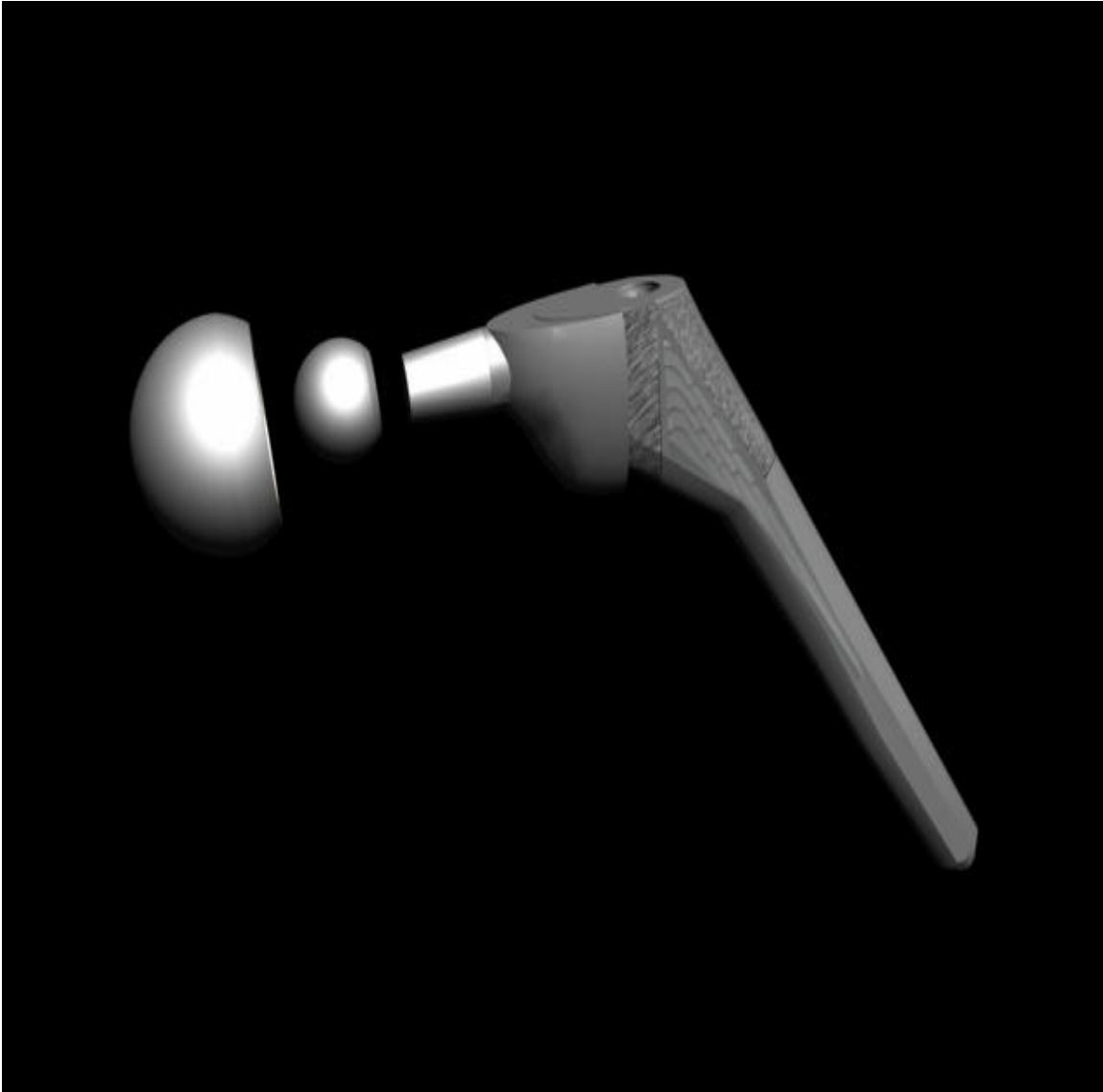


Figura 3-9. Representación tridimensional de la prótesis Cadera *OmniFit C*, dentro del ambiente del lenguaje de programación Lingo.

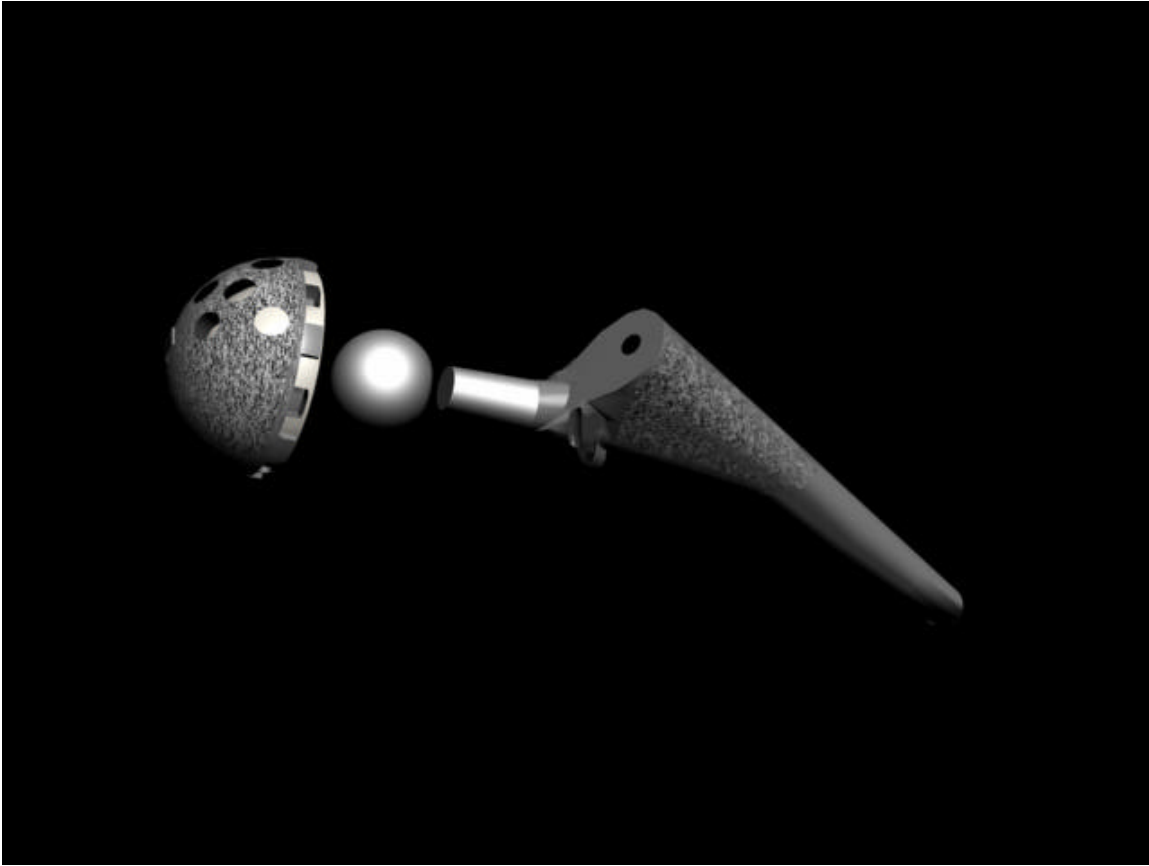


Figura 3-10. Representación tridimensional de la prótesis Cadera *Omnifit HA*, dentro del ambiente del lenguaje de programación Lingo.

El tener estas piezas dentro de nuestro ambiente de programación tridimensional nos permite crearle atributos más complejos como la creación de cámaras, luces y barras de navegación que interactúen con ellas.

Clase cámaras:

Esta clase se encarga de crear las cámaras necesarias para interactuar con el clase *viewport* a través de su método *crea_camara* la explicación de la existencia de esta clase se debe a la necesidad de tener diferentes vistas que nos permitan captar visualmente los componentes de un mundo tridimensional, Para esta aplicación fueron creadas tres tipos de cámaras:

- Cámara perspectiva
- Cámara arriba
- Cámara de frente

Todas las cámaras están creadas en referencia a la posición del objeto en un mundo tridimensional, en este caso los objetos son las prótesis de cadera.

El siguiente fragmento de código lingo muestra cómo se crea y asignan las cámaras ya mencionadas.

-- crea camaras

```
camara_top = escena.newCamera("camara_top")
camara_front = escena.newCamera("camara_front")
camara_perpectiva = escena.camera[1]
camara_perpectiva.translate(5,0,10)
--- genera marcos y relacion de las dos camaras de viewport
camara_top.transform = sprite("modelo3d").camera.transform
camara_front.transform = sprite("modelo3d").camera.transform
sprite("modelo3d").addcamera(camara_top,3)
sprite("modelo3d").addcamera(camara_front,4)
camara_top.fieldofview = sprite("modelo3d").camera.fieldofview * 1
camara_front.fieldofview = sprite("modelo3d").camera.fieldofview * 2
-- angulos de camaras
camara_top.rotate(0,50,-40,escena.model("bastago_final"))
camara_front.rotate(0,-40,0,escena.model("bastago_final"))
```

La siguiente figura 3-11 muestra las tres cámaras asignadas a un modelo de prótesis.

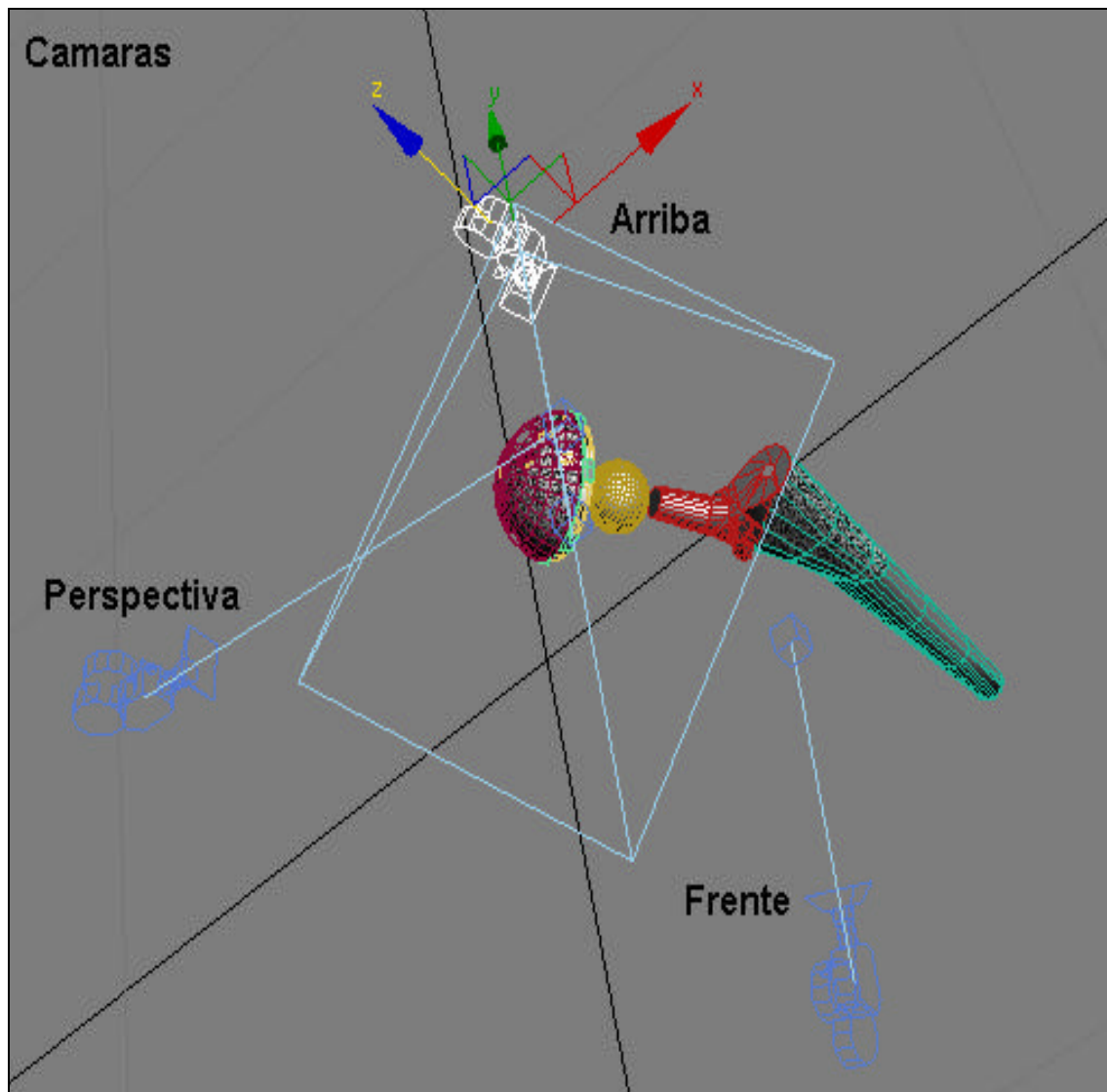


Figura 3-11 Posición de cámaras

Clase viewport:

Esta clase se encarga de interactuar con las cámaras creadas por el método *crear_camaras*, utilizando sus métodos para asignarle un espacio de visión a cada cámara.

La figura 3-12 muestra las tres cámaras asignadas al objeto tridimensional

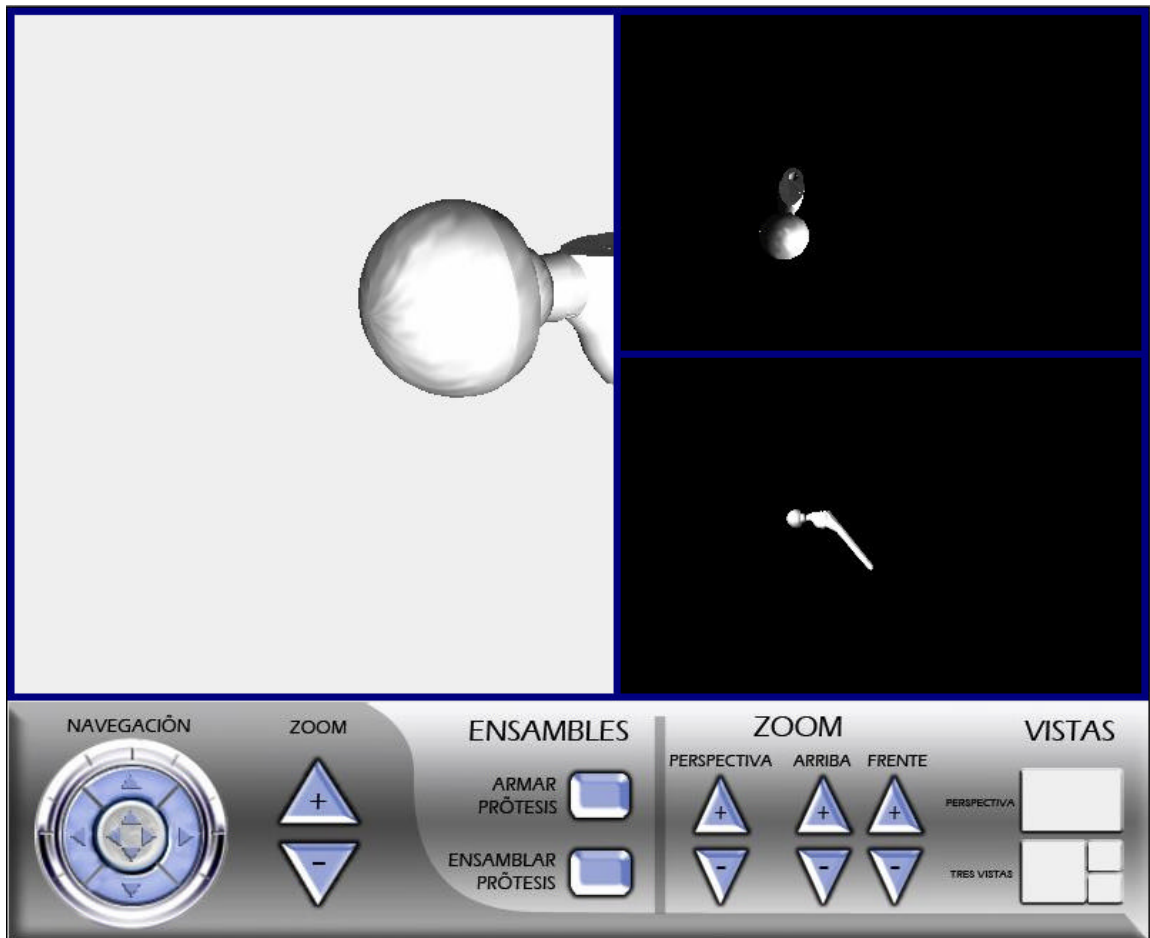


Figura 3-12 Vista de las tres cámaras

Los marcos y/o rectángulos que podemos observar en la figura d-4 son creados a través del método *crea_marcos* el cual delimita los espacios para cada toma de la cámara a través de rectángulos; utilizando los métodos *agrega_viewport* y

crea_mamara_rect . Al crear estos *viewports* se le heredan a las cámaras diversas propiedades que pueden ser manipulables a través del lenguaje Lingo en el mundo tridimensional.

Clase zoom_camaras:

Esta clase es la encargada de hacer acercamiento/alejamientos con los objetos en el mundo virtual, esta clase interactúa a través de las clases *viewport*, *interaccion_viewport* con la clase *barra_navegacion* para lograr su cometido de generar tomas de acercamiento y alejamiento. En el lenguaje Lingo para lograr este efecto es necesario relacionar los vectores de posición del mundo real con los vectores del modelo tridimensional para crear vectores espaciales que nos permitan tomarlos como referencia y así poder asignarlos a los vectores de posición de cada una de las cámaras.

Clase barra_navegacion:

Esta clase fue creada utilizando Flash de Macromedia director inc., el objetivo principal de esta barra es proporcionar al usuario una interface de interacción con los objetos tridimensionales en el mundo virtual.

La figura 3-13 muestra los controles visuales creados con Flash para la interacción.

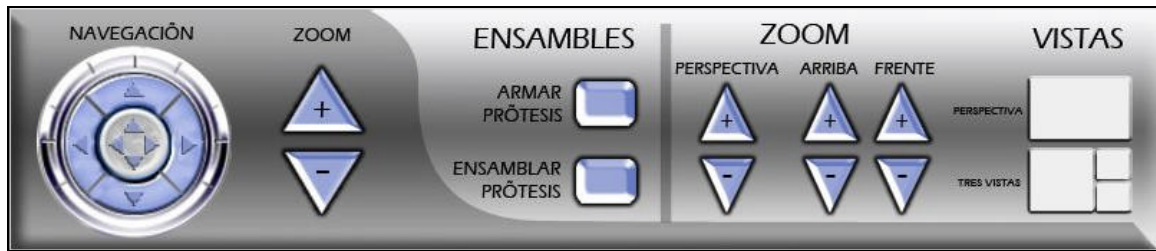


Figura 3-13 Barra de navegación

Para lograr esta comunicación lingo permite interactuar nativamente con el lenguaje *JavaScript* que utiliza Macromedia Flash, para lograrlo esta clase tiene que interactuar constantemente con las clases *interaccion_viewport* y *interaccion_componentes*.

Clase *interaccion_viewport*:

Esta clase se encarga de interactuar con la clase *barra_navegacion* a través de los tipos de vistas creados por la clase *viewport* , permite determinar qué botones se presionaron correspondientes a los tipos de vista a través de su método *selección_vista*.

Clase *interaccion_componentes*:

Esta clase determina qué función debe de aplicar ya sea la función de girar, hacer acercamiento y/o alejamientos, armar prótesis o ensamblar prótesis dependiendo de que botón se haya presionado, para lograr esto utiliza su método *comportamiento*.

3.6 I-WEB: Interfase de portal de Internet

Este módulo se encarga de unir el resultado de todos los módulos permitiéndoles interactuar entre ellos. Para lograr la comunicación entre ellos se utilizó la herramienta Macromedia MX (*Macromedia inc*), que permite trabajar con todos los lenguajes de programación utilizados en el presente trabajo, estos lenguajes son tan diversos como Lingo, JavaScript, PHP y HTML. Todos ellos en su conjunto permiten formar la aplicación totalmente interactiva y funcional en Internet.

La figura 3-14 muestra el resultado final de todos los módulos a través de una pagina de Internet.

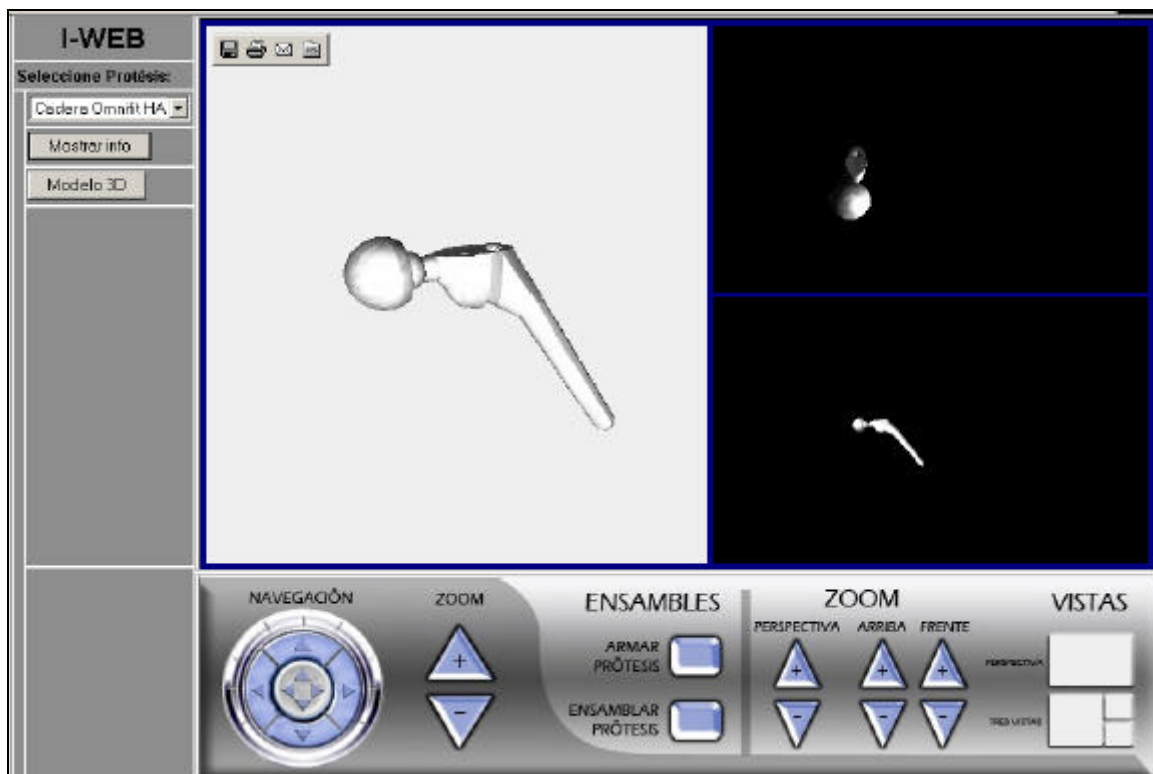


Figura 3-14 Unión de los módulos

Capítulo IV

4.1 Resultados

La herramienta interactiva a sido posible desarrollarla a través de diversas plataformas y componentes que en su conjunto permiten generar el ambiente tridimensional interactivo y los textos e imágenes visuales. Los módulos mencionados en el capítulo 3 son la mayoría del contenido de la herramienta, Pero existe un módulo en particular que es el que se encarga de unir todos los procesos a través de importantes métodos, el diagrama 5L muestra la relación que existe entre los módulos y sus métodos y parámetros que comparten entre ellos para la generación del módulo **I-WEB**.

El diagrama 5 en UML describe los diferentes procesos que conforman la herramienta desarrollada:

Conexión Principal:

Este proceso es el encargado de generar conexiones a las bases de datos que permiten consultar tanto los datos de biomateriales, biocompatibilidad y modelo tridimensional representado en formato Shockwave (Macromedia Inc) para los diferentes módulos.

Los módulos BioM-BD y BioCH-BD cumplen su función entregando sus resultados en lenguaje PHP a través del servidor de internet *Apache* al igual que el módulo SIM-PTC entregando los atributos tridimensionales de los componentes de la

prótesis, también entrega los atributos de la barra de control para la manipulación del ambiente tridimensional.

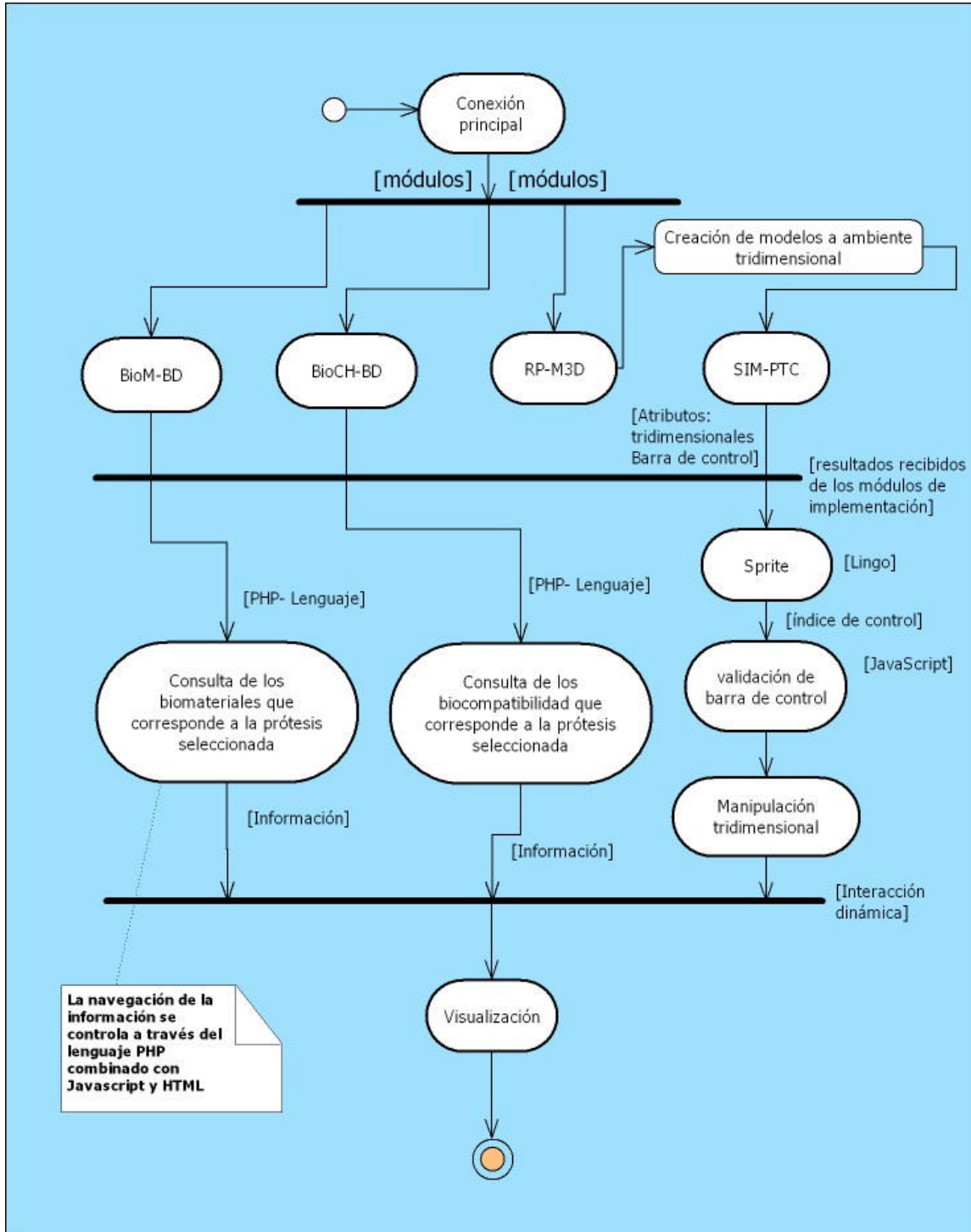


Diagrama 5 en UML. Procesos del módulo I-WEB

El módulo RP-M3D se encarga de modelar las partes de la prótesis con la herramienta 3DStudio Max respetando sus dimensiones reales, permitiendo la creación de estos modelos tridimensionales al ambiente tridimensional del lenguaje lingo a través del convertidor de formato 3DS (Formato 3DStudio Max, AutoDesk Inc) a formato W3D (ShockWave, Macromedia Inc) generando los atributos para su manipulación tales como ubicación en el espacio tridimensional a través de un vector de posición, atributos de rotación, translación y posición.

El proceso *Consulta de los biomateriales que corresponde a la prótesis seleccionada* interactúa con el módulo de BioM-BD a través del lenguaje de programación PHP utilizando las clases y métodos expuestos en el capítulo 3 entregando la información correspondiente al proceso de visualización.

El proceso *Consulta de los biocompatibilidad que corresponde a la prótesis seleccionada* interactúa con el módulo de BioCH-BD a través del lenguaje de programación PHP utilizando las clases y métodos expuestos en el capítulo 3 entregando la información correspondiente al proceso de visualización.

El proceso *Sprite* esta desarrollado en el lenguaje Lingo, permitiendo combinar dos tipos de formatos y generando su comunicación permanente entre ellos, para lograr esta comunicación es necesario combinar el lenguaje Lingo con JavaScript que es el lenguaje que trabaja el componente *barra de control*. Un

ejemplo del código que activa un solo botón de la barra de control (ver Figura 3-12m capítulo 3) para activar las diferentes vistas de cámaras del ambiente tridimensional:

```
function regresa_boton(botton){  
  If(_click_botton="vista_1"){ return tipo = "vista 1"  
    .  
    .  
    .  
  }  
}
```

El código anterior determina que botón fue presionado y qué valor debe de entregar al lenguaje lingo, generando un *índice de control*.

La *validación de barra de control* en JavaScript genera a través del índice de control entregado por el proceso Sprite activar las funciones de acercamiento y/o alejamiento (Zoom), Rotación y manejo de cámaras como podemos ver en el siguiente fragmento de código Lingo:

```
On exit frame  
-- entrega valor del botón que fue seleccionado  
Botton = sprite("barra_control").getVariable(botton)  
  Case "acercamiento": camara_perspectiva.fieldofview = 90  
  Case "alejamiento": camara_perspectiva.fieldofview = -90  
  end case  
End
```

El código anterior detecta qué botón de la barra de control se está activando y, de acuerdo al valor correspondiente; selecciona un comportamiento a ejecutar. En el caso del ejemplo anterior, se puede seleccionar entre dos tipos de casos: es el alejamiento y acercamiento; al seleccionar cualquiera de ellos asignará un

acercamiento o alejamiento modificando el campo de visión (“fieldofview”) asignándolo a la camara_perspectiva.

El proceso *Manipulación tridimensional* genera los siguientes métodos para la manipulación de los componentes de las prótesis:

- Rotación: el siguiente comando en Lingo permite rotar los componentes de la prótesis.

`Sprite(“mundo3d”).model(componente).rotation(vector)`

- o componente: Indica el nombre de la pieza a rotar, puede ser el vástago, cabeza acetabular etc..
- o vector: contiene el sentido de la rotación en los ejes x, y, z, por ejemplo `vector = vector(10,0,0)`, genera una rotación 10 grados en el eje “x” en el sentido del reloj, pero no hay rotación en los otros dos ejes.

- alejamiento y/o acercamiento: el siguiente comando en Lingo permite modificar el campo de visión de las cámaras aumentando o disminuyendo su ángulo de visión a través del siguiente comando.

`Sprite(“mundo3d”).camara(cámara).fieldofview = ángulo`

- o cámara: Indica el nombre de la cámara a modificar su ángulo de visión.

- o ángulo: este puede tomar un valor angular máximo de 160 y un mínimo de 30 grados, entre mayor sea el ángulo mayor será el acercamiento logrado y viceversa.
- translación (movimiento de cámaras): el siguiente comando permite mover la cámara principal que está fija a los modelos de las prótesis. Para moverla se utiliza el siguiente comando:
`Sprite("mundo3d").model(componente).translation=vector`
 - o componente: Indica el nombre de la pieza a rotar, puede ser el vástago, cabeza acetabular etc..
 - o vector: contiene el sentido de la rotación en los ejes x, y, z, por ejemplo `vector = vector(10,0,0)`, genera un moviendo 10 (centímetros o metros) de la unidad de medida que se haya seleccionado. En este caso son centímetros en el eje "x" el sentido del desplazamiento lo indica el signo de "+/-" donde el signo "+" indica desplazamiento hacia la derecha y el signo "-" en el sentido contrario, pero no hay desplazamiento en los otros dos ejes.

- posición de los componentes de las prótesis: El lenguaje Lingo permite ubicar un componente espacialmente en un mundo virtual a través de su vector de posición; éste es importante porque sin él estaríamos navegando a la deriva en un mundo virtual donde sería equivalente a navegar en el mar sin tener una brújula que nos indique en qué sentido viajamos. En este proceso se utilizan las coordenadas de posición tanto del mundo virtual que fueron asignadas al crear los modelos diseñados en 3DStudio Max a formato shockwave, así como las coordenadas de las cámaras y modelos de prótesis en el espacio tridimensional.

El siguiente comando calcula las coordenadas espaciales de posición del mundo virtual:

```
Posición_mundo = member("mundo3d").getWorldPosition
```

Asigna un vector con coordenadas x,y,z tomando como referencia el vector inicial (0,0,0).

El siguiente comando toma el vector de posición de las camaras tomando como referencia el vector inicial de la posición del mundo virtual:

```
Posición_camaras = member("mundo3d").camaras.getWorldPosition
```

Por ultimo se generan los vectores de los componentes de las prótesis a través del comando:

```
member("mundo").model(componente).getWorldPosition(#reference)
```

- o componente: Indica el nombre de la pieza a calcular sus vectores de posición espacial, puede ser el vástago, cabeza acetabular etc..
- o #reference: Esta referencia es importante ya que hay que indicarle con referencia a que va a determinar su posición, las características de rotación y translación dependen de estos vectores para hacer su función.

El proceso *Visualización* muestra de forma dinámica los resultados de todos los procesos mencionados a través de un espacio virtual bidimensional que permite interactuar con diversos lenguajes ya mencionados, el espacio virtual creado es de 800x400 píxeles para el mundo virtual y 800x150 píxeles para la barra de control. Ver figura 4-1 para conocer el resultado final de todos los procesos mencionados.

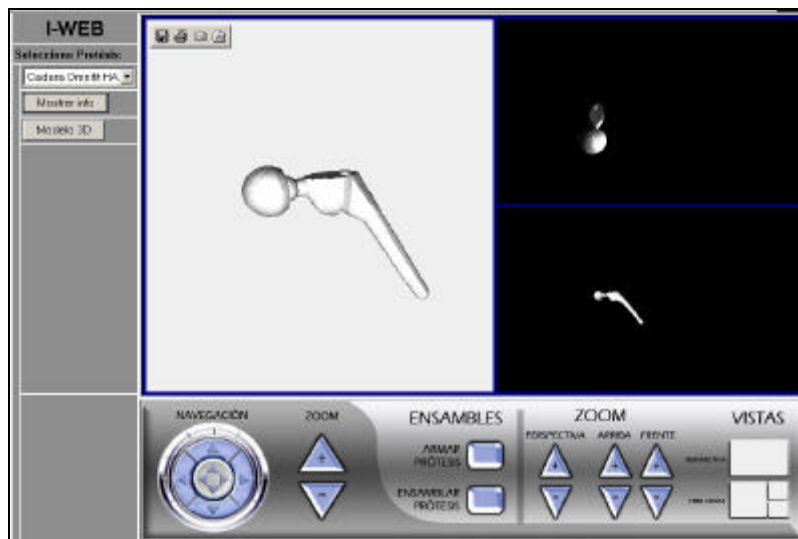


Figura 4-1. Resultado final de los procesos desarrollados en el módulo I-WEB

Conclusiones

El presente trabajo proporciona una herramienta básica interactiva que permite a los médicos conocer y seleccionar la prótesis de cadera total que un paciente requiera. Proporciona también un medio interactivo de entrenamiento que permita conocer las partes de una prótesis así como materiales y características particulares de las mismas.

Alcances de la herramienta desarrollada

- Proporciona una guía visual y a través de textos y modelos tridimensionales, no proporciona soluciones dimensionales, solo presenta la representación visual a escala de los dos modelos de prótesis representados.
- Al ser una herramienta que funciona a través de internet, el buen desempeño en la descarga de la información quedara sujeta a la velocidad de conexión con que el usuario cuente.
- Sólo contiene información de dos prótesis las cuales fueron seleccionadas para el presente trabajo. Esta herramienta tiene la capacidad de incrementar los modelos de prótesis.
- Los modelos tridimensionales fueron diseñados con la herramienta 3DStudio Max. La herramienta desarrollada es compatible con otras herramientas de desarrollo CAD como LightWare inc o Maya inc a través del formato ShockWave.

Recomendaciones

La herramienta desarrollada representa una plataforma básica que puede ser fortalecida a través de incorporación de las siguientes recomendaciones:

- Análisis de imágenes de radiografía que permita comparar dimensiones de la cadera sana y establecer una correlación con la cadera dañada.
- Diseño de sistema experto que permita hacer recomendaciones a los médicos para el tipo de prótesis que sea más conveniente para el paciente.
- Diseño de algoritmos de redes neuronales para el análisis de datos por población para el diseño de las prótesis.
- Análisis de elemento finito aplicado a las prótesis de cadera seleccionadas para determinado paciente, pudiendo con esto determinar la mejor opción en materiales de fabricación.

El tema de investigación queda abierto al abordaje en los siguientes temas:

- Manipulación de características de dimensiones métricas.
- Generación de código CNC para maquinaria de fabricación.
- Uso de Maquinaria de ingeniería inversa para la comprobación y correlación de las piezas fabricadas.

Referencias bibliográficas

- [1] Dr. Jaime Paulos Arenas, - Reemplazos Articulares -, Profesor Adjunto Departamento de Ortopedia y Traumatología, Pontificia Universidad Católica de Chile.1998;27:131134
<http://escuela.med.puc.cl/paginas/publicaciones/Boletin/html/PatolProstata/PatolProstata13.html>
- [2] M.H. Luque, - Diseño de una prótesis para la columna vertebral mediante el análisis estructural por elementos finitos - Facultad de Ingeniería Mecánica, Universidad Nacional de Ingeniería. Lima, Peru; verman@terra.com.pe
- [3] Vallet-Regí, M., González-Calbet, JM, (1997) Anales de la Química Internacional Edition. Suplemento 1 vol. 93.1,S6-S14
- [4] "Squirt Shape--Development of an Automated Fabrication Technique to Enhance Prosthetics CAD/CAM." Proceedings of the 2nd National VA Rehabilitation Research & Development Conference, Arlington, VA, Feb 20 - 22, 2000.
- [5] "Design and Validation of a Machine for Reproducible Precision Insertion of Femoral Hip Prostheses for Preclinical Testing", Journal of Biomechanical Engineering, Vol. 122 / 203, April 2000.
- [6] "PolywareTM, software for Accurade 3D Measure, tool for Orthopaedic Analysis, <http://www.draftware.com/pw/polyware.htm>
- [7] Ing. Maria Elvira Zemman, - Proyecto Arom - Instituto Nacional de Bioingeniería de la UCV, Universidad Central, Caracas Venezuela.
<http://www.ucv.ve/cebio/aromnuevo.htm>
- [8] M.L. Martínez Muneta, G. Romero Rey y J. Félez Mindán,- proyecto WEBD -Universidad Politécnica de Madrid, ETS Ingenieros Industriales. Departamento de Ingeniería Mecánica y Fabricación, E-mail: muneta;gromero;jfelez@etsii.upm.es
- [9] Vallet-Regí, M., (2000) "Biomateriales para sustitución y reparación de Tejidos", Departamento de Química Inorgánica y Bioinorgánica, Facultad de Farmacia. Universidad Complutense 28040-MADRID
- [10] Domínguez-Hernández VM, Carbajal RMF, Rico MG, Urriolagoitia CG
Análisis de una prótesis no convencional bloqueada para cadera mediante el método del elemento finito Rev Mex Ing Biomed 2003; 24 (1): 37-44

- [11] Baleani, M., Viceconti, M., Muccini, R., Ansaloni, M., "Endurance verification of custom-made prostheses," *International Journal of Fatigue* 22: 865-871, 2003
- [12] Chang, P.B., Williams, B.J., et al., "Design and analysis of robust total joint replacements: finite element model experiments with environmental variables," *Journal of Biomechanical Engineering* 123: 239-246, 2001
- [13] Fernandes, P., Rodrigues, H., Jacobs, C., and Pellegrini, V., "A material optimization model for bone remodeling around cementless hip stems," *European Conference on Computational Mechanics, ECCM'99, Munich, 1999.*
- [14] Joshi, M.G., Advani, S.G., Miller, F., and Santare, M., "Analysis of a femoral hip prosthesis designed to reduce stress shielding," *Journal of Biomechanics* 33: 1655-1662, 2000.
- [15] Pivonka, R., "Alteration of femoral stem stiffness and the effect on proximal stress shielding in the human femur." A thesis. University of Louisville. Louisville, KY. December, 2003.
- [16] Srinivasan, S., de Andrade, J.R., Biggers Jr., S.B., Latour Jr., R.A., "Structural response and relative strength of a laminated composite hip prosthesis: effects of functional activity," *Biomaterials* 21: 1929-1940, 2000.
- [17] Wolff, J., "Veber die innere Architectur der Knochen und ihre bedeutung fur die frage vom knochenwachstum," *Virchow's Arch. Path. Anat.* 50: 389-453, 1870.
- [18] Partridge Allen , "Real – Time Interactive 3D Games, creating 3D games in Macromedia Director 8.5", Sams Editorial, 2002, ISBN 0672322854
- [19] Catanese Paul, "Director's Third Dimension, Fundamental of 3D programming in Director 8.5", Que Editorial, 2002, ISBN 0672322285
- [20] Rozenzweig Gary, "Using Macromedia Director MX", Que Editorial, 2005, ISBN 045822254554

Anexo

Código fuente desarrollado en Lingo

Generado en la versión Macromedia Director MX 9.0.

```
Global nombre,escena,camara_front, camara_perpectiva,
camara_top,v_barra_p,h_barra,v_barra
Global nombre_obj,coordenadas_ini,i
on preparemovie
    set nombre ="protesis1"
    set v_barra_p="barra325x5"
    set h_barra="barra5x600"
    set v_barra="barra800x5"
    i=0
    resetworld(member(nombre))
end preparemovie

on startmovie
    set escena = member(nombre)
    camara_perpectiva = escena.camera[1]
    member("texto").text="Posicione el cursor del ratón sobre la pieza, para ver
sus características principales ..."
    agrega_marco_inicial
end

on crea_camaras
    -- mueve la camara de perspectiva
    if(camara_perpectiva=void) then
        camara_perpectiva = escena.camera[1]
    end if

    if(camara_top=void) then
        camara_top = escena.newCamera("camara_top")
    end if
    if(camara_front=void) then
        camara_front = escena.newCamera("camara_front")
    end if
end
```

```

on agrega_marco_inicial
  --- Asigna marcos para camara 1 el principal
  -- marco vertical
  crea_camaras
  if (escena.texture(v_barra)=void) then
    escena.newTexture(v_barra,#fromcastmember,member(v_barra))
  end if
  camara_perpectiva.addOverlay(escena.texture(v_barra),point(0,0),0)
  if (escena.texture("v_barra_abajo")=void) then
    escena.newTexture("v_barra_abajo",#fromcastmember,member(v_barra))
  end if
  camara_perpectiva.addOverlay(escena.texture(v_barra),point(0,480),0)
  -- marco horizontal
  if (escena.texture(h_barra)=void) then
    escena.newTexture(h_barra,#fromcastmember,member(h_barra))
  end if
  camara_perpectiva.addOverlay(escena.texture(h_barra),point(0,0),0)
  if (escena.texture("h_barra_derecha")=void) then
    escena.newTexture("h_barra_derecha",#fromcastmember,member(h_barra))
  end if
  camara_perpectiva.addOverlay(escena.texture(h_barra),point(795,0),0)

end

on agrega_viewport
  if(camara_front<> void) then
    crea_camaras
    -- camara_perpectiva.translate(5,0,10)

    --- genera marcos y relacion de las dos camaras de viewport
    camara_top.transform = sprite("modelo3d").camera.transform
    camara_front.transform = sprite("modelo3d").camera.transform
    sprite("modelo3d").addcamera(camara_top,3)
    sprite("modelo3d").addcamera(camara_front,4)
    camara_top.fieldofview = sprite("modelo3d").camera.fieldofview * 1
    camara_front.fieldofview = sprite("modelo3d").camera.fieldofview * 2
    camara_top.rotate(0,50,-40,escena.model("vastago"))
    camara_front.rotate(0,-40,0,escena.model("vastago"))
    --
    -- Asigna recuadro a camara_top
    camara_top.rect = rect(425,0,800,240)
    -- Asigna recuadro a camara_front
    camara_front.rect = rect(425,240,800,485)
    crea_marco_viewport

```

```

    end if
end

on remueve_todos_overlay
    repeat with i = 1 to sprite("modelo3d").camera.overlay.count
        sprite("modelo3d").camera.removeOverlay(i)
    end repeat

    agrega_marco_inicial

end

on crea_marco_viewport
    crea_camaras
    -- marco vertical
    if (escena.texture(v_barra_p)=void) then
        escena.newTexture(v_barra_p,#fromcastmember,member(v_barra))
    end if
    camara_front.addOverlay(escena.texture(v_barra),point(0,0),0)
    if (escena.texture("v_barra_p1")=void) then
        escena.newTexture("v_barra_p1",#fromcastmember,member(v_barra))
    end if

    camara_front.addOverlay(escena.texture(v_barra),point(0,240),0)
    if (escena.texture("v_barra_p2")=void) then
        escena.newTexture("v_barra_p2",#fromcastmember,member(v_barra))
    end if

    camara_top.addOverlay(escena.texture(v_barra),point(0,0),0)
    if (escena.texture("h_barra_p1")=void) then
        escena.newTexture("h_barra_p1",#fromcastmember,member(h_barra))
    end if

    camara_front.addOverlay(escena.texture(h_barra),point(370,0),0)
    if (escena.texture("h_barra_p2")=void) then
        escena.newTexture("h_barra_p2",#fromcastmember,member(h_barra))
    end if

    camara_top.addOverlay(escena.texture(h_barra),point(370,0),0)

    -- marco horizontal
    if (escena.texture("h_barra_viewport")=void) then
        escena.newTexture("h_barra_viewport",#fromcastmember,member(h_barra))
    end if
end

```

```

end if
camara_front.addOverlay(escena.texture(h_barra),point(0,0),0)
if (escena.texture("h_barra_viewport2")=void) then
    escena.newTexture("h_barra_viewport2",#fromcastmember,member(h_barra))
end if

camara_top.addOverlay(escena.texture(h_barra),point(0,0),0)
end

on nombre_objeto
    pt = the mouseLoc - point(sprite("modelo3d").left, sprite("modelo3d").top)
    nombre_obj = sprite("modelo3d").camera.modelUnderLoc(pt)
    -- member("texto").text = nombre_obj.name
end

on checa_mouse_coordenadas
    -- Asigna el area de trabajo y las coordenadas del mouse y la posicion del
    objeto
    -- member("texto").text = "Posición de cursor: x: " && _mouse.mouseH && "y:
    " && _mouse.mouseV

end

on partes
    -- member("texto").text= string(_mouse.mouseMember.name)
    --member("texto").text = "Posición de cursor: x: " && _mouse.mouseH && "y:
    " && _mouse.mouseV
    -- put nombre_obj
    if nombre_obj <> void then
        member("texto").text = nombre_obj.name

        case(nombre_obj.name) of
            "vastago": member("texto").text = "El vástago es compuesto de "

            -- otherwise : member("texto").text = "Posicione el cursor del ratón
            sobre la pieza, para ver sus características principales..."
        end case
    else
        member("texto").text = "Posicione el cursor del ratón sobre la pieza, para ver
        sus características principales..."
    end if
end
end

```

```

on exitframe me
  nombre_objeto
  partes

  if sprite("barra").hitTest(the mouseLoc) = #button then
    if the mouseDown = TRUE then
      boton= sprite("barra").getVariable("boton",false)
      put boton
    else
      boton =void
    end if
  else
    boton = void
  end if

  if (_mouse.stillDown) then
    _player.cursor(294)
  else
    _player.cursor(0)
  end if

  if (_mouse.rightMouseDown) then
    _player.cursor(290) -- Rotar y 290 close hand
  else

  end if

end

```