

Universidad de las Ciencias Informáticas

Facultad 6



Título: “Aplicación de escritorio para la representación gráfica de árboles genealógicos del sistema alasARBOGEN 2.0.”

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas.

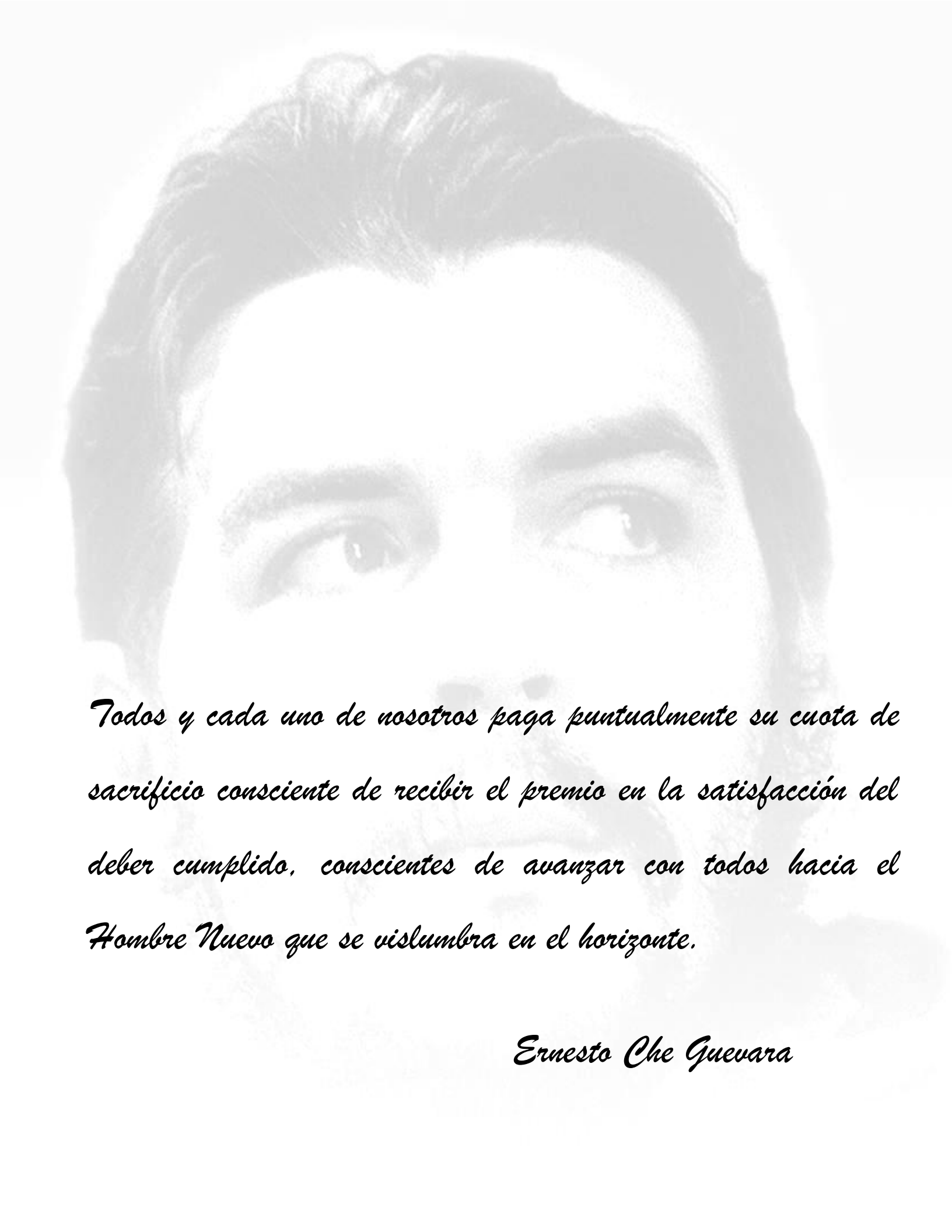
Autores:

Yosvany Arrastia Machin
Rayko Emilio Torres Cruz

Tutores:

Ing. Reynaldo Alvarez Luna
Ing. Reynaldo Rosado Roselló

Ciudad de La Habana, Junio 2010
“Año 52 de la Revolución”



Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio consciente de recibir el premio en la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte.

Ernesto Che Guevara

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Yosvany Arrastia Machin

Firma del Autor

Rayko Emilio Torres Cruz

Firma del Autor

Ing. Reynaldo Alvarez Luna

Firma del Tutor

Ing. Reynaldo Rosado Roselló

Firma del Tutor

DATOS DE CONTACTO

Ing. Reynaldo Alvarez Luna
Universidad de las Ciencias Informáticas, Habana, Cuba
rluna@uci.cu

Ing. Reynaldo Rosado Roselló
Universidad de las Ciencias Informáticas, Habana, Cuba
rrosado@uci.cu

AGRADECIMIENTOS

A nuestros tutores Reynaldo Luna, Reynaldo Rosado y Yunier Santana por el apoyo incondicional que siempre nos brindaron.

A Yadi, Yisel, Asencio, Silvio, Yoan y Rosnel, el equipo de desarrollo de alas ARBOGEN 2.0 y a todos aquellos que de una forma u otra nos ayudaron en el desarrollo de la tesis.

A todos nuestros amigos que desde primer año han compartido estos maravillosos 5 años.

En general agradecer todos aquellos que nos ayudaron a que hoy estemos aquí dedicándoles estas palabras.

DEDICATORIA

Rayko Emilio Torres Cruz

A mis padres Martha Cruz y Candido Torres que siempre me apoyaron en todo y me dieron fuerzas para continuar.

A mi tío Vicente, mis abuelitas Ana Margarita y Eliodora Prieto,

A mis hermanos Luis Armando, Raicel y Raidel que espero que algún día me dediquen unas palabras como estas.

A toda mi familia en general por todo su apoyo.

A todas mis amistades de la UCI.

Yosvany Arrastia Machin

A mis padres Mariolys Machin y Joaquin Arrastia, que siempre me apoyaron y fueron los que me inspiraron a llegar hasta el final.

A mis abuelos, por ayudarme y aconsejarme en todo momento.

A mis tíos por siempre estar ahí cuando necesitaba de su ayuda o consejo.

En general le agradezco a toda mi familia por su apoyo incondicional en este período tan importante de mi vida.

A todas mis amistades.

Al claustro de profesores que de una forma u otra posibilitó que me graduara como Ingeniero en Ciencias Informáticas.

RESUMEN

Desde el surgimiento del Centro Nacional de Genética Médica (CNGM), como centro rector de la red nacional de genética médica, se desarrollaron diversos estudios e investigaciones, que requirieron el diseño de aplicaciones informáticas para elevar la capacidad de análisis y la obtención de resultados en menor tiempo. Una de las herramientas básicas para el trabajo del genetista en la comunidad, en su trabajo preventivo es el árbol genealógico, por lo que se implementó en la Universidad de las Ciencias Informáticas. Surge la necesidad del desarrollo de una aplicación informática que permitió la representación gráfica de los árboles genealógicos, denominada alasARBOGEN 1.0 una solución informática que con el paso del tiempo y los avances tecnológicos, está imposibilitada para dar una prestación de servicios adecuados, atendiendo a las condiciones y necesidades actuales de los estudios genéticos en Cuba.

Como solución al problema se desarrolla en la UCI de conjunto con el CNGM, dentro del sistema alasARBOGEN 2.0, una aplicación de escritorio multiplataforma, que satisface las necesidades de los estudios genéticos en la actualidad. Permite guardar los datos en una Base de Datos (BD) centralizada, insertar y relacionar los individuos, exportar imágenes parciales o totales del árbol, exportar reportes e imprimir el árbol, así como los datos del mismo.

PALABRAS CLAVE

alasARBOGEN, Árboles Genealógicos, CNGM, Genética.

TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
DEDICATORIA	II
RESUMEN.....	III
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 ÁRBOLES GENEALÓGICOS.....	5
1.2 SISTEMAS PARA EL TRABAJO CON ÁRBOLES GENEALÓGICOS.....	5
1.2.1 GenoPro.....	5
1.2.2 BitGen II.....	6
1.2.3 GDS (Sistema General de Documentación Familiar)	6
1.2.4 GenealogíaMac	8
1.2.5 Cyrillic.....	9
1.2.6 alasARBOGEN 1.0	10
1.3 METODOLOGÍAS DE DESARROLLO.....	10
1.3.1 RATIONAL UNIFIED PROCCES.....	10
1.4 LENGUAJES DE PROGRAMACIÓN.....	12
1.4.1 JAVA	12
1.5 HERRAMIENTAS A UTILIZAR.....	13
1.5.1 ECLIPSE 3.3 EUROPA.....	13
1.5.2 VISUAL PARADIGM.....	14
CONCLUSIONES	14
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	15
2.1 OBJETO DE ESTUDIO.....	15
2.1.1 OBJETIVOS ESTRATÉGICOS DE LA ORGANIZACIÓN	15
2.1.2 FLUJO ACTUAL DE LOS PROCESOS.....	15
2.2 OBJETO DE AUTOMATIZACIÓN.....	16
2.3 MODELO DE DOMINIO.....	16
2.4 ESPECIFICACIÓN DE LOS REQUISITOS DE LA APLICACIÓN.....	18
2.4.1 REQUERIMIENTOS FUNCIONALES.....	18
2.4.2 REQUERIMIENTOS NO FUNCIONALES.....	19
2.5 DEFINICIÓN DE LOS CASOS DE USO DEL SISTEMA.....	21
2.5.1 ACTORES DEL SISTEMA.....	21
2.5.2 DIAGRAMA DE CASOS DE USO DEL SISTEMA.....	22
2.5.3 DESCRIPCIÓN DE LOS CASOS DE USO DEL SISTEMA.....	22
2.5.3.1 BUSCAR EN EL ÁRBOL GENEALÓGICO.....	23
2.5.3.2 ORDENAR ÁRBOL GENEALÓGICO.....	29
2.5.3.3 MOSTRAR VISTA PREVIA DEL ÁRBOL GENEALÓGICO.....	30
2.5.3.4 IMPRIMIR ÁRBOL GENEALÓGICO.....	31

2.5.3.5 GENERAR REPORTE DEL ÁRBOL GENEALÓGICO.....34

CONCLUSIONES35

CAPÍTULO 3: DISEÑO DEL SISTEMA.....36

3.1 ESTILO ARQUITECTÓNICO UTILIZADO.36

3.1.1 MODELO VISTA CONTROLADOR (MVC)36

3.2 PATRONES DE DISEÑO UTILIZADOS.....37

3.2.1 PATRÓN STATE37

3.2.2 PATRÓN SINGLETON37

3.2.3 PATRONES GRASP.....38

3.2.3.1 PATRÓN EXPERTO.....38

3.2.3.2 PATRÓN CREADOR.....38

3.3 DIAGRAMAS DE CLASES.39

3.4 DIAGRAMAS DE SECUENCIAS.42

3.5 VISOR DE PROPIEDADES46

CONCLUSIONES47

CAPÍTULO 4: IMPLEMENTACIÓN48

4.1 DIAGRAMA DE COMPONENTES.48

4.2 FRAGMENTOS DE CÓDIGO49

4.2.1 Visor de propiedades49

4.2.2 Entorno visual52

4.3 PRUEBAS53

CONCLUSIONES60

CONCLUSIONES61

RECOMENDACIONES62

REFERENCIAS BIBLIOGRÁFICAS.....63

BIBLIOGRAFÍA.....65

ANEXOS.....66

GLOSARIO71

INTRODUCCIÓN

Una de las principales prioridades de la Revolución Cubana, es la salud pública, que siempre ha estado encaminada a mejorar las condiciones de vida de toda la población, llevando la salud hasta los lugares menos accesibles con un carácter preventivo.

Con el propósito de ampliar las investigaciones y de extender el tratamiento y cuidado del sistema primario de salud, Fidel inaugura el CNGM el 5 de agosto de 2003, para llevar a cabo un monitoreo más detallado de todas las enfermedades cuyas causas fueran genéticas [1].

Junto con el CNGM, se concibe la red nacional, la cual está conformada por 184 centros de genética, de los cuales son 169 municipales, 14 provinciales y el CNGM que es la institución rectora de estos. El principal objetivo de ellos, es llevar a cabo el programa nacional para el diagnóstico, manejo y prevención de Enfermedades Genéticas y Defectos Congénitos [1].

Un aspecto fundamental en el trabajo de los genetistas para lograr un estudio profundo de las enfermedades genéticas en nuestra población, es la representación gráfica de árboles genealógicos de los pacientes que se presentan a la consulta.

Una de las soluciones puestas en práctica en el CNGM para la representación de árboles genealógicos, fue el uso del Cyrillic. Este software es una herramienta propietaria para la representación de árboles genealógicos, que no cubre las demandas de prestaciones necesarias para el trabajo de los especialistas y las necesidades del país para el estudio genético. Para el uso de dicha herramienta, el país se ve en la necesidad de pagar la licencia por cada estación de trabajo en la que se vaya a utilizar. Esto constituye un gasto financiero considerable que el país no puede cubrir. Entre las principales deficiencias que tiene este software, se encuentran que es una herramienta que está disponible solamente en idioma inglés y no es adaptable a las características del Sistema de Salud Cubano.

A partir de las deficiencias encontradas en la solución que se estaba poniendo en práctica en el CNGM, se decidió en conjunto con el Ministerio de Informática y las Comunicaciones, específicamente en la Universidad de las Ciencias Informáticas (UCI), desarrollar una aplicación (alasARBOGEN 1.0) que cumpliera con los requisitos que se necesitaban para los estudios genéticos en el país.

Actualmente en los centros de genética médica, los genetistas hacen la representación del árbol genealógico haciendo uso de la aplicación alasARBOGEN 1.0, siendo esta de gran ayuda para el trabajo y disminuyendo el costo para la economía cubana. En la actualidad los estudios genéticos del país tienen necesidades diferentes de las que existían en el momento del desarrollo, por lo que no cubre las actuales

demandas, y los especialistas detectaron un conjunto de problemas relacionados con la organización del entorno de trabajo y la interfaz rígida del sistema, aspectos que influyen negativamente en el tiempo que deben dedicar a construir los árboles genealógicos en las consultas y analizarlos.

La organización visual debe estar dirigida a la organización generacional existente. Para poder llevar a cabo un mayor seguimiento y control en el estudio de un individuo, se evidencia la necesidad de poder hacer una buena gestión de las muestras de laboratorio, así como el almacenamiento de los resultados arrojados por estas. El control de las enfermedades de tipo genéticas, es además, un pilar fundamental para el trabajo diario de los genetistas, así como su forma de representar cierta afectación de un individuo en su árbol genealógico. Las relaciones existentes entre las personas, constituyen un aspecto fundamental para determinar posibles afectaciones futuras en descendientes de una pareja, es por ello que un control adecuado de las relaciones entre las personas, forma parte de una de las principales necesidades de los estudios genéticos en la actualidad.

En la red de salud nacional “Infomed” existen una amplia gama de servicios que dándoles el uso necesario, haría el trabajo de los especialistas más eficiente, es por ello que se requiere de una conexión con dichos servicios, con el objetivo de hacer una alimentación necesaria de los datos pertinentes de los pacientes que se estén estudiando.

Atendiendo a las características del país y la situación actual de nuestra economía, constituye una prioridad de primer orden contar con entornos de trabajo que sean multiplataforma. El objetivo principal es minimizar los gastos económicos relacionados con la compra de licencias para software propietarios. Estas necesidades existentes hacen de la versión 1.0 de alasARBOGEN, una aplicación que presenta carencias para su utilización en la representación gráfica de los árboles genealógicos.

El **problema científico de la investigación** se ha determinado como: La necesidad de representación gráfica de árboles genealógicos dada las condiciones actuales de los estudios genéticos en Cuba.

Se define como **objeto de estudio**: Los sistemas para la representación gráfica de árboles genealógicos.

El **campo de acción**: Las aplicaciones de escritorio multiplataforma para la representación gráfica de árboles genealógicos.

Para darle cumplimiento al problema científico expuesto, se define como **objetivo general** de la investigación, desarrollar una aplicación de escritorio multiplataforma para la representación gráfica de árboles genealógicos, que satisfaga los requerimientos actuales de los estudios genéticos en Cuba.

Para darle cumplimiento al objetivo general se plantean como **objetivos específicos**:

- Identificar las funcionalidades de la nueva versión.
- Diseñar la aplicación de escritorio para la representación de árboles genealógicos.
- Implementar la aplicación.

En función de los objetivos específicos enunciados, se definen como tareas de la investigación:

- Investigación de las tendencias y tecnologías para desarrollar aplicaciones de escritorio sobre plataformas libres.
- Modelación del negocio.
- Identificación de los requerimientos funcionales y no funcionales.
- Descripción de la arquitectura propuesta.
- Realización del modelo de Diseño.
- Realización del modelo de implementación.
- Implementación de los componentes.
- Realización de pruebas funcionales.

Aporte práctico esperado del trabajo:

Una aplicación de escritorio desarrollada en Java (Multiplataforma) que contiene las funcionalidades de la primera versión del sistema, integra además la persistencia de datos en BD y una serie de técnicas avanzadas para el cálculo de indicadores en el árbol. Permitirá la integración a alasMEDIGEN y aportará componentes visuales para su reutilización en problemas similares.

El presente trabajo estará estructurado por 4 capítulos. A continuación se describe de manera breve el contenido de cada uno de ellos.

Capítulo 1: Fundamentación Teórica: En este capítulo se abordará el estado actual de las aplicaciones para construir árboles genealógicos en el mundo y en nuestro país. Serán abordados brevemente temas relacionados con la genética médica y los árboles genealógicos. Se incluye además un estudio y selección de las metodologías, herramientas y tecnologías que se van a utilizar en el desarrollo de la aplicación, atendiendo a las tendencias actuales en el desarrollo de software a nivel mundial.

Capítulo 2: Características del Sistema: En este capítulo se describe el objeto de estudio. Se explica el problema existente en el CNGM relacionado con alasARBOGEN 1.0, el cual dio paso al desarrollo de esta investigación. Se especifican los requerimientos de la aplicación, y se definen los casos de uso, así como una descripción textual. Se presenta una propuesta del sistema.

Capítulo 3: Diseño del Sistema: En este capítulo se le da cumplimiento a los requisitos a través del diseño de la aplicación, describiendo como implementar, enfocado a como el sistema cumple los objetivos planteados en los requisitos funcionales y no funcionales. Se explica la arquitectura utilizada así como los principales patrones y se realizan los diagramas de clases y los diagramas de interacción, según los casos de uso más relevantes definidos en el capítulo anterior.

Capítulo 4: Implementación: En este capítulo se describe como fue implementada la aplicación. Se muestran algunos fragmentos de códigos no triviales de algunas funcionalidades y clases importantes. Se describen las pruebas funcionales realizadas al sistema y algunas vistas de la aplicación.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este capítulo se abordará el estado actual de las aplicaciones para construir árboles genealógicos en el mundo y nuestro país. Se abordarán brevemente temas relacionados con la genética médica y los árboles genealógicos. Se incluye además un estudio y selección de las metodologías, herramientas y tecnologías que se van a utilizar en el desarrollo de la aplicación, atendiendo a las tendencias actuales en el desarrollo de software a nivel mundial.

1.1 ÁRBOLES GENEALÓGICOS.

El árbol genealógico es una representación gráfica que expone los datos genealógicos de un individuo en una forma organizada y sistemática, sea en forma de árbol o tabla. Puede ser ascendente, es decir, que expone los antepasados o ancestros de un sujeto, o puede ser descendente, que expone todos los descendientes del sujeto. Para realizar un árbol genealógico es necesario, primeramente, haber realizado una investigación genealógica o genealogía del individuo [2].

1.2 SISTEMAS PARA EL TRABAJO CON ÁRBOLES GENEALÓGICOS.

Fueron analizados como parte de la investigación científica una representación de los sistemas más conocidos en la representación de árboles genealógicos existentes en la actualidad. La gran mayoría de estos sistemas están orientados a la construcción del árbol familiar, sin llegar a hacer un análisis profundo y un seguimiento de las enfermedades genéticas de un individuo de la familia. Estos son software que en la mayoría de las ocasiones se difunden por internet y están siempre disponibles en la web. Están basados en la construcción de árboles familiares, brindando la posibilidad de añadirles fotos personales y una diversidad de datos, según las posibilidades de la aplicación. A continuación, se hará un estudio de algunos ejemplos de los sistemas existentes.

1.2.1 GenoPro

GenoPro es una aplicación para dibujar árboles familiares y genogramas. Es un software para graficar árboles genealógicos que ha tenido gran aceptación a nivel mundial y se encuentra disponible en 25 idiomas. Permite construir el árbol genealógico con un asistente familiar que brinda la posibilidad de agregar o eliminar hijos con gran sencillez [3].

GenoPro permite la posibilidad de dibujar una familia sin importar cuán compleja. Cuenta con un rango completo de funcionalidades para hacer análisis y manejar escenarios complejos, y una vez terminado, brinda la posibilidad de hacer salvadas en archivos que se pueden cargar en una gran variedad de aplicaciones de Windows y ayuda a crear un CD con el árbol familiar.

Se conoce que es una aplicación muy utilizada por genealogistas, terapeutas, trabajadores sociales, profesionales de la salud e investigadores [3].

Para poder utilizar esta aplicación se requiere del pago de la licencia de propietario.

1.2.2 BitGen II

BitGen v2.0 es otra de las aplicaciones que se encuentran disponibles en el mercado para la confección de Árboles Genealógicos de una familia, partiendo de datos previamente almacenados relativos a esta. Es una aplicación en español, que tiene como objetivo principal, facilitar la elaboración de árboles genealógicos en una computadora [4].

La interfaz de BitGen II para la confección de un árbol genealógico presenta cuatro pergaminos de tamaño DIN-A4, bordeados por una pareja de reglas para medir adecuadamente la posición de cada elemento gráfico. Además, contiene una barra de herramientas y los controles de movimientos, desde donde se pueden indicar todas las funciones que se quieren realizar [5].

En esta versión se incluye un apartado de Heráldica en el que aparecen algunos de los linajes españoles más comunes. Se ha incluido también un diccionario heráldico, con los elementos más comunes y explicaciones sobre su uso [4]. Por otro lado, presenta una enciclopedia heráldica, donde se muestra una lista con todos los apellidos que contiene la aplicación, acompañados del blasón correspondiente a ese linaje. Además, añade información sobre el origen e historia del apellido. Una delicia para los aficionados a esta materia [5].

BitGen II es la segunda versión de un software al que aún le falta dinamismo, rapidez, intuición y un funcionamiento más cercano al de aplicaciones destinadas a trabajar en entornos operativos Windows [5]. Es un software propietario.

1.2.3 GDS (Sistema General de Documentación Familiar)

Es una aplicación que está basada en la estructura de dos apellidos y es multilingüaje, encontrándose disponible en la actualidad en Español, Catalán e Inglés. Sus características extendidas lo convierten en un sistema que va más allá de la pura genealogía, convirtiéndolo en un repositorio de toda la información

familiar, ya sea de estructura familiar, de almacenamiento de documentos, fotografías, voz, videos y multimedia[6].

Es una aplicación que presenta una gama de funciones únicas, sobresalientes a otros programas de genealogía, tal como [6]:

- **Uso de dos apellidos:** Aceptación de los sistemas Español, Portugués (Apellidos inversos) y anglosajón, en la misma Base de Datos.
- **Multilinguaje dinámicamente:** Cambio de idioma sin parar la ejecución del programa.
- **Análisis de apellidos:** Hasta más de 250000 primeros apellidos, por el método Melgar.
- **Cálculo de la relación familiar:** entre personas, aún en relaciones lejanas, por uniones conyugales múltiples.
- **Álbum de Foto:** Cada fotografía puede tener hasta 200 rectángulos vinculados a personas de la Base de Datos u otras informaciones. Álbum temático. Establecimiento de una jerarquía de temas en los cuales se pueden incluir un número ilimitado de fotografías.
- Creación de **documentos de texto** (utilizando su editor de textos, **Microsoft Word** ©, por ejemplo), **archivos con su voz**, o entrada de fotografías desde el **escáner**, desde cualquier directorio o desde una **cámara digital**, directamente desde GDS.
- Creación de archivos PDF o TIF multipágina a partir de escáner. - Aceptación de cualquier documento **multimedia**, formando parte de su Base de Datos.
- **Importación** de archivos **GEDCOM** (Genealogy communication), el estándar de comunicación entre genealogistas.
- **Exportación** a archivos **GEDCOM** para ser importados por otros programas de Genealogía.
- **Exportación** a archivos de texto para ser explotados por otros programas. (Por ejemplo **Microsoft Excel**).
- Creación de su propia **Web** con todo o parte del contenido de la Base de Datos.
- Creación de la **página HTML** de una persona.
- Envío de fotografías e informes directamente por **e-mail**.

La última versión disponible es **GDS 6.0.18** a la cual se le agregó o mejoró algunas funciones, entre las cuales se resaltan [6]:

- Posibilidad de indicar que una persona está muerta durante la actualización aunque no se conozca ningún dato de su fallecimiento.
- Búsqueda avanzada de personas que dada una fecha de nacimiento (Exacta, antes de, después de, entre dos años), estén vivas en este momento. Lo mismo ocurre para las personas fallecidas. Se considera, además, que si no hay datos de fallecimiento de una persona, ésta es fallecida si han transcurrido 120 años desde su nacimiento a efectos de esta búsqueda.
- Posibilidad de eliminar una o más personas y su descendencia del gráfico de descendientes. Una vez desplegado el gráfico, mediante el ratón se eliminan las personas que no se desea que aparezcan, bien ellas mismas, bien su descendencia.
- Durante la comprobación de las cadenas se añade la de existencia de bucles de antepasados.

Es un software propietario.

1.2.4 GenealogíaMac

Esta es una aplicación que en su más reciente versión, **GenealogíaMac v4.4.1** está diseñado para crear árboles genealógicos de todo tipo. Permite agregar fotos, personalizar colores y fuentes. Se divide en tres áreas fundamentales: edición, visualización y publicación de los árboles genealógicos.

Editar permite introducir nuevas personas y familias, establecer la pertenencia de hijos a las familias, y especificar la relación entre los padres. Debajo de los campos Apellidos y Nombre, se encuentra la lista de informaciones adicionales que se puede configurar libremente para introducir otros datos. Los iconos al lado de las personas, permiten introducir o borrar personas de la Ficha familiar, abrir ventanas de información, y acceder a los padres de una persona [7].

A la hora de visualizar ofrece múltiples posibilidades para la representación gráfica de sus datos recopilados. La tabla de antepasados, la tabla de descendientes. **GenealogíaMac** deja la libertad para configurar el gráfico según sus preferencias, con o sin imágenes. Las Líneas de la vida sirven de orientación general y la Estadística muestra las edades que alcanzaron las personas en su árbol genealógico calculando el promedio. En la Lista se detallan todas las personas de su árbol genealógico ordenadas según el principio que le convenga. El resumen contiene los datos de la Ficha familiar actual. Se pueden imprimir todas y cada una de las representaciones [7].

Permite grabar su árbol completo con todas las informaciones e imágenes en CD. Esta utilidad puede servirle tanto para realizar copias de seguridad como para distribuir el árbol genealógico entre sus familiares [7].

Permite exportar el árbol completo en formato HTML para su publicación en internet. Para este fin el programa está provisto de las herramientas para configurar la presentación deseada para internet e incluir las imágenes correlacionadas. Genera copias comprimidas de sus imágenes automáticamente para garantizar una visualización óptima en internet [7].

Es una aplicación propietaria, que incluye variadas opciones que no son necesarias en los estudios genéticos cubanos.

1.2.5 Cyrillic

Posee todas las características necesarias para dibujar líneas genéticas. Es el más completo, funcional y fácil de usar. Es un software diseñado para el manejo de datos genéticos en el sistema operativo Windows. Todas las herramientas están dispuestas de forma que usted pueda, rápida y fácilmente, manejar los datos de los que disponga [8].

Sus interfaces están diseñadas de forma agradable y permiten moverse sin ningún esfuerzo de una fase del proceso a otra. La implantación de numerosas barras de herramientas especiales y cajas de diálogo hacen que su manejo sea muy intuitivo, pero si usted necesitara ayuda, todas las respuestas a sus cuestiones están disponibles instantáneamente a través de una completa ayuda en línea [8].

Cyrillic puede almacenar un elevado rango de datos para cada individuo, incluyendo información personal, clínica y genética. Todos los datos son accesibles a través de una caja de diálogo y los cambios que hagamos en ellos son reflejados automáticamente en el dibujo del árbol genético [8].

Como características importantes presenta:

- Actualización automática de los datos mientras se está dibujando.
- Perfeccionamiento del manejo para casos de gemelos y embarazos múltiples.
- Cálculo de riesgos para enfermedades familiares, permitiendo hacer el cálculo de riesgo de enfermedad para un individuo dentro de una familia donde exista una determinada enfermedad.
- Automatización del proceso de construcción de los árboles genealógicos, lo cual es necesario en el trabajo diario de los genetistas.

Esta aplicación es propietaria, esto implica un gasto financiero muy grande para la economía del país.

1.2.6 alasARBOGEN 1.0

alasARBOGEN 1.0 es una aplicación de escritorio capaz de darle solución a las necesidades genéticas existentes en Cuba en el momento de su desarrollo. Entre sus principales características se encuentra, la disponibilidad en idioma español, tiene en cuenta las peculiaridades de nuestro país, haciendo un uso adecuado de la simbología internacional. Cuenta con una serie de limitaciones ante las necesidades actuales, que hacen de esta aplicación una opción que no es factible en la actualidad. Presenta una pobre gestión de las muestras de laboratorio y el manejo de las relaciones entre los individuos. En cuanto a la estructura y organización gráfica del trabajo, presenta problemas ya que el árbol genealógico no queda organizado por generación, forzando al genetista a perder tiempo en la organización del árbol. Como uno de los principales problemas que presenta, que se encuentra desarrollado en software propietario, siendo posible su uso, solamente para ordenadores que cuenten con el sistema operativo Windows XP SP 2 en español y que cuente con la versión 1.1 o superior del framework .NET.

1.3 METODOLOGÍAS DE DESARROLLO.

Se entiende por metodologías de desarrollo de software, a una colección de documentación formal que hace referencia a los procesos, las políticas y los procedimientos que intervienen en el desarrollo de un software. Tiene como finalidad garantizar la eficacia (cumplimiento de los requisitos) y la eficiencia (minimizar las pérdidas de tiempo) del proceso de desarrollo del software [9].

En las metodologías los procesos se descomponen en niveles inferiores, convirtiéndose en tareas o actividades fundamentales.

1.3.1 RATIONAL UNIFIED PROCESS (RUP).

RUP es una metodología de desarrollo de software orientado a objetos, que consiste en un conjunto de actividades necesarias para transformar los requerimientos de usuarios en el sistema de software. Está especializado para diversos tipos de software de sistemas, diversas áreas de aplicación, diferentes tipos de organizaciones y diferentes tamaños de proyectos [11].

RUP tiene una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo con el objetivo de desarrollar software de alta calidad, teniendo en cuenta quién hace qué, cuándo y cómo [12]. Su meta es asegurar la producción del software de alta calidad, que resuelva las necesidades de los usuarios dentro de un presupuesto y tiempo establecidos [13].

El proceso de desarrollo de RUP está definido por tres características especiales: Dirigido por Casos de Uso, Centrado en la Arquitectura, Iterativo e Incremental. Dentro de la metodología seleccionada se van a desempeñar los siguientes roles, obteniendo los artefactos que se hacen referencia a continuación.

✓ **Roles y Artefactos**

Un rol es una definición abstracta de un conjunto de actividades realizadas y de artefactos obtenidos. Los roles son realizados típicamente por un individuo, o un conjunto de individuos, trabajando juntos en equipo. Un miembro del equipo de proyecto cumple normalmente muchos roles. Los roles describen cómo los individuos se comportan en el negocio y qué responsabilidades tienen [14].

De acuerdo con las necesidades de la investigación, se desarrollarán fundamentalmente, los roles de analista de sistema, diseñador y desarrollador. Estos están comprendidos dentro de los grupos analistas y desarrolladores definidos en los equipos de trabajo de esta metodología de desarrollo.

Analista de Sistema: Conduce y coordina los requerimientos y los Casos de Uso, modelando y delimitando la funcionalidad del sistema; por ejemplo, estableciendo que actores y casos de uso existen y cómo interactúan [14].

Los artefactos a realizar desempeñando el rol de analista de sistema son:

- ✓ **Modelo de casos de uso:** Es un modelo de funciones previstas del sistema y su entorno, que está compuesto por casos de usos, actores del sistema y sus relaciones.
- ✓ **Glosario de términos:** Define los términos comunes que se utilizan en el proyecto.

Diseñador: El diseñador es el encargado de diseñar una parte del sistema dentro de los límites de los requisitos. Debe ser consecuente con la arquitectura y el proceso de desarrollo del proyecto.

Los artefactos a realizar desempeñando el rol de diseñador son:

- ✓ **Realización de los casos de uso:** La realización de los casos de usos, describe la forma particular en que un caso de uso es realizado dentro del modelo de diseño.
- ✓ **Clases del diseño:** Las clases del diseño son la descripción de cómo un grupo de objetos comparten las mismas responsabilidades, relaciones, operaciones y atributos.
- ✓ **Paquetes del diseño:** Es un conjunto de clases, relaciones, realizaciones de casos de uso, diagramas y otros paquetes. Se utiliza para estructurar el modelo de diseño dividiéndolo en partes más pequeñas.

Desarrollador: Es el responsable del desarrollo y pruebas de componentes, de acuerdo con las normas aprobadas del proyecto, para su integración en grandes subsistemas. Es responsable además de desarrollar y probar los componentes y subsistemas correspondientes.

Los artefactos a realizar desempeñando el rol de implementador son:

- ✓ **Pruebas de desarrollo:** Son las pruebas que efectúan los desarrolladores en el transcurso de la implementación del sistema.
- ✓ **Elementos de implementación:** Son los componentes físicos que forman el sistema, incluyen los directorios y los archivos. Incluyen además los archivos de código del software (fuentes, binarios o ejecutables), archivos de datos, archivos de documentación, así como los de ayuda en línea.
- ✓ **Artefactos de instalación:** Los artefactos hacen referencia al software y las documentaciones necesarias para la instalación del producto.

1.4 LENGUAJE DE PROGRAMACIÓN.

Un lenguaje de programación, que está especialmente diseñado para escribir instrucciones que sean capaces de darle órdenes a una computadora, para hacer procesos determinados e interactuar con el hardware. Está definido por una gramática y un conjunto de reglas y símbolos definidos en un alfabeto [15].

1.4.1 JAVA

Java es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems a principios de los años 90. En sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

✓ **Sintaxis:**

La sintaxis de Java se deriva en gran medida de C++. Pero a diferencia de éste, que combina la sintaxis para programación genérica, estructurada y orientada a objetos, Java fue construido desde el principio para ser completamente orientado a objetos. Todo en Java es un objeto (salvo algunas excepciones), y todo reside en alguna clase.

El diseño de Java, su robustez, el respaldo de la industria y su fácil portabilidad han hecho de Java uno de los lenguajes con un mayor crecimiento y amplitud de uso en distintos ámbitos de la industria de la informática.

✓ En aplicaciones de escritorio

Hoy en día existen multitud de aplicaciones gráficas de usuario basadas en Java. El entorno de ejecución Java (JRE), se ha convertido en un componente habitual en los PC de usuarios de los sistemas operativos más usados en el mundo. Además, muchas aplicaciones Java lo incluyen dentro del propio paquete de la aplicación de modo que se ejecuten en cualquier PC.

En las primeras versiones de la plataforma Java, existían importantes limitaciones en las APIs de desarrollo gráfico (AWT). Desde la aparición de la biblioteca Swing la situación mejoró substancialmente.

✓ Plataformas soportadas

Una versión del entorno de ejecución Java JRE (Java Runtime Environment) está disponible en la mayoría de equipos de escritorio. Sin embargo, Microsoft no lo ha incluido por defecto en sus sistemas operativos. En el caso de Apple, éste incluye una versión propia del JRE en su sistema operativo, el Mac OS. También es un producto que por defecto aparece en la mayoría de las distribuciones de GNU/Linux [16].

Se decide utilizar Java por las ventajas que el mismo proporciona, tales como: sencillez y eficiencia, multiplataforma y portabilidad, por lo que se concluye que siendo utilizado de forma adecuada, se logrará cumplir con eficiencia los objetivos planteados.

1.5 HERRAMIENTAS A UTILIZAR.

1.5.1 ECLIPSE 3.3 EUROPA

Eclipse es un Entorno Integrado de Desarrollo (IDE), para todo tipo de aplicaciones libres. Es una herramienta para el programador, principalmente para el desarrollo de aplicaciones Java, facilitando al máximo la gestión de proyectos mediante el control de versiones CVS o con subversión (con subclipse), es posible también exportar e importar proyectos.

Permite añadir nuevas funcionalidades al editor, a través de nuevos módulos ('plugins'), para programar en otros lenguajes de programación además de Java como C/C++, PHP, Python, Ruby, Cobol, etc. Es un IDE Multiplataforma (GNU/Linux, Solaris, Mac OSX, Windows), soportado para distintas arquitecturas. Estructurado por plugin, hace sencillo añadir nuevas características y funcionalidades. Tiene resaltado de sintaxis, autocompletado, tabulador de un bloque de código seleccionado. Brinda asistentes (Wizards) para la creación, exportación e importación de proyectos y para generar esqueletos de códigos (templates) [17].

1.5.2 VISUAL PARADIGM

Visual Paradigm es una herramienta CASE para UML que es muy fácil de usar, soporta la última anotación UML 2.1, ingeniería inversa así como directa, generación de código, importación desde Rational Rose, exportación/importación XML, generador de informes, editor de figuras e integración con MS Visio, plugin, integración IDE con Eclipse, NetBeans y otros.

Es una herramienta que provee el modelado de negocio y un lenguaje de mapeo para lenguajes de programación de alto nivel como son Java y .NET.

Es un estándar que se encuentra ampliamente utilizado en el mundo por las empresas, para el modelado en el proceso de desarrollo de software. Una de las principales características que presenta es que su diseño se ve centrado en los casos de uso y enfocado en el desarrollo de un software con mayor calidad. Genera la documentación del proyecto en distintos formatos, destacándose el formato pdf y permite el control de versiones [18].

CONCLUSIONES

Después de hacer un análisis del estado del arte de los software para la representación de árboles genealógicos, se evidencian las dificultades para su aplicación en nuestro país, relacionadas fundamentalmente con la dificultad para la adquisición de licencias y la falta de correspondencia con las características de los estudios genéticos de Cuba. Se describieron las características fundamentales de las tecnologías, metodología y herramientas seleccionadas para el desarrollo de la investigación, definidas previamente por el equipo de arquitectos del sistema.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

En este capítulo se describe el objeto de estudio. Se explica el problema existente en el CNGM relacionado con las ARBOGEN 1.0, el cual dio paso al desarrollo de esta investigación. Se especifican los requerimientos de la aplicación, y se definen los casos de uso, así como una descripción textual, presentando una propuesta del sistema.

2.1 OBJETO DE ESTUDIO

2.1.1 OBJETIVOS ESTRATÉGICOS DE LA ORGANIZACIÓN

“La red nacional de genética médica está estructurada por 184 centros de genética ubicados en todos los municipios y provincias del país que coordinados por el Centro Nacional de Genética Médica conducen el Programa Nacional para el diagnóstico, manejo y prevención de enfermedades genéticas y defectos congénitos. Esta red institucional realiza acciones asistenciales, docentes y de investigación en el campo de las enfermedades, con carácter genético en la población cubana, con el propósito de mejorar la calidad de vida y elevar el bienestar de nuestro pueblo (Centro Nacional de Genética Médica).” [19]

En la actualidad la red institucional de genética médica brinda servicios de consulta a la población, donde los centros municipales y provinciales constituyen los centros de atención primaria. Los pacientes se presentan a las consultas de los genetistas y se determina si es necesario hacer una representación gráfica del árbol genealógico para estudiar el caso.

2.1.2 FLUJO ACTUAL DE LOS PROCESOS.

Después de analizar el cuadro clínico del paciente el genetista decide si se le va a realizar un estudio genético y si es necesario hacer la representación gráfica del árbol genealógico. Cuando se decide hacer la representación gráfica del árbol genealógico, el genetista procede a preguntarle todos los vínculos familiares, relaciones de familias y relaciones de parejas, así como los datos que puedan aportar información útil para el estudio genético. Una vez que se recoge toda la información el genetista procede a realizar el árbol genealógico en la aplicación las ARBOGEN 1.0 y posteriormente procede a guardarlo en fichero, no existiendo una forma de almacenamiento centralizada. El trabajo de introducir la información referente a los pacientes y familiares, es un proceso que se dificulta, debido a la falta de integración que tiene las ARBOGEN 1.0 con las diferentes aplicaciones de salud. Esta información se encuentra disponible en diferentes sistemas, como es el caso del Registro del Ciudadano (RIS), donde se

encuentran todos los datos primarios de los pacientes. Existe además falta de integración con alasMEDIGEN, que es el Sistema de Genética donde están disponibles los registros de enfermedades genéticas. En el caso de que los genetistas necesiten hacer un intercambio de criterios sobre cierto árbol genealógico, se ven obligados a hacer un intercambio vía correo electrónico para que el otro especialista pueda emitir su criterio sobre el caso que se está estudiando.

2.2 OBJETO DE AUTOMATIZACIÓN.

La aplicación permitirá hacer una representación del árbol genealógico de un paciente. Recoger la información de interés para el estudio genético perteneciente al individuo, a la familia, así como a las relaciones de pareja y datos de interés que puedan aportar información útil para el estudio del caso. Presenta una mejora en la gestión visual de los individuos, así como en las operaciones de inserción y actualización de la información representada en el árbol genealógico. El manejo de las relaciones existentes entre los individuos del árbol genealógico y la simbología para la representación de las enfermedades genéticas se hará de una forma más eficiente, al igual que la gestión de las muestras de laboratorio, posibilitando que se tenga un seguimiento de los pacientes de interés. Como nuevas funcionalidades a automatizar se encuentran, una vista previa de impresión y un servicio de reporte en el formato pdf. Se garantizará la persistencia de los datos en una BD centralizada, así como la persistencia de los datos en archivos binarios en la PC. Se automatizará el proceso de exportación del árbol genealógico que se está estudiando en formato jpg. Permitirá realizar búsquedas internas en el árbol genealógico atendiendo a diferentes criterios según sea de interés para el genetista y el ordenamiento gráfico del árbol genealógico atendiendo a la organización generacional que este presenta.

2.3 MODELO DE DOMINIO.

Para el desarrollo del sistema alasARBOGEN 2.0 se tiene en cuenta el negocio descrito en alasARBOGEN 1.0 [19]. Atendiendo a las necesidades actuales de los estudios genéticos, el sistema debe ser capaz de integrarse con los servicios que brinda Infomed, es por ello que se propone el modelo de dominio que se muestra en la figura Fig. 1. En el modelo de dominio se exponen los principales conceptos existentes en el negocio, quedando en evidencia que no existe ninguna integración de alasARBOGEN 1.0 con el Registro del Ciudadano (RIS), donde se encuentran todos los datos primarios de las personas y AlasMEDIGEN que es el sistema de genética, donde se encuentran registrados todas

las enfermedades genéticas, malformaciones y defectos congénitos. Esta falta de conectividad hace que el proceso de búsqueda de datos de los pacientes sea muy complejo, e incluso en ocasiones haya que introducir o actualizar los datos de forma manual en la mayoría de los estudios genéticos, no pudiendo contemplarse de esta forma estudios anteriores pertenecientes al paciente o familiares del mismo. No existe además una forma de persistencia centralizada de los datos gestionados por alasARBOGEN 1.0, desaprovechándose las condiciones de conectividad existentes en los Centros Municipales de Genética Médica (CMGM), Centros Provinciales de Genética Médica (CPGM) y Centro Nacional de Genética Médica (CNGM), así como otras instituciones de salud interesadas en el uso de la aplicación. Se dificulta el intercambio de información de forma segura entre distintos especialistas, debido a la imposibilidad de consumir el servicio que brinda el componente encargado de la Autenticación, Autorización y Trazas (SAAA) de la red de salud. Este intercambio de información podría ser más provechoso en el análisis de los casos que se estén estudiando y aprovechado en el desarrollo de un buen diagnóstico.

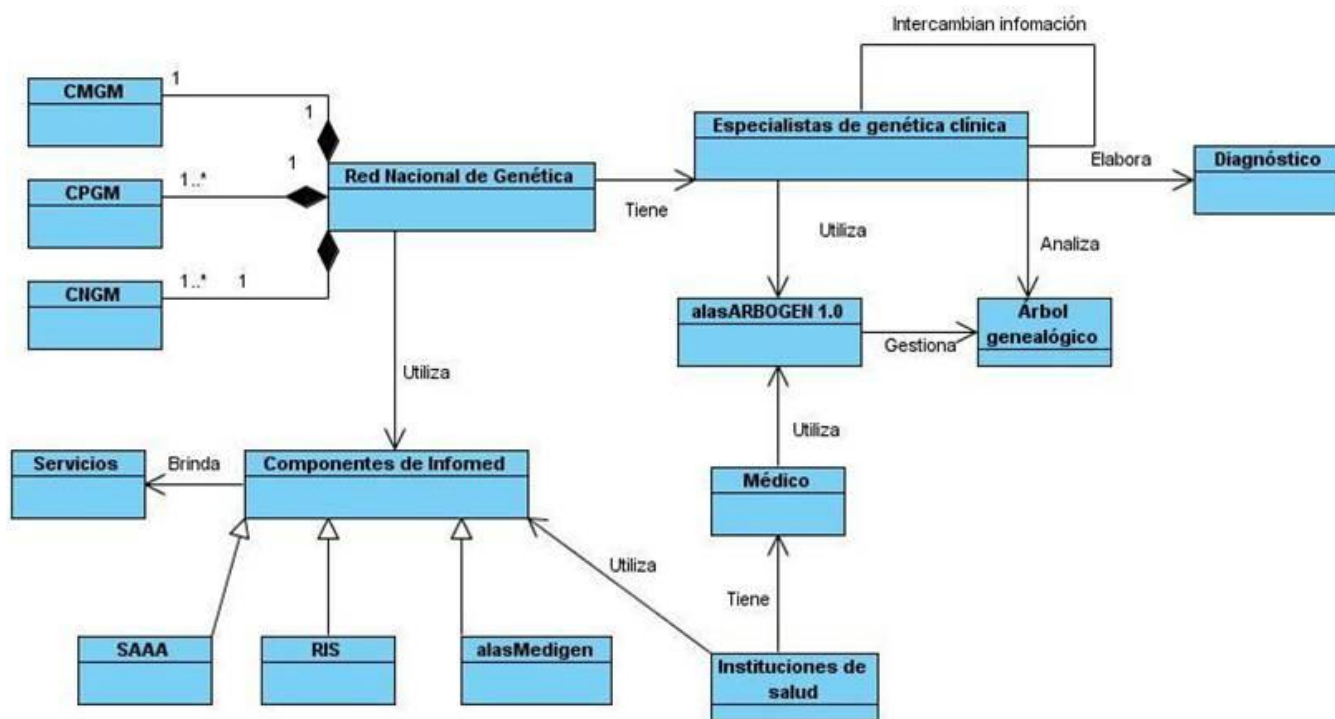


Fig. 1: Modelo de dominio.

2.4 ESPECIFICACIÓN DE LOS REQUISITOS DE LA APLICACIÓN.

2.4.1 REQUERIMIENTOS FUNCIONALES.

RF1. Gestionar Gráficamente un Individuo

- RF 1.1 Insertar un individuo.
- RF 1.2 Modificar datos de un individuo.
- RF 1.3 Eliminar un individuo.
- RF 1.4 Mostrar datos de un individuo.
- RF 1.5 Trasladar gráficamente un individuo.

RF2. Gestionar Relaciones entre Individuos.

- RF 2.1 Crear relación entre individuos.
- RF 2.2 Eliminar relación entre individuos.

RF3. Gestionar Símbolos.

- RF 3.1 Crear nuevo símbolo.
- RF 3.2 Modificar símbolos.
- RF 3.3 Eliminar un símbolo.

RF4. Gestionar Enfermedades.

- RF 4.1 Crear una nueva enfermedad.
- RF 4.2 Eliminar una enfermedad.

RF5. Gestionar Muestras.

- RF 5.1 Insertar nuevas muestras.
- RF 5.2 Modificar una muestra.
- RF 5.3 Eliminar una muestra.

RF6. Gestionar Familia.

- RF 6.1 Crear una nueva familia.
- RF 6.2 Buscar una familia.

RF 6.3 Actualizar datos de una familia.

RF 6.4 Guardar familia.

RF 6.5 Exportar familia.

RF7. Gestionar Dimensiones del Árbol Genealógico.

RF 7.1 Aumentar tamaño del árbol.

RF 7.2 Disminuir tamaño del árbol.

RF8. Buscar en el Árbol Genealógico.

RF9. Ordenar Árbol Genealógico.

RF10. Mostrar Vista Previa.

RF11. Imprimir Árbol Genealógico.

RF12. Generar Reporte del Árbol Genealógico.

2.4.2 REQUERIMIENTOS NO FUNCIONALES.

Los requerimientos no funcionales son cualidades o propiedades que el sistema debe tener, tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad y portabilidad.

✓ Apariencia e interfaz externa

El acceso a la aplicación y su interfaz de presentación será simple y más amigable, posibilitando que sea de fácil uso por el genetista. Tendrá un ambiente de trabajo similar independiente del sistema operativo en que se trabaje. Predominará en la aplicación el color azul. La interfaz externa debe estar diseñada para verse en cualquier resolución igual o superior a 1024x768. Se mostrará en pantalla una vista de datos tabular que permite la edición de la información de los individuos. Se presentan opciones del individuo más flexible para el cliente.

✓ Usabilidad

El acceso al sistema se realizará de forma fácil y rápida. El mismo contará con menús que satisfaga las necesidades de los usuarios. Este puede ser usado por cualquier persona que posea conocimientos básicos en el manejo de una computadora.

✓ **Rendimiento**

La velocidad de procesamiento de la aplicación debe ser rápida, al igual que la capacidad de respuesta.

✓ **Portabilidad.**

El sistema debe permitir ejecutarse en cualquiera de los sistemas operativos más usados en la actualidad. Destacándose el sistema operativo GNU Linux.

✓ **Soporte**

Las necesidades de los usuarios deben quedar satisfechas después de la puesta en explotación del sistema, a través de actualizaciones y mejoras. Para lograr esto se elaborarán una serie de manuales y videos tutoriales, así como la atención en línea.

✓ **Seguridad**

La información correspondiente a la asociación de los individuos y el parentesco puede ser considerada como confidencial, según las particularidades de las familias que se estudian a través de la aplicación. (Esta proporciona información de los familiares y sus descendencias). Debido a que no pueden efectuarse cambios sobre la información por personas no autorizadas, debe diseñarse un mecanismo de seguridad estricto que regule el acceso al sistema. Este debe manejarse de manera centralizada e independiente a cualquier sistema, y sus implicados deben estar bien identificados y comprometidos con el resguardo de la información.

La seguridad será manejada por el SAAA, a través de servicios web. Además, el sistema debe tener un mecanismo propio para gestionar la seguridad a través de niveles de acceso establecidos por los administradores del sistema.

✓ **Software**

Se requiere para el funcionamiento de la aplicación disponer del un sistema operativo que tenga la Máquina Virtual de Java 5.0 o versiones superiores.

✓ **Disponibilidad**

El sistema debe estar disponible a tiempo completo, y recuperarse rápidamente ante cualquier tipo de fallo. Deben ser creadas copias de respaldo de forma periódica, de manera tal que el sistema pueda restaurarse en caso de fallo crítico o pérdida de información. El sistema debe ser capaz de funcionar, independientemente de que los servicios de los diferentes componentes de SISalud no estén disponibles.

✓ Hardware

Para el desarrollo y ejecución del sistema se precisó que los requisitos estuvieran en función de los requerimientos que propone J2EE para el desarrollo de aplicaciones Java:

- Conexión al servidor a través de un modem o tarjeta de red.
- Procesador Pentium 3 o superior.
- 256 MB RAM como mínimo (Recomendado 512 MB RAM o superior).
- 500 MB de espacio libre en disco (Recomendado 800MB de espacio libre en disco).
- Se hace necesario contar con una impresora para la impresión de los reportes.

✓ Requisitos legales

Las tecnologías y herramientas que se utilicen para desarrollar el sistema deben estar bajo la licencia de software libre.

✓ Persistencia

La información generada por el sistema debe ser almacenada en bases de datos de forma permanente, con el objetivo de poder generar reportes y realizar estudios posteriores. Además la información persistirá en ficheros, en el caso de que la base de datos no se encuentre disponible en un momento determinado.

✓ Restricciones de diseño e implementación

El despliegue debe realizarse a través de una aplicación web, para lo que se utilizará la tecnología Java Web Start. La aplicación debe posibilitar trabajar en un ambiente conectado-desconectado.

2.5 DEFINICIÓN DE LOS CASOS DE USO DEL SISTEMA

2.5.1 ACTORES DEL SISTEMA

Actor	Descripción
Genetista	El especialista en genética es el que interactúa con el sistema y el único encargado de realizar el árbol genealógico.

2.5.2 DIAGRAMA DE CASOS DE USO DEL SISTEMA.

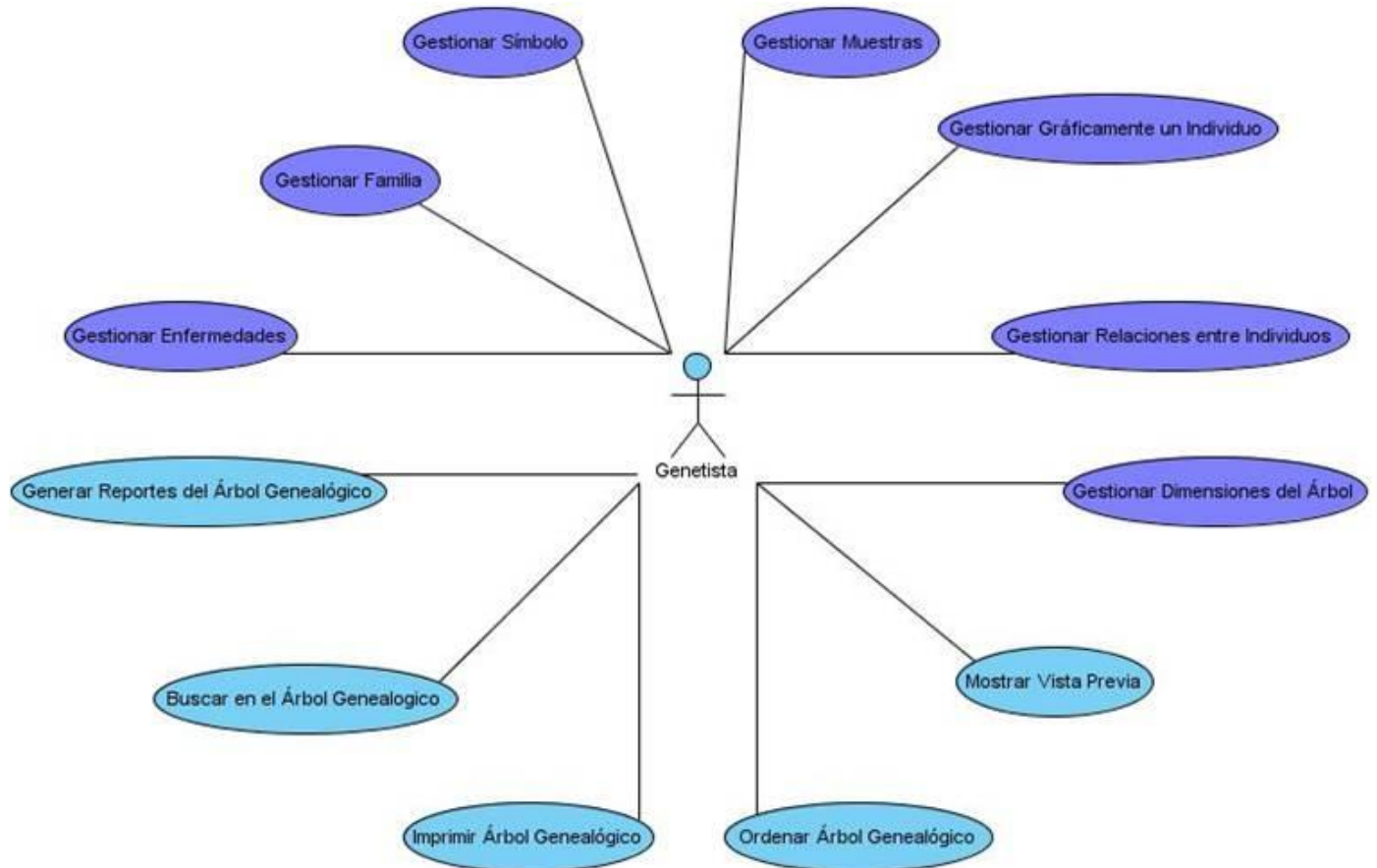


Fig. 2: Modelo de Objetos del Negocio.

2.5.3 DESCRIPCIÓN DE LOS CASOS DE USO DEL SISTEMA.

Los casos de uso del sistema Gestionar Familia, Gestionar Símbolos, Gestionar Enfermedades, Gestionar Dimensiones del Árbol, Gestionar Muestras, Gestionar Gráficamente un Individuo y Gestionar Relaciones entre Individuos son los casos de uso del sistema descritos en la aplicación de escritorio alasARBOGEN 1.0 [19]. En el caso de Gestionar Muestras, Gestionar Símbolos y Gestionar Enfermedades se le introdujeron mejoras que no afectan circunstancialmente las descripciones realizadas de los casos de uso. Es por ello que solamente se procede a realizar las descripciones de los casos de uso Imprimir Árbol Genealógico, Ordenar el Árbol Genealógico, Mostrar Vista previa del Árbol Genealógico, Buscar en el Árbol Genealógico y Generar Reporte del Árbol Genealógico.

2.5.3.1 BUSCAR EN EL ÁRBOL GENEALÓGICO.

Caso de Uso	Buscar en el Árbol Genealógico.
Actores	Genetista.
Resumen	El caso de uso comienza cuando el genetista realiza una búsqueda en el árbol genealógico. El sistema muestra una ventana de diálogo, para que el genetista seleccione el criterio de búsqueda por el que desea buscar en el árbol genealógico. El caso de uso termina cuando se efectúa la búsqueda correspondiente al criterio o el genetista decida cancelar la búsqueda.
Precondiciones	El sistema tiene que estar iniciado y el árbol genealógico debe haber sido creado con anterioridad.
Referencia	R8.
Prioridad	Normal
Flujo Normal de Eventos	
Actores del Sistema	Respuesta del Sistema
1. El genetista da clic en el menú "Edición" de la barra de menú.	2. El sistema muestra todas las opciones disponibles del menú "Edición". <ul style="list-style-type: none"> ➤ Aumentar ➤ Disminuir ➤ Ordenar Árbol ➤ Buscar en el Árbol Genealógico. ➤ Seleccionar Todo ➤ Eliminar
3. El genetista da clic en la opción "Buscar en el Árbol Genealógico" del menú "Edición".	4. El sistema muestra una ventana de diálogo que tiene los siguientes criterios de búsqueda: <ul style="list-style-type: none"> ➤ Nombre. ➤ Apellidos. ➤ CI.

	<ul style="list-style-type: none"> ➤ Área_salud ➤ Intento_suicidio ➤ Enfermedades_genéticas.
5. El genetista selecciona el criterio de búsqueda que va a utilizar.	<p>6. El sistema en dependencia de la operación que se le solicita hace lo siguiente:</p> <ul style="list-style-type: none"> ➤ Si se decide buscar por el criterio de búsqueda “Intento de suicidio”, ir a la sección “Buscar Intentos de suicidio”. ➤ Si se decide buscar por el criterio de búsqueda “Nombre”, ir a la sección “Buscar Nombre”. ➤ Si se decide buscar por el criterio de búsqueda “Enfermedades genéticas”, ir a la sección “Buscar Enfermedades genéticas”. ➤ Si se decide buscar por el criterio de búsqueda “Apellidos”, ir a la sección “Buscar Apellidos”. ➤ Si se decide buscar por el criterio de búsqueda “CI”, ir a la sección “Buscar CI”. ➤ Si se decide buscar por el criterio de búsqueda “Unidad de salud”, ir a la sección “Buscar Unidad de salud”.
Sección “Buscar Intentos de suicidio”	
1. El genetista selecciona la opción “Intento de suicidio”.	2. El sistema muestra una tabla en la ventana de diálogo “ <i>MuestraBúsqueda</i> ”, donde visualiza las personas encontradas que tuvieron intentos de suicidio.
3. El genetista revisa los datos, posteriormente da clic en el botón “Cerrar”.	4. El sistema cierra la ventana de diálogo “ <i>MostrarBúsqueda</i> ” y finaliza el caso de uso.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Flujos Alternos Sección “Buscar Intentos de Suicidio”	
	2.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
3.1 El genetista da clic en el botón “Cerrar.	4.1 El sistema cierra la ventana de diálogo “ <i>MuestraBúsqueda</i> ” y termina el caso de uso.
Sección “Buscar Nombre”	
1. El genetista selecciona la opción “Nombre”.	2. El sistema muestra una caja de texto para que el genetista introduzca el nombre de la persona que desee buscar.
3. Una vez que el genetista termina de introducir el nombre de la persona que desea buscar, da clic en el botón “Buscar”, para efectuar la búsqueda.	4. El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se muestra una tabla en la ventana de diálogo “ <i>MuestraBúsqueda</i> ”, donde se visualiza la información deseada por el genetista.
5. El genetista revisa los datos, posteriormente da clic en el botón “Cerrar”.	6. El sistema cierra la venta de diálogo “ <i>MuestraBúsqueda</i> ” y termina el caso de uso.
Flujo Alternativo Sección “Buscar Nombre”	
Actores del Sistema	Respuesta del Sistema
	4.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
5.1 El genetista da clic en el botón “Cerrar.	6.1 El sistema cierra la ventana de diálogo “ <i>MuestraBúsqueda</i> ” y termina el caso de uso.
Sección “Buscar Enfermedades genéticas”	
Actores del Sistema	Respuesta del Sistema
1. El genetista selecciona la opción	2. El sistema muestra en la ventana de diálogo

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

“Enfermedades genéticas”.	“ <i>MuestraBusqueda</i> ” una lista que contiene los tipos de enfermedades genéticas, que puede tener una persona en el árbol genealógico.
3. El genetista selecciona un tipo de enfermedad genética y da clic en el botón buscar.	4. El sistema muestra en la ventana de diálogo “ <i>MuestraBusqueda</i> ” una tabla que contiene las personas del árbol genealógico, que tienen esa enfermedad especificada.
5. El genetista analiza los datos encontrados por el sistema y da clic en el botón “Cerrar”.	6. El sistema cierra la ventana “ <i>MuestraBusqueda</i> ” y finaliza el caso de uso.
Flujo Alterno Sección “Buscar Enfermedades genéticas”	
Actores del Sistema	Respuesta del Sistema
	4.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
5.1 El genetista da clic en el botón “Cerrar.”	6.1 El sistema cierra la ventana de diálogo “ <i>MuestraBusqueda</i> ” y termina el caso de uso.
Sección “Buscar Apellidos”	
Actores del Sistema	Respuesta del Sistema
1. El genetista selecciona la opción “Apellidos”.	2. El sistema muestra una caja de texto para que el genetista introduzca los apellidos de la persona que desee buscar.
3. Una vez que el genetista termina de introducir los apellidos de la persona que desea buscar, da clic en el botón “Buscar”, para efectuar la búsqueda.	4. El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se muestra una tabla en la ventana de diálogo “ <i>MuestraBusqueda</i> ”, donde se visualiza la información deseada por el genetista.
5. El genetista revisa los datos, posteriormente da clic en el botón	6. El sistema cierra la venta de diálogo “ <i>MuestraBusqueda</i> ” y termina el caso de uso.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

“Cerrar”.	
Flujo Alternativo Sección “Buscar Apellido”	
Actores del Sistema	Respuesta del Sistema
	4.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
5.1 El genetista da clic en el botón “Cerrar”.	6.1 El sistema cierra la ventana de diálogo “ <i>MuestraBúsqueda</i> ” y termina el caso de uso.
Sección “Buscar CI”	
Actores del Sistema	Respuesta del Sistema
1. El genetista selecciona la opción “CI”.	2. El sistema muestra una caja de texto para que el genetista introduzca el carnet de identidad de la persona que desee buscar.
3. Una vez que el genetista termina de introducir el número de carnet de identidad de la persona que desea buscar, da clic en el botón “Buscar”, para efectuar la búsqueda.	4. El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se muestra una tabla en la ventana de diálogo “ <i>MuestraBúsqueda</i> ”, donde se visualiza la información deseada por el genetista.
5. El genetista revisa los datos, posteriormente da clic en el botón “Cerrar”.	6. El sistema cierra la ventana de diálogo “ <i>MuestraBúsqueda</i> ” y termina el caso de uso.
Flujo Alternativo Sección “Buscar CI”	
Actores del Sistema	Respuesta del Sistema
	4.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
5.1 El genetista da clic en el botón “Cerrar”.	6.1 El sistema cierra la ventana de diálogo “ <i>MuestraBúsqueda</i> ” y termina el caso de uso.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Sección “Buscar Área de Salud”	
Actores del Sistema	Respuesta del Sistema
1. El genetista selecciona la opción “Área_Salud”.	2. El sistema muestra en la ventana de diálogo “ <i>MuestraBusqueda</i> ” una lista que contiene las áreas de salud disponibles en la base de datos.
3. El genetista selecciona una unidad de salud y da clic en el botón buscar.	4. El sistema muestra en la ventana de diálogo “ <i>MuestraBusqueda</i> ” una tabla que contiene las personas del árbol genealógico, que pertenecen a esa área de salud.
5. El genetista analiza los datos encontrados por el sistema y da clic en el botón “Cerrar”.	6. El sistema cierra la ventana “ <i>MuestraBusqueda</i> ” y finaliza el caso de uso.
Flujo Alternativo Sección “Buscar Área de Salud”	
Actores del Sistema	Respuesta del Sistema
	4.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
5.1 El genetista da clic en el botón “Cerrar”.	6.1 El sistema cierra la ventana de diálogo “ <i>MuestraBusqueda</i> ” y termina el caso de uso.
Prototipo de Interfaz	Ver Anexo 1.

2.5.3.2 ORDENAR ÁRBOL GENEALÓGICO.

Caso de Uso	Ordenar Árbol Genealógico
Actores	Genetista
Resumen	El caso de uso se inicia cuando el genetista da clic en la opción ordenar árbol. El caso de uso finaliza una vez que se ordena el árbol genealógico.
Precondiciones	Debe estar iniciada la aplicación. Para ordenar el árbol genealógico, debe estar creado.
Referencia	R9.
Prioridad	Normal
Flujo Normal de Eventos	
Actores del Sistema	Respuesta del Sistema
1. El genetista da clic en el menú “Edición” de la barra de menú.	2. El sistema muestra todas las opciones disponibles del menú “Edición”. <ul style="list-style-type: none"> ➤ Aumentar ➤ Disminuir ➤ Ordenar árbol ➤ Buscar en el Árbol Genealógico. ➤ Seleccionar Todo ➤ Eliminar
3. El genetista da clic en la opción “Ordenar árbol” del menú “Edición”.	4. El sistema ordena el árbol atendiendo a la jerarquía existente.
Prototipo de Interfaz	Ver Anexo 2.
Poscondiciones	El árbol genealógico queda ordenado jerárquicamente.

2.5.3.3 MOSTRAR VISTA PREVIA DEL ÁRBOL GENEALÓGICO.

Caso de Uso	Mostrar Vista Previa del árbol Genealógico.
Actores	Genetista
Resumen	El caso de uso se inicia cuando el genetista escoge la opción vista previa, se muestra una nueva ventana que contiene el árbol genealógico en la forma que se va a imprimir. El caso de uso termina cuando el genetista cierra la vista previa.
Precondiciones	La aplicación debe estar iniciada. Para mostrar la vista previa debe estar creado el árbol genealógico.
Referencia	R10
Prioridad	Normal
Flujo Normal de Eventos	
Actores del Sistema	Respuesta del Sistema
1. El genetista da clic en el menú "Archivo".	2. El sistema muestra la lista de opciones del menú "Archivo". <ul style="list-style-type: none"> ➤ Nuevo ➤ Abrir ➤ Guardar ➤ Guardar Como ➤ Imprimir ➤ Exportar ➤ Salir
3. El genetista da clic en el menú "Imprimir".	4. El sistema muestra los submenú pertenecientes al menú "Imprimir" <ul style="list-style-type: none"> ➤ Imprimir ➤ Vista Previa ➤ Vista Tabular
5. El genetista da clic en el menú "Vista Previa".	6. El sistema muestra una ventana de diálogo que cuenta con una vista previa de cómo se

	imprimirá el árbol genealógico.
7. El genetista cierra la ventana de diálogo y finaliza el caso de uso.	
Prototipo de Interfaz	Ver Anexo 3.

2.5.3.4 IMPRIMIR ÁRBOL GENEALÓGICO.

Caso de Uso	Imprimir árbol Genealógico
Actores	Genetista
Resumen	El caso de uso se inicia cuando el genetista va a imprimir el árbol genealógico de la persona. El sistema le muestra la ventana de diálogo correspondiente a la opción. El caso de uso finaliza una vez que se imprima el árbol genealógico, o el genetista decida cancelar la opción.
Precondiciones	Debe estar iniciada la aplicación. Para imprimir el árbol genealógico, debe estar creado.
Referencia	R11.
Prioridad	Normal
Flujo Normal de Eventos	
Actores del Sistema	Respuesta del Sistema
1. El genetista da clic en el menú "Archivo".	2. El sistema muestra la lista de opciones del menú "Archivo". <ul style="list-style-type: none"> ➤ Nuevo ➤ Abrir ➤ Guardar ➤ Guardar Como ➤ Imprimir ➤ Exportar ➤ Salir

3. El genetista da clic en el menú “Imprimir”.	4. El sistema muestra los submenú pertenecientes al menú “Imprimir” <ul style="list-style-type: none"> ➤ Imprimir ➤ Vista Previa ➤ Vista Tabular
5. El genetista da clic en el menú de impresión que desea.	6. El sistema en dependencia de la opción que se seleccione va a las siguientes secciones. <ul style="list-style-type: none"> ➤ En caso que el genetista seleccione la opción “Imprimir”, el sistema va a la sección “Imprimir”. ➤ En caso que el genetista seleccione la opción “Vista Previa”, el sistema va a la sección “Vista Previa”. ➤ En caso que el genetista seleccione la opción “Vista Tabular”, el sistema va a la sección “Vista Tabular”.

Sección “Imprimir”

Actores del Sistema	Respuesta del Sistema
	1. El sistema muestra una ventana de diálogo para la impresión.
2. El genetista da clic en el botón “Imprimir” para realizar la impresión del árbol genealógico.	3. El sistema imprime el árbol genealógico y termina el caso de uso.

Flujo Alternativo de Eventos Sección “Imprimir”

Actores del Sistema	Respuesta del Sistema
2.1 El genetista da clic en el botón “Cancelar” de la ventana de diálogos.	3.1 El sistema cierra la ventana de diálogos.

Sección “Vista Previa”

Actores del Sistema	Respuesta del Sistema
	1. El sistema muestra la ventana de diálogo

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	“Vista Previa”.
2. El genetista da clic en botón “Imprimir”.	3. El sistema muestra una ventana de diálogo para la impresión.
4. El genetista da clic en el botón “Imprimir” para realizar la impresión del árbol genealógico.	5. El sistema imprime el árbol genealógico y termina el caso de uso.
Flujo Alternativo de Eventos Sección “Vista Previa”	
Actores del Sistema	Respuesta del Sistema
2.1 El genetista da clic en el botón “Cancelar” de la ventana de diálogos.	3.1 El sistema cierra la ventana de diálogos.
Sección “Vista Tabular”	
Actores del Sistema	Respuesta del Sistema
	1. El sistema muestra una ventana de diálogo para la impresión.
2. El genetista da clic en el botón “Imprimir” para realizar la impresión de los datos representados en la vista tabular, pertenecientes a las personas del árbol genealógico.	3. El sistema imprime los datos de las personas del árbol genealógico y termina el caso de uso.
Flujo Alternativo de Eventos Sección “Vista Tabular”	
Actores del Sistema	Respuesta del Sistema
2.1 El genetista da clic en el botón “Cancelar” de la ventana de diálogos.	3.1 El sistema cierra la ventana de diálogos.
Prototipo de Interfaz	Ver Anexo 4.

2.5.3.5 GENERAR REPORTE DEL ÁRBOL GENEALÓGICO.

Caso de Uso	Generar Reporte del árbol Genealógico.
Actores	Genetista
Resumen	El caso de uso se inicia cuando el genetista procede a generar un reporte del árbol genealógico. El sistema le muestra una ventana de diálogo para que seleccione el directorio en el que desea guardar el reporte generado. El caso de uso termina cuando se guarde el reporte generado, o el genetista seleccione la opción cancelar.
Precondiciones	Debe estar iniciada la aplicación. Para generar reporte del árbol genealógico, este debe estar creado.
Referencia	R12.
Prioridad	Normal
Flujo Normal de Eventos	
Actores del Sistema	Respuesta del Sistema
1. El genetista da clic en el menú "Archivo".	2. El sistema muestra la lista de opciones del menú "Archivo". <ul style="list-style-type: none"> ➤ Nuevo ➤ Abrir ➤ Guardar ➤ Guardar Como ➤ Imprimir ➤ Exportar ➤ Salir
3. El genetista da clic en el menú "Exportar".	4. El sistema muestra los submenú pertenecientes al menú "Exportar" <ul style="list-style-type: none"> ➤ Reporte del árbol ➤ Imagen del árbol

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

5. El genetista da clic en el menú “Reporte del árbol”.	6. El sistema muestra una ventana de diálogo donde pide un directorio para guardar el reporte.
7. El genetista selecciona el directorio donde desea guardar el reporte generado por la aplicación y da clic en el botón aceptar.	8. El sistema genera el reporte en el directorio seleccionado y finaliza el caso de uso.
Flujo Alternativo de Eventos	
Actores del Sistema	Respuesta del Sistema
7.1 El genetista da clic en el botón cancelar de la ventana de diálogo y termina el caso de uso.	
Prototipo de Interfaz	Ver Anexo 5.

CONCLUSIONES

En este capítulo se definió detalladamente el modelo de dominio y las necesidades de los estudios genéticos en la actualidad. Se identificaron los requerimientos funcionales y no funcionales del sistema, así como los actores y casos de uso del sistema. Se realizó el diagrama de casos de uso del sistema, quedando reflejadas las relaciones existentes entre los actores y los casos de uso. Se describieron los casos de uso y se diseñaron los prototipos de interfaz de los casos de uso descritos.

CAPÍTULO 3: DISEÑO DEL SISTEMA

En este capítulo se le da cumplimiento a los requisitos a través del diseño de la aplicación, describiendo cómo implementar, enfocado a como el sistema cumple los objetivos planteados en los requisitos funcionales y no funcionales. Se explica la arquitectura utilizada así como los principales patrones y se realizan los diagramas de clases y los diagramas de interacción, según los casos de uso más relevantes definidos en el capítulo anterior.

3.1 ESTILO ARQUITECTÓNICO UTILIZADO.

Un estilo arquitectónico define familias de sistemas en términos de un patrón de organización estructural. Define un vocabulario de tipos de componentes y conectores y un conjunto de restricciones de cómo cambiar esos componentes y conectores [20].

Para el desarrollo de la aplicación se utilizó como estilo arquitectónico el **Modelo Vista Controlador**.

3.1.1 MODELO VISTA CONTROLADOR (MVC)

En patrón Modelo-Vista-Controlador divide una aplicación en 3 áreas fundamentales: entrada, procesamiento y salida, utilizando para ello la siguiente abstracción [21].

Modelo: Es el objeto que representa los datos del programa. Encapsula los datos y las funcionalidades. El modelo es independiente de cualquier representación de salida y comportamiento de entrada.

Vista: Muestra la información al usuario. Pueden existir múltiples vistas del modelo. Cada vista tiene asociado un componente controlador.

Controlador: Reciben las entradas, usualmente como eventos. Los eventos son traducidos a solicitudes de servicio para el modelo o la vista.

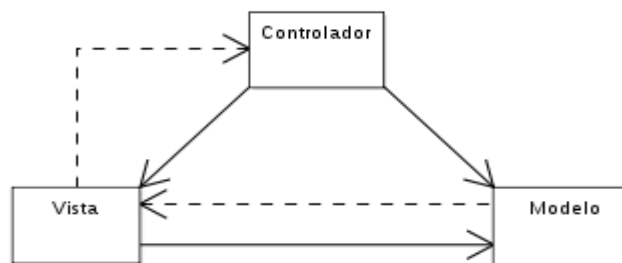


Fig. 3: Diagrama Modelo Vista Controlador.

3.2 PATRONES DE DISEÑO UTILIZADOS.

Los patrones de diseño son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objeto. Son puntos de partida para la solución de problemas que se repiten con frecuencia en el desarrollo de software.

3.2.1 PATRÓN STATE

Es un patrón de comportamiento que permite a un objeto modificar su comportamiento cada vez que cambia su estado interno.

Problema: Una instancia de la clase persona de tipo *Desconocido* puede cambiar su estado atendiendo al sexo.

Solución: En la solución se aplica el patrón Gang of Four GOF de comportamiento, a nivel de objeto los definimos como STATE.

Aplicación: Una instancia de la clase persona actúa en dependencia de su sexo, debido a esto, están definidas tres tipos de persona según su sexo; Hombre, Mujer y Desconocido. La persona cuenta con atributo de tipo sexo, y existe un oyente de evento en la clase "*FamiliaVisual*", el cual es el encargado de recibir la información de cuando ocurre un evento de cambio de sexo. En el caso de la persona de tipo *Desconocido*, se crea una instancia nueva de la clase persona de tipo del sexo que se especificó, conservando el valor de los atributos. Garantizamos así que atendiendo al cambio de estado en el sexo, se comporte de acuerdo con el tipo de persona que es.

3.2.2 PATRÓN SINGLETON

Es un patrón de creación dentro de los patrones GOF del que hace referencia a las instancias únicas de los objetos.

Problema: Se admiten exactamente una instancia de una clase. Los objetos necesitan un punto de acceso global.

Solución: Se define un método estático en la clase que devuelva el SINGLETON.

Aplicación: Para dar solución a este problema se crean métodos estáticos en las clases "*Conexión*" y "*OperacionesDatos*" a la hora de obtener las instancias para acceder a la base datos. En el caso de que se requiera hacer una descarga de algún nomenclador de la base de datos hacia un fichero binario local,

se hacen llamadas a los métodos de forma estática de dichas clases, permitiendo esto que de cualquier parte de la aplicación se pueda acceder al método haciendo una instanciación única de los objetos.

3.2.3 PATRONES GRASP

3.2.3.1 PATRÓN EXPERTO

Es un patrón que expresa la intuición de que los objetos hacen las cosas según la información que tiene, reforzando en el encapsulamiento.

Problema: Que la responsabilidad de las instancias de una clase sea acorde a información con que cuenta la clase.

Solución: Asignar la responsabilidad a las clases que cuenta con la información necesaria para cumplir con tal responsabilidad.

Aplicación: Es necesario que por cada cambio que se realiza en los datos en tiempo de ejecución, se repinten los controladores de forma total. La responsabilidad es asignada a la clase “*FamiliaVisual*”, que a pesar de ser parte de la capa Vista, es la encargada de pintar el árbol. Esta clase tiene todos los controles que representan a cada persona.

3.2.3.2 PATRÓN CREADOR

Es un patrón que asigna la responsabilidad de creación de otros objetos, a un objeto en específico.

Problema: Solo puede crear instancias de los objetos los encargados de registrarlos, que son los que lo van a contener.

Solución: Que las instancias de las clases que registran objetos, sean los encargados de crear estos objetos.

Aplicación: La clase “*FamiliaVisual*” es la contenedora de todas las instancias de “*PersonaVisual*” y “*Arbol*” es quien contiene la instancia de “*Familia*”, que a su vez es la que contiene todas las instancias de “*Persona*”. Cada persona que se representa en el árbol tiene una clase “*PersonaVisual*” que actúa como Vista y una clase “*Persona*”, que es la que pertenece al Modelo y es capaz de guardar los datos. A la hora de crear una persona, el evento es captado por “*FamiliaVisual*”, encargándose esta de crear a “*PersonaVisual*”, que es la que registra. Informa a la clase “*Arbol*”, que es la que tiene la clase “*Familia*” que a su vez es la que registra las personas, para que cree la instancia de “*Persona*” correspondiente a la “*PersonaVisual*” creada anteriormente.

3.3 DIAGRAMAS DE CLASES.

En el diagrama de clases del diseño se reflejan gráficamente las especificaciones de las clases e interfaces de la aplicación. Se evidencian además los diferentes tipos de relaciones que existen entre las clases involucradas en el desarrollo del sistema.

A continuación, describimos algunos aspectos fundamentales del diseño.

Los diagramas de clases del diseño están estructurados según lo que plantea el patrón MVC, el paquete “Visual”, representa la vista, el paquete “Controlador” representa como su nombre lo indica la parte Controladora del patrón y el paquete “Entidades” representa el modelo. El paquete “Utiles” contiene un grupo de clases que sirven de soporte a la implementación. Con el objetivo de ilustrar de forma más precisa el diagrama de clases del diseño por casos de uso, los diagramas de clases aparecen con las propiedades y métodos ocultos.

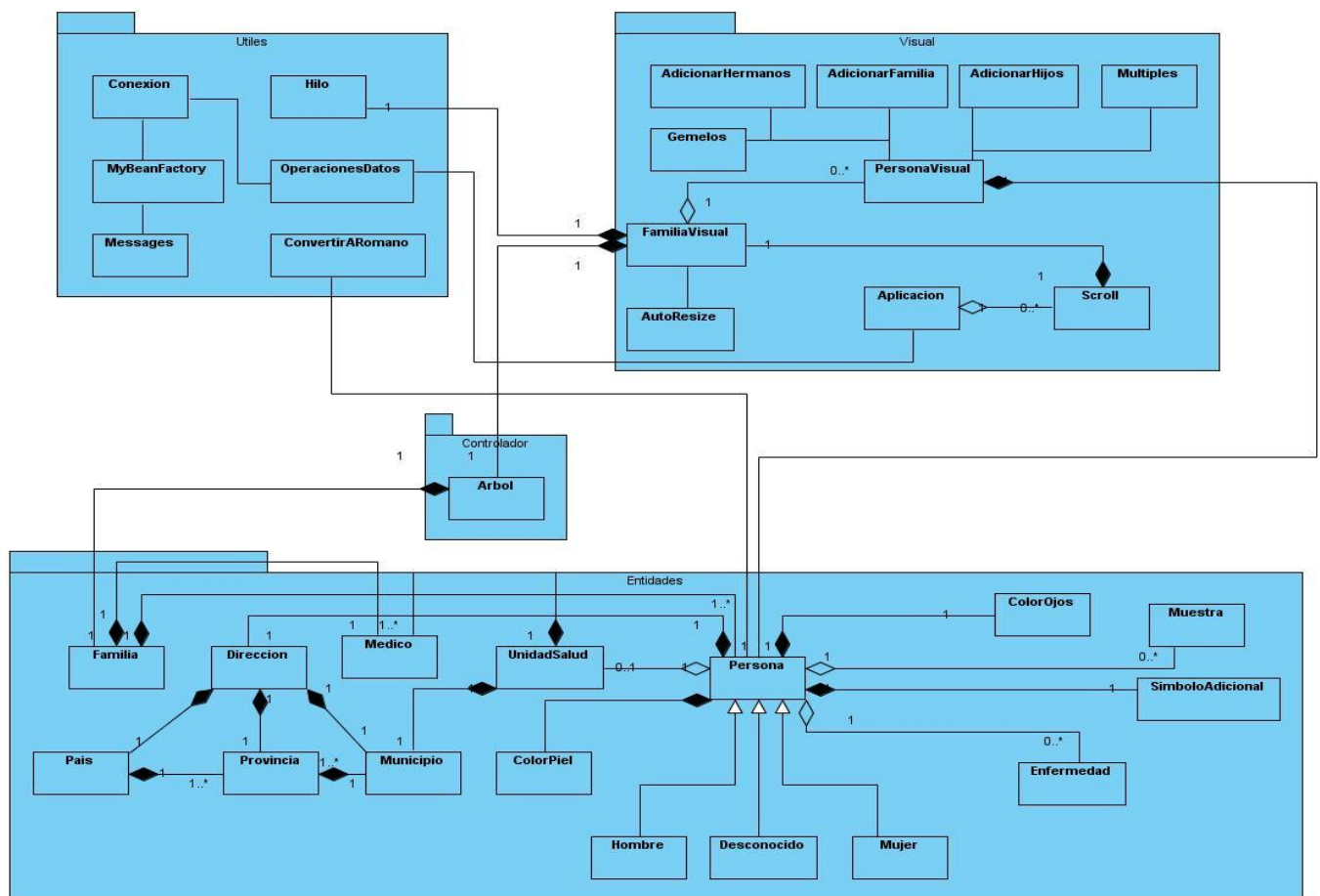


Fig. 4 Diagrama de clases del diseño: CU Gestionar Gráficamente un Individuo.

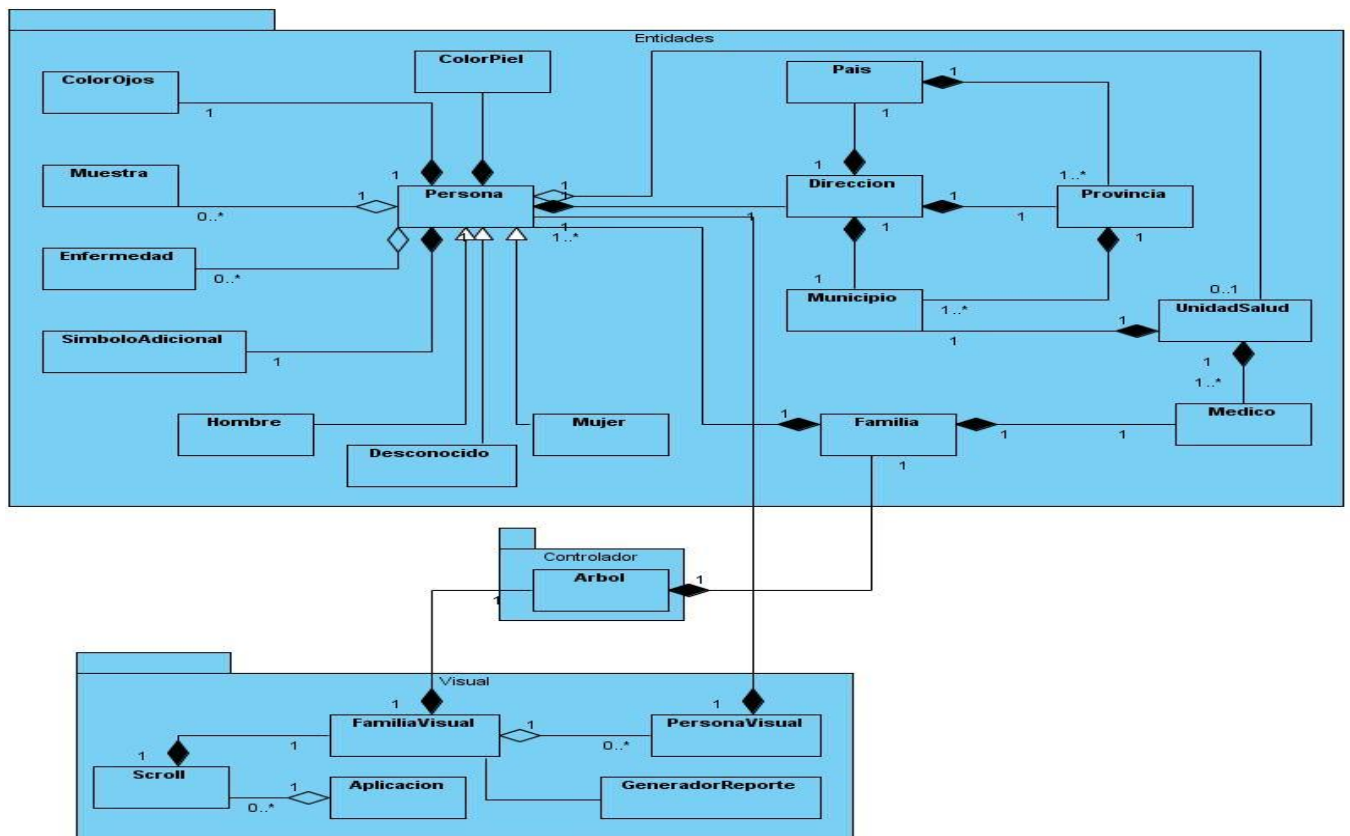


Fig. 5 Diagrama de clases del diseño: CU Generar Reporte del Árbol Genealógico.

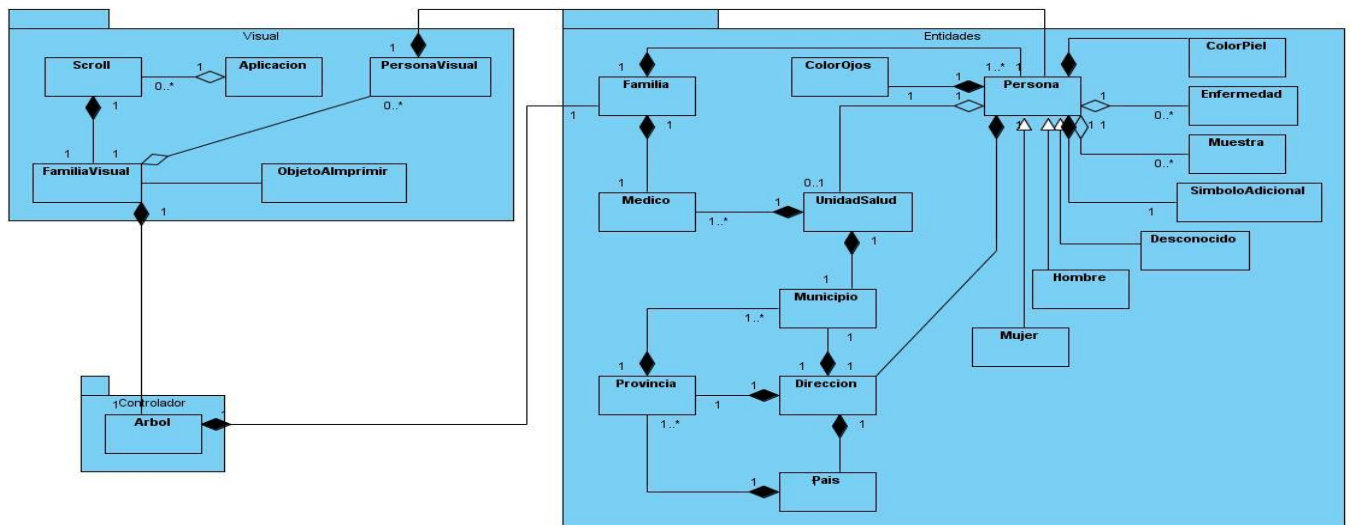


Fig. 6 Diagrama de clases del diseño: CU Imprimir Árbol Genealógico.

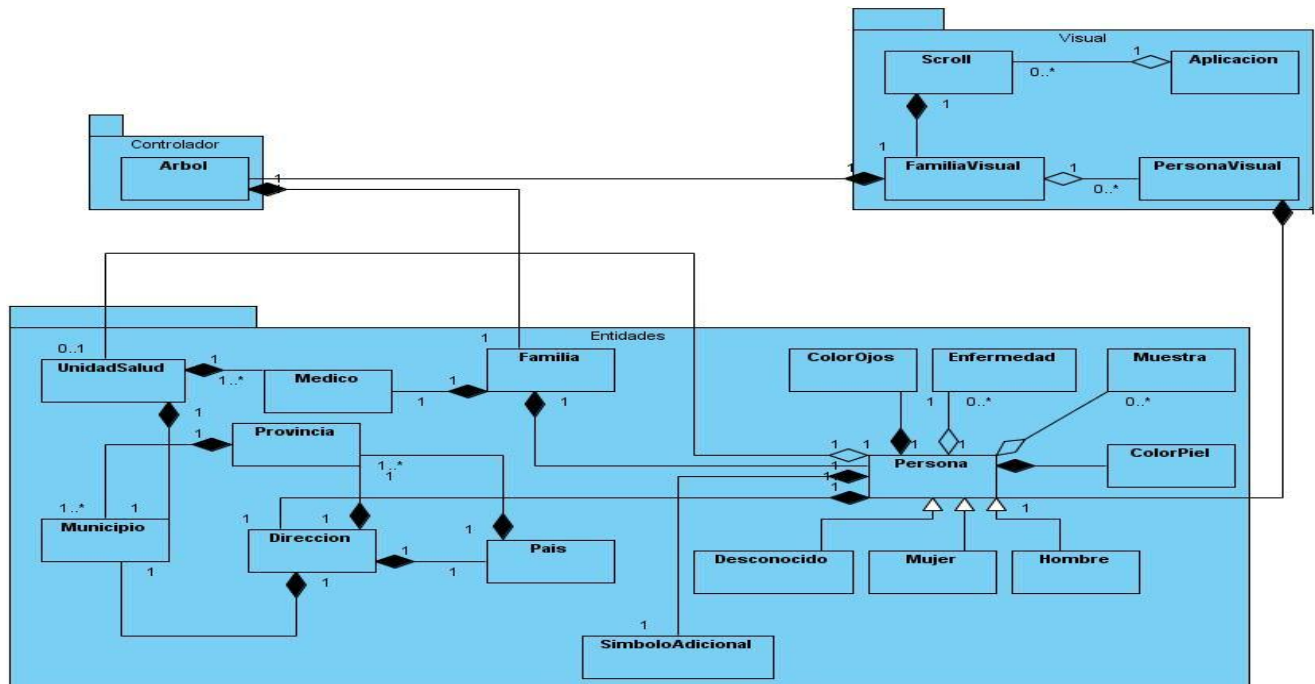


Fig. 7 Diagrama de clases del diseño: CU Ordenar Árbol Genealógico.

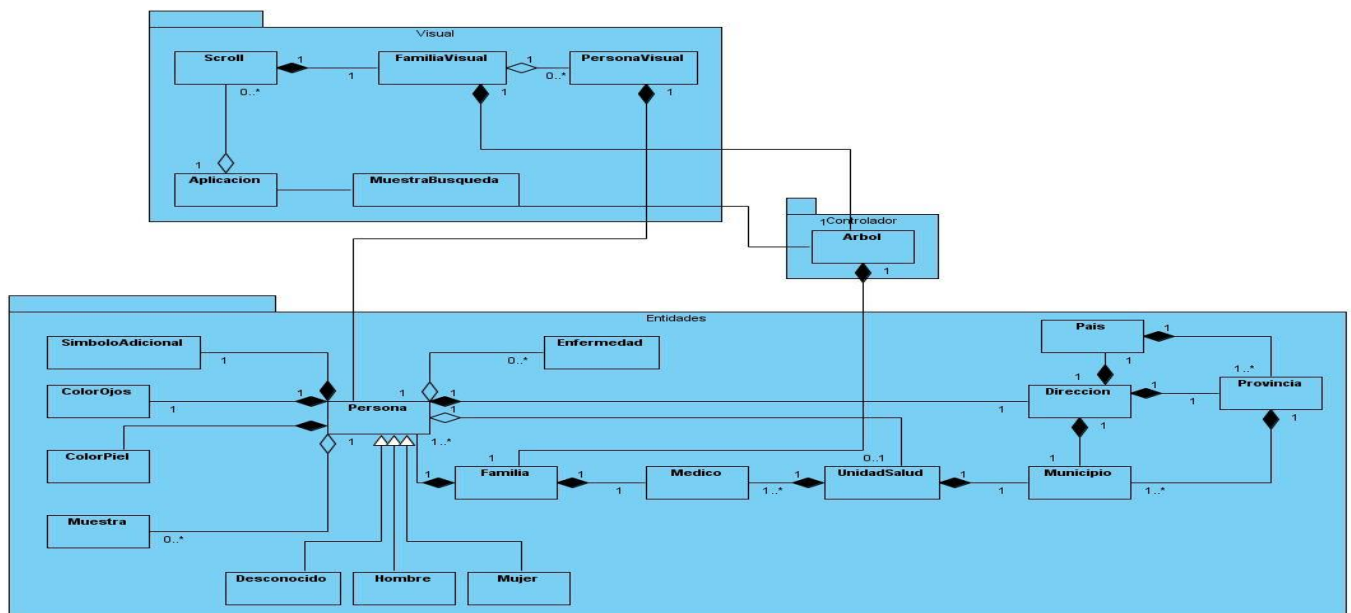


Fig. 8 Diagrama de clases del diseño: CU Buscar en el Árbol Genealógico.

3.4 DIAGRAMAS DE SECUENCIAS.

Un diagrama de secuencia es usado para modelar la interacción entre objetos en un sistema ordenadas a través del tiempo. Muestra los objetos y las secuencias que intervienen en el escenario.

Se realizaron los diagramas de secuencia de los objetos de la aplicación, mostrándose la interacción entre ellos. A continuación se muestran los diagramas de secuencia que corresponden con los casos de uso que se describen con anterioridad.

La Fig. 9 representa la interacción entre las clases que intervienen en la Sección Insertar un Individuo del caso de uso Gestionar gráficamente un individuo. Se muestra una dependencia del tipo de sexo que existe a la hora de crear el individuo, haciendo posible que se creen las personas acorde al tipo de sexo del individuo. Por cada persona que se representa es necesario tener las clases “*PersonaVisual*” y “*Persona*”. La clase “*PersonaVisual*” representa la vista del objeto persona, la cual se adiciona en la clase “*FamiliaVisual*”. La clase “*Persona*” representa el modelo, y se adiciona en la clase “*Familia*”, denotando de esta forma que para cada persona que se cree, existe un objeto del modelo y un objeto de la vista que es el encargado de hacer su representación gráfica.

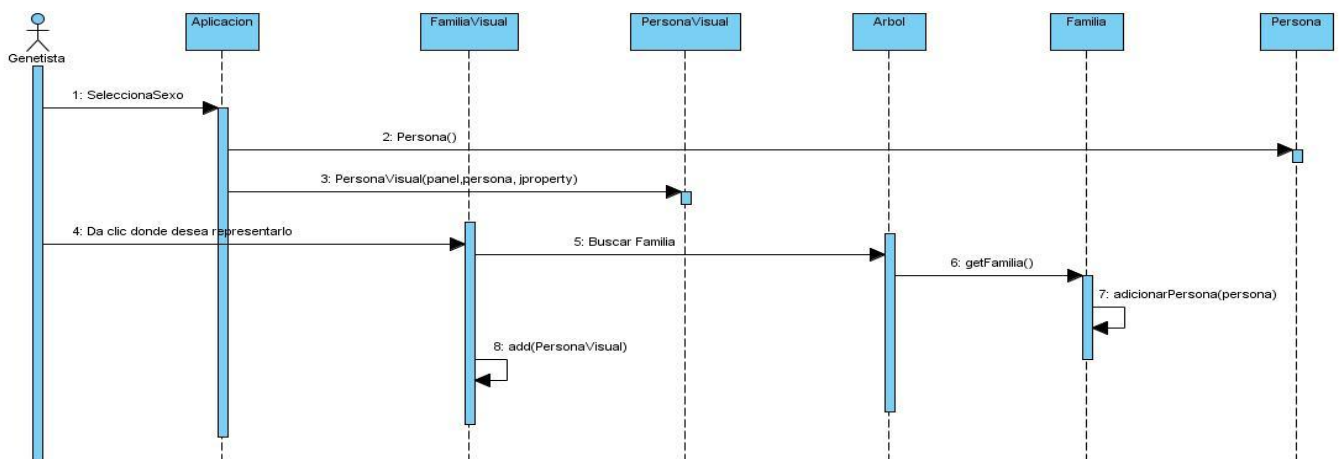


Fig. 9 Diagrama de secuencias del CU Gestionar gráficamente un individuo. Sección Insertar un Individuo.

La Fig. 10 representa la interacción entre las clases que intervienen en la Sección Eliminar un Individuo del caso de uso Gestionar gráficamente un individuo.

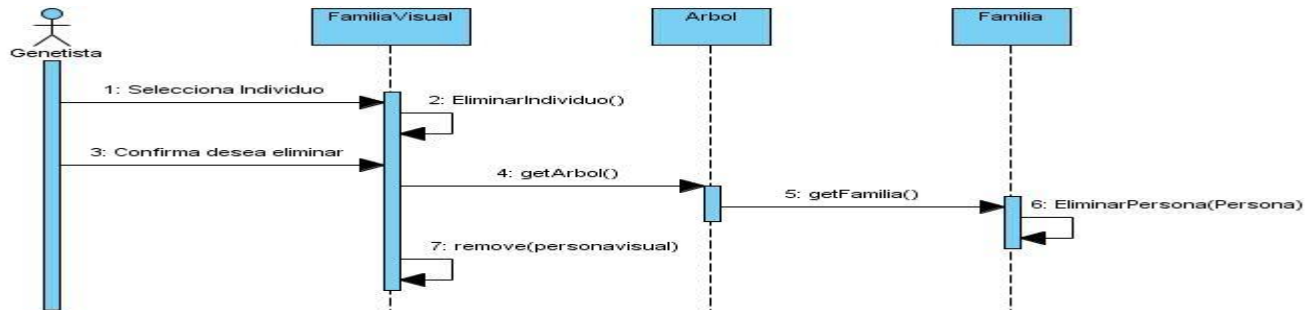


Fig. 10 Diagrama de secuencias del CU Gestionar gráficamente un individuo. Sección Eliminar un Individuo.

La Fig. 11 representa la interacción entre las clases que intervienen en el caso de uso Generar Reporte del Árbol Genealógico. Se muestra una relación entre la clase “Aplicacion” y “FamiliaVisual”, siendo en la primera donde se selecciona la opción de generar reporte, utilizándose los datos del árbol que están contenidos en “FamiliaVisual”. Posteriormente se hace una llamada a la clase “GenerarReporte”, que es la encargada de generar los reportes del árbol.

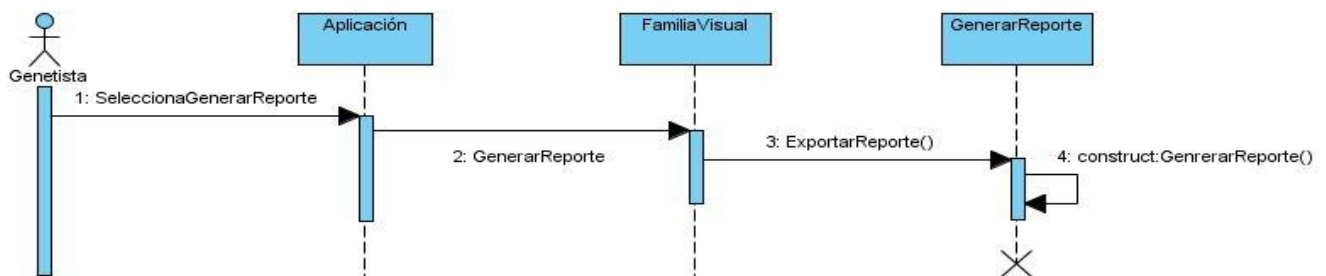


Fig. 11 Diagrama de secuencias del CU Generar Reporte del Árbol Genealógico.

La Fig. 12 representa la interacción entre las clases que intervienen en el caso de uso Imprimir Árbol Genealógico, en la sección Imprimir Vista Tabular. Se muestra una relación entre la clase “Aplicacion” y la clase “FamiliaVisual”, donde se hace una llamada a “FamiliaVisual” para efectuar la impresión de los datos de la tabla de la vista tabular.

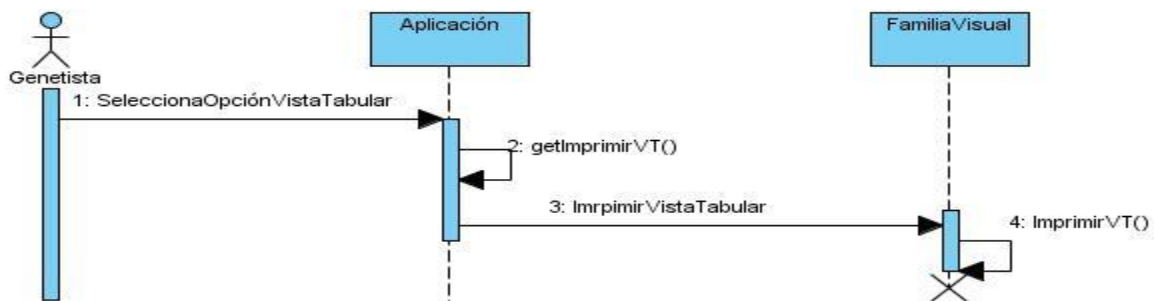


Fig. 13 Diagrama de secuencias del CU Imprimir Árbol Genealógico. Sección Imprimir Vista Tabular.

La Fig. 14 representa la relación entre las clases que intervienen en el caso de uso Imprimir Árbol Genealógico, en la sección Imprimir. Se muestra una relación entre la clase “*Aplicacion*” y la clase “*FamiliaVisual*”, donde se hace una llamada a “*FamiliaVisual*” para efectuar la impresión de la imagen del árbol genealógico representado.

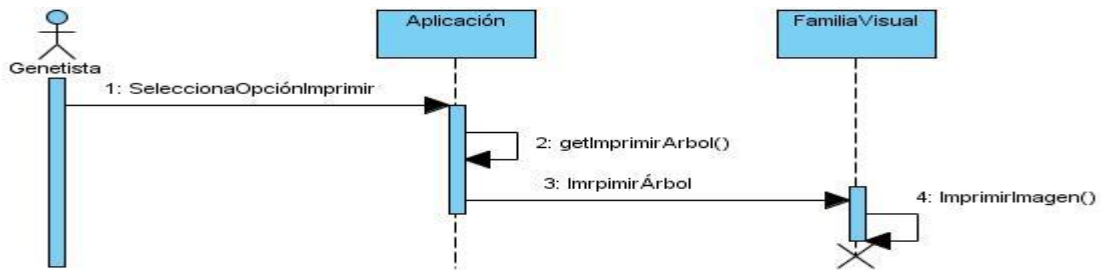


Fig. 14 Diagrama de secuencias del CU Imprimir Árbol Genealógico. Sección Imprimir.

La Fig. 15 representa la relación entre las clases que intervienen en el caso de uso Imprimir Árbol Genealógico, en la sección Imprimir Vista Previa. Se muestra una relación entre la clase “*Aplicacion*” y la clase “*FamiliaVisual*”, donde se hace una llamada a “*FamiliaVisual*” para efectuar la impresión de la imagen del árbol genealógico representado, a través de la vista previa de la aplicación. La clase “*AutoResize*” permite crear la imagen representada en “*FamiliaVisual*”, en dimensiones más pequeñas.

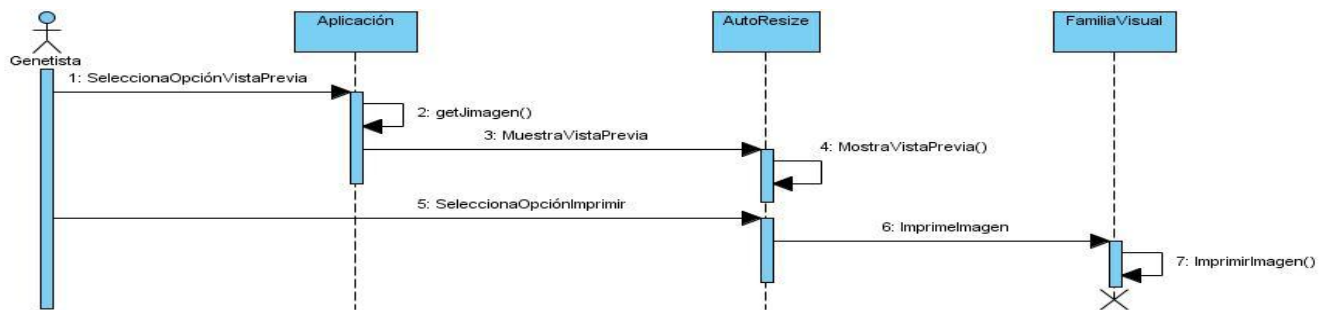


Fig. 15 Diagrama de secuencias del CU Imprimir Árbol Genealógico. Sección Imprimir Vista Previa.

La Fig. 16 representa la relación entre las clases que intervienen en el caso de uso Ordenar Árbol Genealógico. Se muestra una relación entre las clases “Aplicacion” y “FamiliaVisual”, donde se hace la selección de la opción ordenar árbol genealógico. Seguidamente se notifica a la clase “Arbol” que hay que ordenar el árbol genealógico por las generaciones, atendiendo al criterio de mayor a menor. La clase “Arbol” manda a ordenar en la clase “Familia” y posteriormente se devuelven los resultados ordenados para que sean representados por los controladores visuales de la clase “FamiliaVisual”.

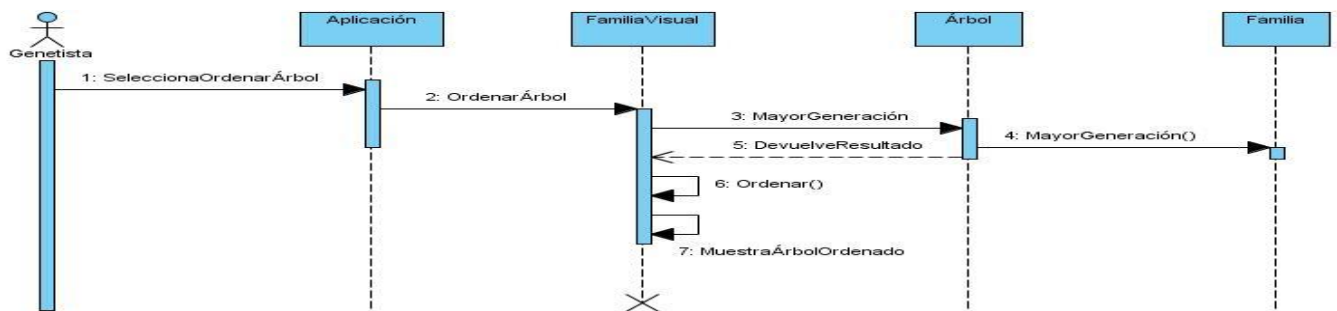


Fig. 16 Diagrama de secuencias del CU Ordenar Árbol Genealógico.

La Fig. 17 representa la interacción entre las clases que intervienen en el caso de uso Buscar en el Árbol Genealógico. Se muestra una relación entre las clases “MuestraBusqueda” y la clase “Arbol”, en la que se le notifica a la clase controladora “Arbol” que se va a efectuar una búsqueda por un criterio determinado. La clase “Arbol” busca en la familia las personas que cumplen con el criterio de búsqueda y posteriormente devuelve los resultados encontrados a la clase “MuestraBusqueda” para que esta los visualice.



Fig. 17 Diagrama de secuencias del CU Buscar en el Árbol Genealógico.

3.5 VISOR DE PROPIEDADES

Diagrama de clases

En la Fig. 16 se muestra el diagrama de clases del visor de propiedades, tiene como característica fundamental que para su uso se hace necesario anotar los atributos de las entidades que se desean mostrar sus propiedades. El componente cuenta con 3 paquetes fundamentales, ellos son "src" que es el paquete donde se tiene la implementación de la entidad que representa la tabla "tableProperty" de propiedades. En este paquete se encuentra además la clase de entrada del componente "PropertyBox" y las clases que representan a los componentes que se van a mostrar como editores de la tabla de propiedad, que no son nativos de la plataforma, es el caso de, "Explorer", "AddComments", "AddMultiple", "FontChooser" y "Picker". Se encuentra además la interface que representa la anotación "PropertyField" y la interface "IPropertyListener" que hace posible registrar un oyente de evento, a través del que se controlan todos los eventos ocurridos en la tabla de propiedades. Se encuentra además la interface "IPropertyBox" que es la que contiene los métodos globales que se van a utilizar para acceder al componente visor de propiedades. Los otros paquetes esenciales que se encuentran dentro del componente, están "Table", que es el paquete que contiene la implementación de la variante de la tabla que hace posible el uso del componente Tree de java. El otro paquete fundamental es "Tree", que es el paquete que tiene la implementación de los nodos del árbol, que representan a cada una de las filas de la tabla "tableProperty".

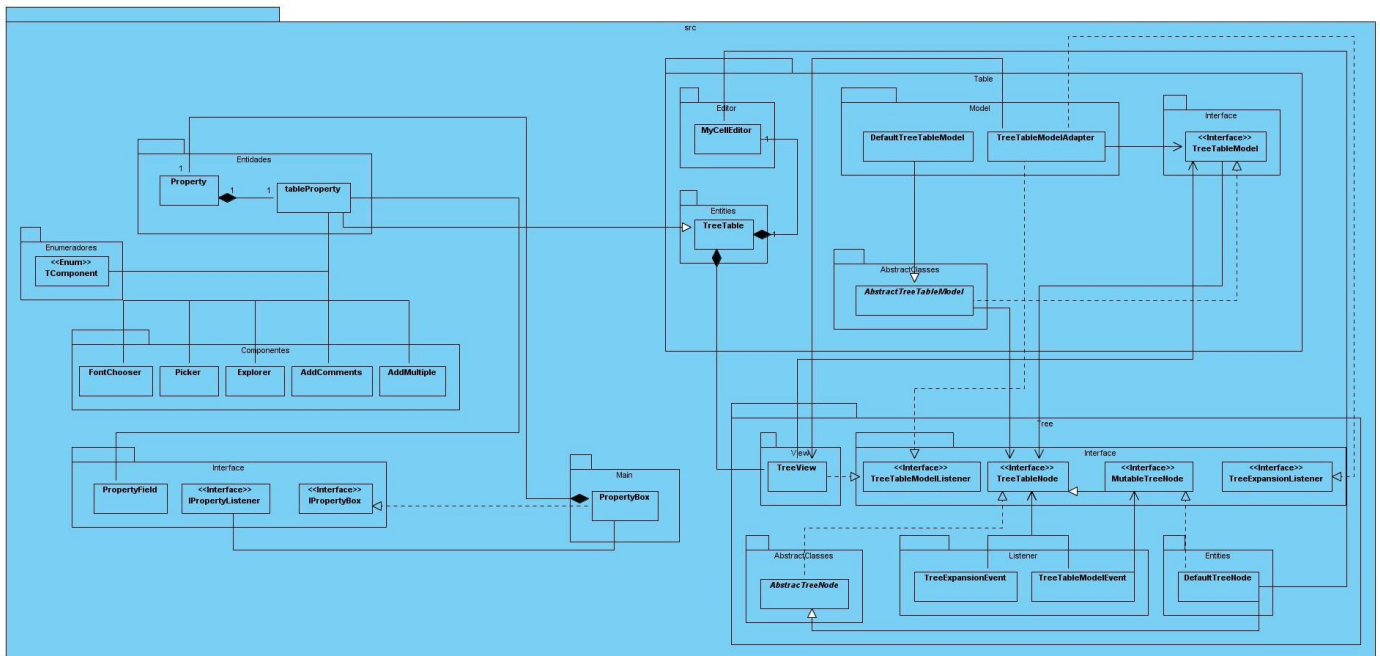


Fig. 18 Diagrama de clases del Visor de Propiedades.

CONCLUSIONES

Como resultado de este capítulo, se representan los diagramas de clases del diseño y los diagramas de interacción del sistema de los cuales se representan los que son novedosos en la versión 2.0 de alasARBOGEN. Se muestra además un diagrama de clases del componente visor de propiedades y se da una explicación a groso modo de los paquetes y clases que se encuentran presentes en su implementación. En todos los casos la explicación de los diagramas de secuencia que se representan, se hace de modo general. Se evidencian los patrones que se utilizan para la implementación, así como el estilo arquitectónico empleado.

CAPÍTULO 4: IMPLEMENTACIÓN.

En este capítulo se describe como fue implementada la aplicación. Algunos fragmentos de códigos no triviales de algunas funcionalidades de la aplicación entre los que se destacan Ordenar Árbol Genealógico y algunas de las funcionalidades que brinda la aplicación para guardar y cargar los datos. Se presentan además algunos casos de prueba de caja negra que se le hicieron a la aplicación, con el objetivo de hacer una validación de la aplicación en la fase de Construcción en el ciclo de vida del RUP.

4.1 DIAGRAMA DE COMPONENTES.

En el diagrama de componentes se representa como está dividido el sistema y las dependencias que existen entre estos componentes. Muestra la organización y dependencia lógica entre los componentes de software, ya sean estos componentes de código fuente, binarios, paquetes bibliotecas, o archivos.

La Fig. 19 representa el diagrama de componentes del sistema que está compuesto por 9 paquetes, donde se relacionan código de fuente y bibliotecas. Es válido aclarar que la figura muestra todos los componentes del sistema sin distinguir por casos de uso, debido a que la estructura es similar en todos los casos.

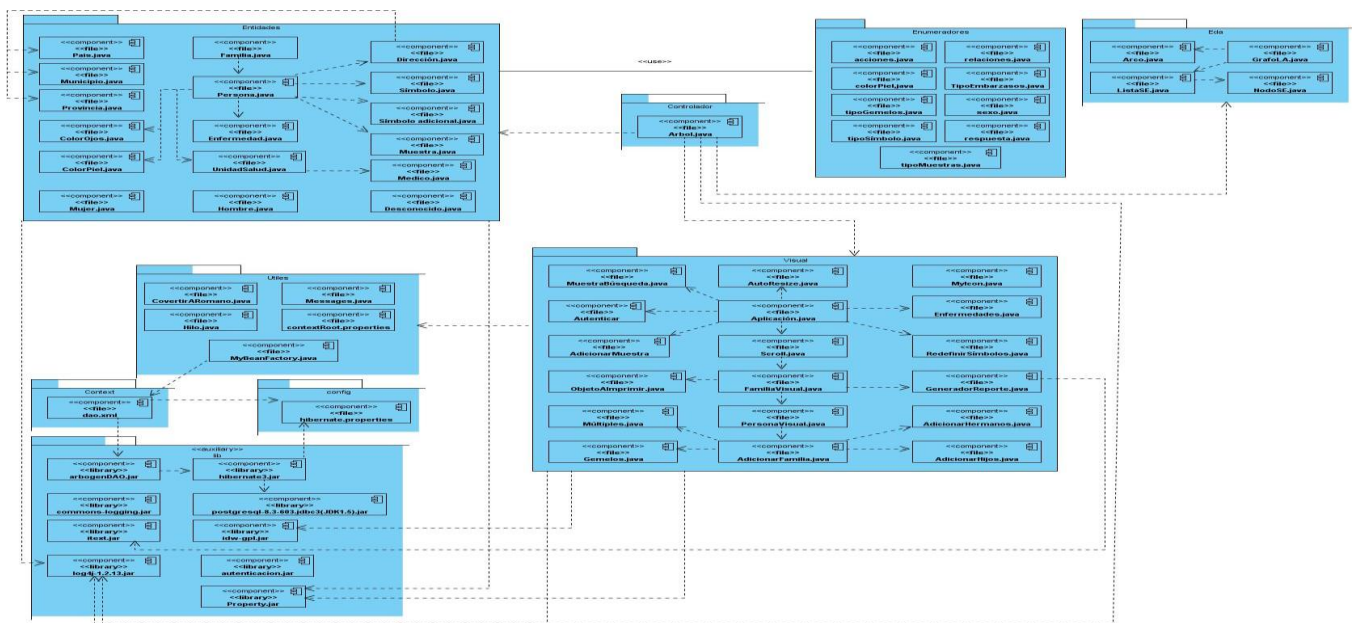


Fig. 19 Diagrama de Componentes.

4.2 FRAGMENTOS DE CÓDIGO

4.2.1 Visor de propiedades

A continuación se describen fragmentos de códigos no triviales de la aplicación que se encuentran en la clase “Persona”, que posibilitan gestionar los datos de la persona a través del visor de propiedades, utilizando los componentes visuales adecuados para visualizar cada atributo anotado.

Para utilizar el visor de propiedades, es necesario hacer el uso de la anotación “*@PropertyField*” que es la que indica que el atributo que se encuentra anotado va a constituir una propiedad de esa entidad. Para conocimiento de los eventos que ocurren en el visor de propiedades se tiene que adicionar un “*Listener*” al visor de propiedades, creando una nueva instancia de la interface “*IPropertyListener*”, haciendo uso del método “*registrarListener(IPropertyListener object)*”. El componente “*property*” reconoce el tipo de editor visual que se va a utilizar acorde a la palabra clave que se le indique haciendo uso de la propiedad “*propertyType()*”. Las palabras que representan los datos son los siguientes:

Comments: Indica que ese atributo al que se está haciendo referencia va a contener un comentario.

Boolean: Indica que dicho atributo, contiene valores booleanos solamente.

Date: Indica que el atributo lo que contiene son valores de fecha.

Color: Indica que el atributo lo que contiene son valores de tipo color.

Font: Indica que en ese atributo lo que se guarda son valores de fuente.

Text: Indica que el atributo representa un texto normal.

List: Indica que el atributo va a representar una lista de valores, posible a desplegarse en un JComboBox.

Tree: Indica que en ese campo se va representar una composición y que se va a crear una sub tabla dentro de la misma tabla, para representar los valores de la composición. Se logra el funcionamiento del JTree.

Add: Representa en algún campo que requiera de la entrada de valores.

AddMultiple: Se tiene en cuenta cuando se va a llenar un atributo de tipo lista, desde una lista de datos origen.

Otro aspecto importante que se puede evidenciar en el visor de propiedades, es que se le puede indicar cuando los valores que va a tener son estáticos y cuando se van a cargar los valores de forma dinámica. Para darle este funcionamiento se utiliza la anotación “*datosExternos()*” que recibe un valor booleano y se le tiene que indicar el nombre del método haciendo uso de la anotación “*nombreMetodo()*”. Esta anotación por defecto está en blanco, por si no se desean cargar los valores de un método externo, en

caso contrario, el nombre del método que se va a especificar en esa anotación, debe pertenecer a la clase que está anotada para cargar los valores externos.

A continuación se muestran fragmentos de códigos de la clase "Persona", donde queda reflejado con claridad como se utilizan las anotaciones para el visor de propiedades.

```
@PropertyField(propertyName="Nombre", propertyCharType = "L", Expresion = "[[a-zA-Z][á-ú][Á-Ú]+\s]*",propertyType="Text", propertyToolTip="Nombre del individuo")  
protected String nombre;
```

```
@PropertyField(propertyName = "Primer Apellido", propertyCharType = "L", Expresion = "[[a-zA-Z][á-ú][Á-Ú]]*",  
propertyType="Text", propertyToolTip="Primer apellido del individuo")  
protected String primerapellido;
```

```
@PropertyField(propertyName = "Segundo Apellido", propertyCharType = "L", Expresion = "[[a-zA-Z][á-ú][Á-Ú]]*",  
propertyType="Text", propertyToolTip="Segundo apellido del individuo")  
protected String segundoapellido;
```

```
@PropertyField(propertyName = "CI", propertyType="Text", propertyCharType = "N", Expresion = "[0-9]*",  
propertyToolTip="Carnet de identidad del individuo")  
protected String ci;
```

```
@PropertyField(propertyName = "Sexo", propertyType="List", noEditable={"Masculino","Femenino"},  
propertyToolTip="Sexo del individuo", datosExternos = true, nombreMetodo = "getListaSexo")  
protected Sexo sex;
```

```
@PropertyField(propertyName = "Color de la Piel", isEditable = true,propertyType="List", propertyToolTip="Color de  
la piel del individuo", datosExternos = true, nombreMetodo = "getListaColorPiel")  
protected ColorPiel colorP;
```

```
@PropertyField(propertyName = "Color de los Ojos", isEditable = true, propertyType="List", propertyToolTip="Color  
de los ojos del individuo", datosExternos = true, nombreMetodo="getColoresOjo")  
protected ColorOjos colorOjos;
```

```
@PropertyField(propertyName = "Edad", propertyCharType = "N",propertyType="Text", propertyToolTip="Edad del  
individuo")  
protected int edad;
```

```
@PropertyField(propertyName = "Talla", propertyType="Text", propertyToolTip="Talla del individuo")  
protected float talla;
```

```
@PropertyField(propertyName = "Peso", propertyType="Text", propertyToolTip="Peso del individuo")  
protected float peso;
```

```
@PropertyField(propertyName = "Fecha Nacimiento", propertyType="Date", propertyToolTip="Fecha en que nace el  
individuo")  
protected Date fechaNacimiento;
```

```
@PropertyField(propertyName = "Fecha Muerte", propertyType="Date", propertyToolTip="Fecha en que muere el individuo")
protected Date fechaMuerte;

@propertyField(propertyName = "Dirección", propertyType="Tree", propertyToolTip="Dirección del individuo")
protected Direccion direccion;

@propertyField(propertyName = "Intento de Suicidio", isEditable= true, propertyType="Boolean",
propertyToolTip="Indica si el individuo a intentado suicidarse")
protected boolean intentoSuicidio;

@propertyField(propertyName = "Propósito", isEditable= true, propertyType="Boolean", propertyToolTip="Indica si el individuo es el propósito del estudio")
protected boolean proposito;

@propertyField(propertyName = "Enfermedades", propertyType="AddMultiple", propertyToolTip="Adicionar enfermedades al individuo", datosExternos = true, nombreMetodo="getListaEnfermedades")
private List<Enfermedad> enfermedades;

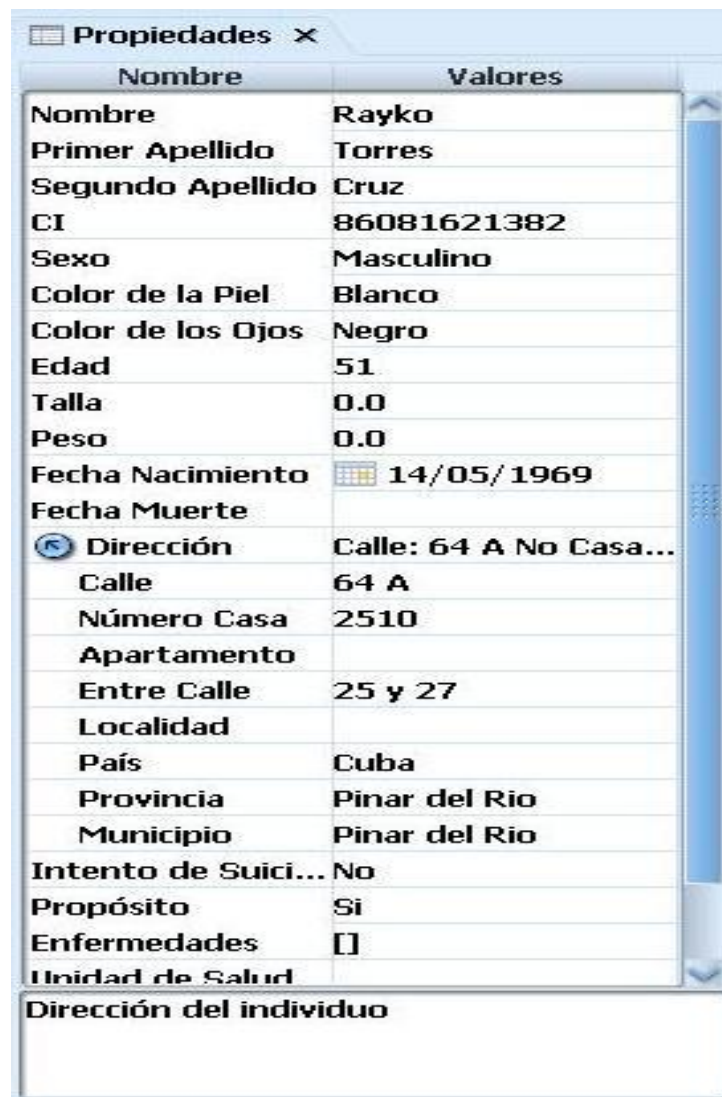
@propertyField(propertyName="Unidad de Salud", propertyType="List", propertyToolTip="Unidad de salud a la que pertenece el individuo", datosExternos = true, nombreMetodo = "getUnidadesSalud")
private UnidadSalud unidadSalud;

@propertyField(propertyName = "Símbolo Adicional", propertyType="List", propertyToolTip="Símbolo adicional del individuo en la familia", datosExternos = true, nombreMetodo="getSimbolosAdicionales")
protected SimboloAdicional simboloAdicional;

@propertyField(propertyName = "Comentario", propertyType="Comments", propertyToolTip="Comentarios sobre las enfermedades del individuo")
protected String comentarios;
```

En el caso de las propiedades que necesitan datos externos, la devolución de los valores debe hacerse a través de una lista genérica de tipo Vector.

```
public Vector<Enfermedad> getListaEnfermedades(){
Vector<Enfermedad> enfermedades = new Vector<Enfermedad>();
try {
    FileInputStream fis = new FileInputStream(new File("Datos/Enfermedades.txt"));
    ObjectInputStream ois = new ObjectInputStream(fis);
    int cantidad = ois.readInt();
    for (int i = 0; i < cantidad; i++) {
        enfermedades.add((Enfermedad) ois.readObject());
    }
    fis.close();
    ois.close();
} catch (FileNotFoundException e) { e.printStackTrace();
} catch (IOException e) { e.printStackTrace();
} catch (ClassNotFoundException e) { e.printStackTrace();
}
return enfermedades;}
```



Nombre	Valores
Nombre	Rayko
Primer Apellido	Torres
Segundo Apellido	Cruz
CI	86081621382
Sexo	Masculino
Color de la Piel	Blanco
Color de los Ojos	Negro
Edad	51
Talla	0.0
Peso	0.0
Fecha Nacimiento	14/05/1969
Fecha Muerte	
<input checked="" type="radio"/> Dirección	Calle: 64 A No Casa...
Calle	64 A
Número Casa	2510
Apartamento	
Entre Calle	25 y 27
Localidad	
País	Cuba
Provincia	Pinar del Rio
Municipio	Pinar del Rio
Intento de Suici...	No
Propósito	Si
Enfermedades	<input type="checkbox"/>
Unidad de Salud	
Dirección del individuo	

Fig. 20 Visor de Propiedades.

4.2.2 Entorno visual

El entorno visual de la aplicación se configuró haciendo uso de la biblioteca "idw-gpl.jar"¹, lo que permite:

- Acoplamiento de nuevas interfaces visuales y áreas de trabajo de forma sencilla.
- Un acceso rápido a todas las áreas de trabajo de la aplicación.
- Cambiar el diseño de la aplicación de forma dinámica y sencilla.

¹ Disponible en: <http://dock.javaforge.com/>

La Fig. 21 muestra la vista principal de la aplicación, se observa como las diferentes áreas de trabajo, como es el caso de “Familia”, “Propiedades” y “Vista Tabular” se encuentran ubicadas haciendo uso de la biblioteca.

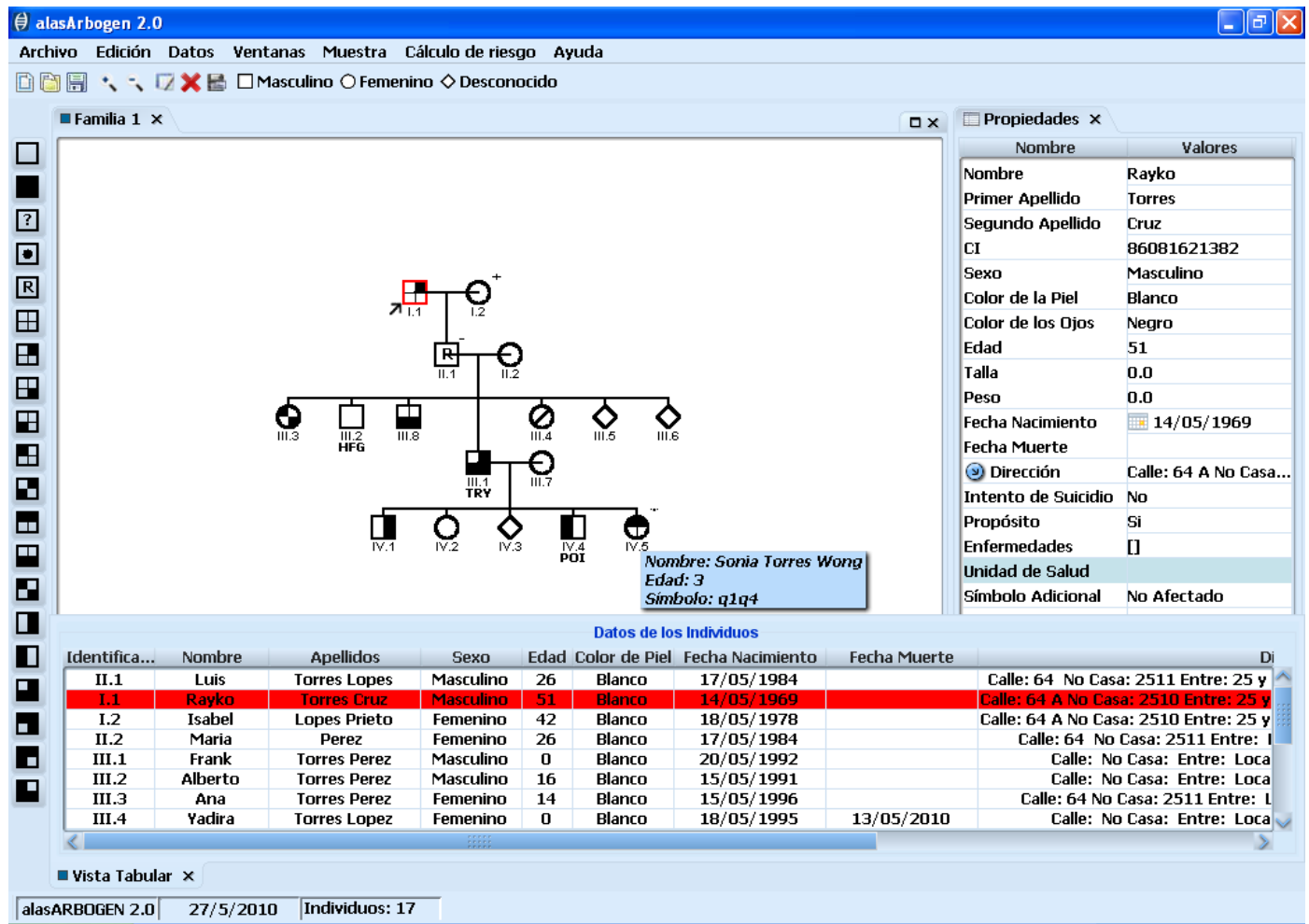


Fig. 21 Vista principal de la aplicación.

4.3 PRUEBAS

A lo largo de todo el proceso de desarrollo del software se realizaron diferentes pruebas funcionales, para éstas se diseñaron casos de pruebas según las funcionalidades descritas en los casos de usos. Cada planilla de caso de prueba recoge la especificación de un caso de uso, dividido en secciones y escenarios, detallando las funcionalidades descritas en él y describiendo cada variable que recoge el caso de uso en cuestión, además quedan plasmadas las revisiones realizadas al caso de prueba; así como un registro de todo aquello que no corresponde a la calidad del software.

A continuación se muestra el caso de prueba asociado al CUS Buscar en el Árbol Genealógico.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC 1: Buscar Intentos de suicidio.	EC 1.1: Seleccionar la opción "Intento de suicidio".	El sistema visualiza las personas encontradas que tuvieron intentos de suicidio.
SC 2: Buscar Nombre.	EC 2.1: Introducir un nombre existente.	El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados y visualiza la información deseada por el genetista.
	EC 2.2: Introducir un nombre no existente.	El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
SC 3: Buscar Enfermedades genéticas.	EC 3.1 Seleccionar la enfermedad genética.	El sistema muestra una tabla que contiene las personas del árbol genealógico, que tienen la enfermedad especificada.
SC 4: Buscar Apellidos	EC 4.1: Introducir un apellido existente.	El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se visualiza la información deseada por el genetista.
	EC 4.2: Introducir dos apellidos existentes.	El sistema no encuentra información relacionada con el criterio de búsqueda

		especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
SC 5: Buscar CI	EC 5.1: Introducir CI	El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se visualiza la información deseada por el genetista.
SC 6: Buscar Área de Salud	EC 6.1: Seleccionar el Área de Salud.	El sistema muestra una tabla que contiene las personas del árbol genealógico, que pertenecen a esa área de salud.

Luego de la descripción de las secciones y los escenarios se describen las variables correspondientes a este CUS.

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	buscar	Botón	No	Inicia la búsqueda.
2	cerrar	Botón	No	Cierra la ventana de búsqueda.
3	selección	Lista desplegable	No	Selecciona el criterio de búsqueda.
4	enfermedades	Lista desplegable	No	Selecciona el tipo de enfermedad que sea buscar.
5	datos	Campo de texto	Sí	Introduce los datos según el criterio de búsqueda.

A continuación se muestran algunas matrices relacionadas con el diseño, donde se probó y evaluó la validez de los datos.

Matriz de datos. CUS Buscar en el árbol genealógico. SC2: Buscar Nombre.

Id. del escenario	Escenario	Variables				Respuesta del sistema	Resultado de la prueba	Flujo central
		1	2	3	5			
EC 2.1	Introducir un nombre existente.	V	V	V	V	El sistema muestra los datos de dicho individuo en una tabla.	Satisfactoria	<ol style="list-style-type: none"> 1. El genetista selecciona la opción "Nombre". 2. El sistema muestra una caja de texto para que el genetista introduzca el nombre de la persona que desee buscar. 3. Una vez que el genetista termina de introducir el nombre de la persona que desea buscar, da clic en el botón "Buscar", para efectuar la búsqueda. 4. El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se visualiza la información deseada por el genetista.

EC 2.2	Introducir un nombre no existente.	V	V	V	V	El sistema no encuentra información relacionada con el criterio "Nombre".	Satisfactoria	4.1. El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
		V	V	V	I			

Matriz de datos. CUS Buscar en el árbol genealógico. SC4: Buscar Apellidos.

Id. del escenario	Escenario	Variables				Respuesta del sistema	Resultado de la prueba	Flujo central
		1	2	3	5			
EC 4.1	Introducir un apellido existente.	V	V	V	V	Muestra el resultado de la búsqueda según el criterio "Apellidos".	Satisfactoria	<ol style="list-style-type: none"> 1. El genetista selecciona la opción "Apellidos". 2. El sistema muestra una caja de texto para que el genetista introduzca los apellidos de la persona que desee buscar. 3. Una vez que el genetista termina de introducir los apellidos de la persona que desea buscar, da clic en el botón "Buscar", para efectuar la búsqueda. 4. El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se visualiza

								la información deseada por el genetista.
EC 4.2:	Introducir dos apellidos existentes.	V	V	V	V	No muestra el resultado de la búsqueda según el criterio "Apellidos".	Insatisfactoria	4.1 El sistema no encuentra información relacionada con el criterio de búsqueda especificado, se muestra un mensaje de alerta al usuario donde se le indica que no se encontró información.
		V	V	V	I			

Matriz de datos. CUS Buscar en el árbol genealógico. SC5: Buscar CI

Id. del escenario	Escenario	Variables				Respuesta del sistema	Resultado de la prueba	Flujo central
		1	2	3	5			
EC 5.1	Introducir CI	V	V	V	V	Muestra el resultado de la búsqueda según el criterio "CI".	Satisfactoria	<ol style="list-style-type: none"> 1. El genetista selecciona la opción "CI". 2. El sistema permite introducir el carnet de identidad de la persona que desee buscar. 3. Una vez que el genetista termina de introducir el número de carnet de identidad de la persona que desea buscar, da clic en el botón "Buscar", para efectuar

								<p>la búsqueda.</p> <p>4. El sistema efectúa la búsqueda atendiendo al criterio y los valores especificados. Se visualiza la información deseada por el genetista.</p>
--	--	--	--	--	--	--	--	--

Matriz de datos. CUS Buscar en el árbol genealógico. SC6: Buscar Área de Salud

Id. del escenario	Escenario	Variables				Respuesta del sistema	Resultado de la prueba	Flujo central
		1	2	3	4			
EC 6.1	Seleccionar el Área de Salud.	V	V	V	V	Muestra el resultado de la búsqueda según el criterio "Área de Salud".	Satisfactoria	<p>1. El genetista selecciona la opción "Área_Salud".</p> <p>2. El sistema muestra una lista que contiene las áreas de salud disponibles en la base de datos.</p> <p>3. El genetista selecciona una unidad de salud y da clic en el botón buscar.</p> <p>4. El sistema muestra una tabla que contiene las personas del árbol genealógico, que pertenecen a esa área de salud.</p>

La siguiente tabla muestra las no conformidades detectadas durante la aplicación de las pruebas, las cuales fueron resueltas satisfactoriamente.

Registro de defectos y dificultades detectados.

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa	No significativa	Estado NC	Resp. equipo de desarrollo
Aplicación	1	En el CUS Buscar en el árbol genealógico en la SC4, cuando se introducen dos apellidos existentes el sistema no muestra el resultado de la búsqueda.	Ventana "Buscar Individuos"	Pruebas de funcionalidad.	X		PD 18/3/10 Resuelta 25/3/10	El sistema muestra los datos del individuo con los apellidos deseados.

CONCLUSIONES.

Como resultado del capítulo se obtuvo el diagrama de componentes y se explicaron algunos aspectos importantes del código de la aplicación. Se realizaron pruebas funcionales a la aplicación obteniéndose resultados satisfactorios.

CONCLUSIONES

- El análisis de los requerimientos funcionales permitió la identificación de nuevas funcionalidades para la aplicación, así como mejoras potenciales en las funcionalidades de alasARBOGEN 1.0.
- Se realizó el diseño de las clases de la aplicación atendiendo al estilo arquitectónico planteado, así como a los patrones de diseño.
- La implementación de los componentes permitió la obtención de una aplicación informática capaz de realizar la representación gráfica de los árboles genealógicos y persistir la información de estos en una base de datos centralizada.
- Como resultado de la investigación se obtiene una aplicación de escritorio multiplataforma para la representación gráfica de los arboles genealógicos, para el sistema alasARBOGEN 2.0.

RECOMENDACIONES

- Realizar pruebas de calidad a la aplicación con el objetivo de ser liberada para su uso generalizado por las instituciones de Genética Médica.
- Implementar las operaciones “Rehacer” y “Deshacer” en futuras versiones.

REFERENCIAS BIBLIOGRÁFICAS

1. Granma. Gobierno cubano apoya estudios en Genética Médica.[En Línea][Citado el: 7 de noviembre del 2009] Disponible en:
<http://www.granma.cu/espanol/2009/julio/vier24/30genetica.html>
2. Babylon. Árbol Genealógico. [En Línea][Citado el: 7 de noviembre de 2009] Disponible en:
http://www.babylon.com/definicion/%C3%A1rbol_geneal%C3%B3gico/spanish
3. GenoPro 2.0.1.4 Programa de Árboles Familiares [En Línea][Citado el: 21 de enero de 2010] Disponible en:
http://www.taringa.net/posts/downloads/1517489/GenoPro-2_0_1_4---Programa-de-%C3%A1rboles-familiares.html
4. BitGen II Heráldica y Árboles Genealógicos [En Línea][Citado el: 21 de enero de 2010] Disponible en:
http://www.quierosoft.com/index.html?target=p_67.html&lang=es
5. BitGen II. Árboles genealógicos y heráldica desde el PC [En Línea][Citado el: 21 de enero de 2010] Disponible en:
<http://www.idg.es/pcworld/BitGen-II.-%C3%81rboles-genealogicos-y-heraldica-desde-/art131220.htm>
6. Sistema General de Documentación Familiar – GDS [En Línea][Citado el: 21 de enero de 2010] Disponible en: http://www.gdsystem.net/v5/es/es_gds_quees.asp
7. GenealogíaMac en su versión 4: ¡Moderna gestión genealógica bajo Mac OS X! [En Línea][Citado el: 21 de enero de 2010] Disponible en:
<http://www.onlymac.de/html/stammbaum4es.html>
8. **Cyrillic** Para dibujo de árboles genéticos y manejo de datos [En Línea][Citado el: 21 de enero de 2010] Disponible en:
<http://www.softwarecientifico.com/paginas/cyrillic.htm>
9. Metodologías ágiles en el Desarrollo de Software [En Línea][Citado el: 21 de enero de 2010] Disponible en: <http://www.willydev.net/descargas/prev/TodoAgil.pdf>
10. Extreme Programming [En Línea][Citado el: 22 de enero de 2010] Disponible en:
http://www.liderdeproyecto.com/articulos/extreme_programming.html

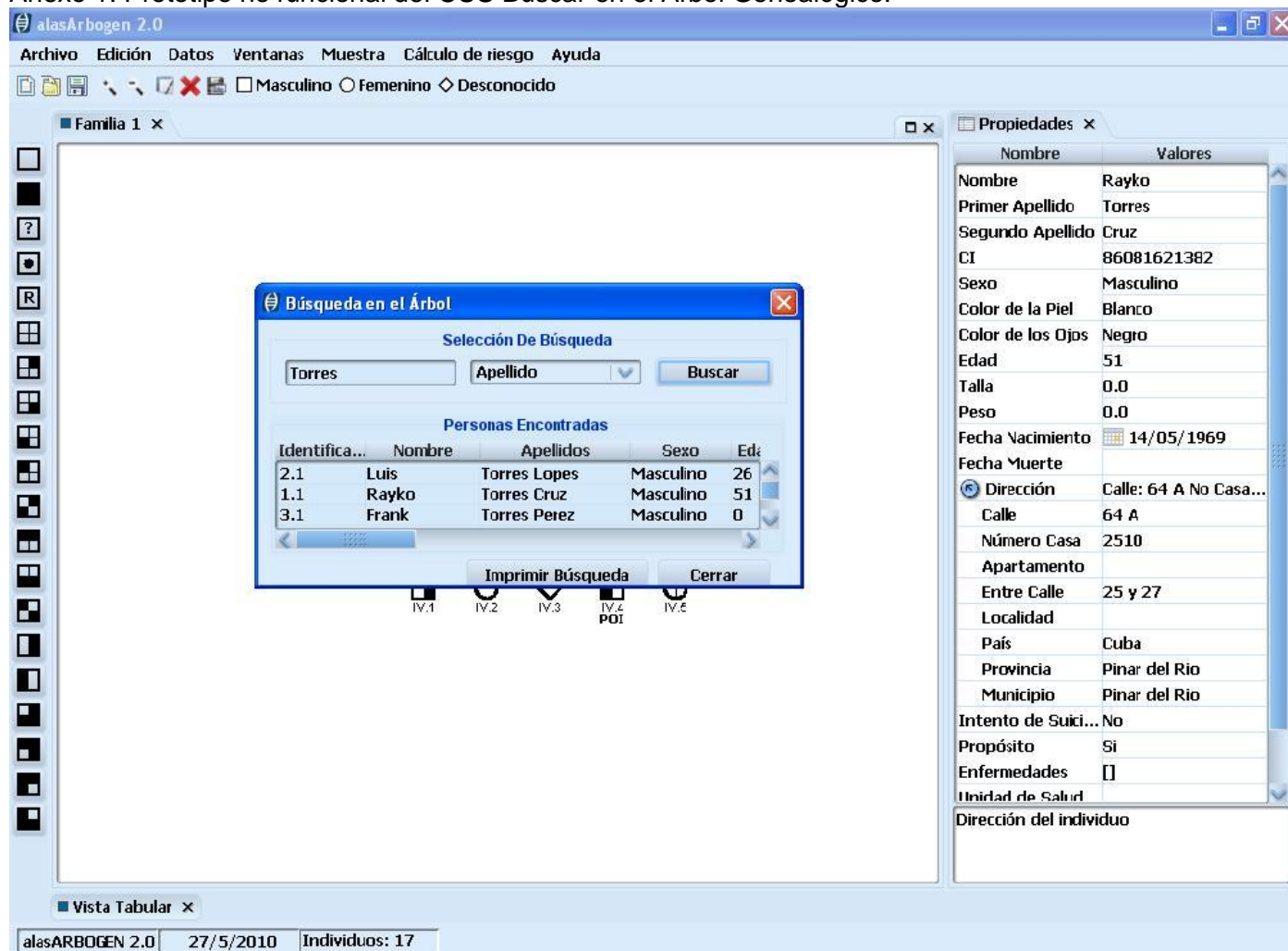
11. LA INGENIERÍA DE SOFTWARE Y RUP [En Línea][Citado el: 22 de enero de 2010] Disponible en: <HTTP://WWW.SLIDESHARE.NET/DERSTIPPENWOLF/LA-INGENIERA-DE-SOFTWARE-Y-RUP>
12. RUP vs XP [En Línea][Citado el: 22 de enero de 2010] Disponible en: <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>
13. Aplicación de la metodología RUP para el desarrollo rápido de aplicaciones basado en el estándar J2EE [En Línea][Citado el: 22 de enero de 2010] Disponible en: http://biblioteca.usac.edu.gt/tesis/08/08_7691.pdf
14. Ayuda extendida del Rational Rose Enterprise Edition 2003. [Citado el: 10 de febrero de 2010].
15. Lenguajes de Programación [En Línea][Citado el: 11 de febrero de 2010] Disponible en: <http://es.kioskea.net/contents/langages/langages.php3>
16. The Java Language Environment [En Línea][Citado el: 11 de febrero de 2010] Disponible en: <http://java.sun.com/docs/white/langenv/>
17. Eclipse [En Línea][Citado el 11 de febrero de 2010] Disponible en: <http://amap.cantabria.es/confluence/display/DEV/Eclipse>
18. Visual Paradigm for UML [En Línea][Citado el: 11 de febrero de 2010] Disponible en: <http://www.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml>
19. Rosado Roselló, Reynaldo y Le Sánchez, Lien. *alasARBOGEN: aplicación informática para la representación de árboles genealógicos*. Habana: UCI, 2008.
20. Arquitectura Modelo Vista Controlador [En Línea][Citado el: 12 de febrero de 2010] Disponible en: <http://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>
21. Algunos tipos de arquitecturas [En Línea][Citado el: 11 de febrero de 2010] Disponible en: <http://homepage.mac.com/imaz/iblog/C612772037/E20050907222635/Medio/Algunos%20Tipos%20de%20Arquitecturas.pdf>

BIBLIOGRAFÍA

- Ariza Rojas, Maribel y Molina García, Juan Carlos. Introducción y principios básicos del desarrollo de software basado en componentes. Septiembre de 2004. [Citado el: 20 de Enero de 2010].
- Ayuda extendida del Rational Rose Enterprise Edition 2003. [Citado el: 18 de abril de 2010].
- Catálogo de Patrones de Diseño J2EE. I.- Capa de Presentación. [En línea] [Citado el: 15 de Noviembre de 2009] Disponible en: <http://www.programacion.net/java/tutorial/patrones/1/>
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. s.l.: PEARSON EDUCACIÓN S.A, 2000.[Citado el: 20 de Enero de 2010]
- Larman, C. UML y Patrones: Introducción al análisis y programación orientada a objetos. México, Prentice Hall, 1999, 536 p.; (MON-001311)[Citado el: 16 de Abril de 2010.] Disponible en: <http://bibliodoc.uci.cu/pdf/reg00062.pdf>
- Pressman, Roger S. Ingeniería del Software: un enfoque práctico. Parte I y II / Madrid, McGraw-Hill, 2002, ed. 5ta. (MON-002581) 601p. [Citado el: 20 de Enero de 2010.] Disponible en: <http://bibliodoc.uci.cu/pdf/reg02689.pdf>
- Smitha Krishna Nagesh , Marina Sum. Designing Patterns With UML. [En línea] 12 de Abril de 2006 [Citado el: 10 de octubre de 2009] Disponible en: http://developers.sun.com/jenterprise/nb_enterprise_pack/reference/techart/uml_patterns.html
- Sun Java Center J2EE Patterns. [En línea] [Citado el: 15 de Noviembre de 2009] Disponible en: <http://java.sun.com/j2ee/patterns/DataAccessObject.html>
- Zukowski, John. Programación en Java J2SE 1.4. Noviembre de 2007. Ciudad Habana, Editorial: Félix Varela, 2007. [Citado el: 15 de Febrero de 2010.]

ANEXOS

Anexo 1. Prototipo no funcional del CUS Buscar en el Árbol Genealógico.



Anexo 2. Prototipo no funcional del CUS Ordenar Árbol Genealógico.

alasArbogen 2.0

Archivo Edición Datos Ventanas Muestra Cálculo de riesgo Ayuda

Ampliar Ctrl-O
Reducir Ctrl-I
Ordenar Árbol
Buscar en el Árbol Genealógico
Seleccionar Todo Ctrl-E
Eliminar

Propiedades x

Nombre	Valores
Nombre	Rayko
Primer Apellido	Torres
Segundo Apellido	Cruz
CI	86081621382
Sexo	Masculino
Color de la Piel	Blanco
Color de los Ojos	Negro
Edad	51
Talla	0.0
Peso	0.0
Fecha Nacimiento	14/05/1969
Fecha Muerte	
Dirección	Calle: 64 A No Casa...
Calle	64 A
Número Casa	2510
Apartamento	
Entre Calle	25 y 27
Localidad	
País	Cuba
Provincia	Pinar del Rio
Municipio	Pinar del Rio
Intento de Suici...	No
Propósito	Si
Enfermedades	
Unidad de Salud	
Dirección del individuo	

Vista Tabular x

alasARBOGEN 2.0 27/5/2010 Individuos: 17

Anexo 3. Prototipo no funcional del CUS Mostrar Vista Previa del Árbol Genealógico.

alasArbogen 2.0

Archivo Edición Datos Ventanas Muestra Cálculo de riesgo Ayuda

Masculino
 Femenino
 Desconocido

Familia 1 x

Propiedades x

Nombre	Valores
Nombre	Rayko
Primer Apellido	Torres
Segundo Apellido	Cruz
CI	86081621382
Sexo	Masculino
Color de la Piel	Blanco
Color de los Ojos	Negro
Edad	51
Talla	0.0
Peso	0.0
Fecha Nacimiento	14/05/1969
Fecha Muerte	
Dirección	Calle: 64 A No Casa...
Intento de Suicidio	No
Propósito	Si
Enfermedades	[]
Unidad de Salud	
Símbolo Adicional	No Afectado
Comentario	

Vista Previa

Vista Previa

Guardar Imprimir Cancelar

Vista Tabular x

alasARBOGEN 2.0 27/5/2010 Individuos: 17

The screenshot displays the 'alasArbogen 2.0' software interface. At the top, there is a menu bar with options: Archivo, Edición, Datos, Ventanas, Muestra, Cálculo de riesgo, and Ayuda. Below the menu bar, there are icons for file operations and a legend for sex (Masculino, Femenino, Desconocido). The main workspace is titled 'Familia 1 x' and contains a genealogical tree diagram. A 'Vista Previa' (Preview) window is open, showing a smaller version of the tree. To the right, a 'Propiedades' (Properties) panel lists various personal and medical data for an individual named Rayko Torres Cruz, including birth date (14/05/1969), sex (Masculino), and address. At the bottom, a status bar shows 'alasARBOGEN 2.0', the date '27/5/2010', and the number of individuals 'Individuos: 17'.

Anexo 4. Prototipo no funcional del CUS Imprimir Árbol Genealógico.

alasArbogen 2.0

Archivo Edición Datos Ventanas Muestra Cálculo de riesgo Ayuda

Nuevo Ctrl-N
Abrir Ctrl-A
Guardar Ctrl-G
Guardar Como
Imprimir
Exportar
Salir

Masculino Femenino Desconocido

Propiedades x

Nombre	Valores
Nombre	Rayko
Primer Apellido	Torres
Segundo Apellido	Cruz
CI	86081621382
Sexo	Masculino
Color de la Piel	Blanco
Color de los Ojos	Negro
Edad	51
Talla	0.0
Peso	0.0
Fecha Nacimiento	14/05/1969
Fecha Muerte	
Dirección	Calle: 64 A No Casa...
Calle	64 A
Número Casa	2510
Apartamento	
Entre Calle	25 y 27
Localidad	
País	Cuba
Provincia	Pinar del Rio
Municipio	Pinar del Rio
Intento de Suici...	No
Propósito	Si
Enfermedades	
Unidad de Salud	
Dirección del individuo	

Vista Tabular x

alasARBOGEN 2.0 27/5/2010 Individuos: 17

Anexo 5. Prototipo no funcional del CUS Generar Reporte del Árbol.

The screenshot shows the 'alasArbogen 2.0' software interface. The main window displays a pedigree chart with four generations (I, II, III, IV). The chart includes symbols for affected individuals (shaded) and deceased individuals (circles with a diagonal slash). Key individuals are labeled: I.1, I.2, II.1, II.2, III.1 (TRY), III.2 (HFG), III.4, III.7, IV.1, IV.2, IV.3, IV.4 (POI), and IV.5. A 'Reporte del Árbol' menu option is visible under the 'Exportar' menu.

On the right side, the 'Propiedades' (Properties) panel is open, displaying personal information for an individual:

Nombre	Valores
Nombre	Rayko
Primer Apellido	Torres
Segundo Apellido	Cruz
CI	86081621382
Sexo	Masculino
Color de la Piel	Blanco
Color de los Ojos	Negro
Edad	51
Talla	0.0
Peso	0.0
Fecha Nacimiento	14/05/1969
Fecha Muerte	
Dirección	Calle: 64 A No Casa...
Calle	64 A
Número Casa	2510
Apartamento	
Entre Calle	25 y 27
Localidad	
País	Cuba
Provincia	Pinar del Rio
Municipio	Pinar del Rio
Intento de Suici...	No
Propósito	Si
Enfermedades	
Unidad de Salud	
Dirección del individuo	

At the bottom of the window, the status bar shows 'alasARBOGEN 2.0', the date '27/5/2010', and 'Individuos: 17'.

GLOSARIO

- **alasMEDIGEN:** Sistema de genética médica que tiene el registro de enfermedades genéticas, malformaciones y defectos congénitos.
- **GENOGRAMA:** es una representación gráfica de una constelación familiar de múltiples generaciones que registra información sobre los miembros de esa familia y sus relaciones.