

Facultad 6

Universidad de las Ciencias Informáticas



**Título: Componente de análisis Biodemográficos para el sistema
SEEGEN-R.**

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

Autor:

Mislaidis Salcedo Díaz.

Tutores:

MsC Yunier Emilio Tejeda Rodríguez

Ing. Yudiel la Rosa González

Ing. Yosvany Arrastia Machín

La Habana, junio de 2015



“Seamos realistas y hagamos lo imposible”

Che.

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 firmamos la presente a los ____ días del mes de _____ del año _____.

Mislaydis Salcedo Díaz

MsC. Yunier Emilio Tejeda Rodríguez

Ing. Yudiel La Rosa González

Ing. Yosvany Arrastia Machín

DATOS DE CONTACTO

Tutores

MsC. Yunier Emilio Tejeda Rodruíguez

Universidad de las Ciencias Informáticas.

yuniere@uci.cu

Ing. Yudiel La Rosa González

Universidad de las Ciencias Informáticas

ylarosag@uci.cu

Ing. Yosvany Arrastia Machin

Universidad de las Ciencias Informáticas

yarrastia@uci.cu

DEDICATORIA

Dedico este trabajo a:

*Mi hijo **Anthony** por ser mi mayor fuente de inspiración a partir de su llegada a esta vida.*

*A mis padres **Gladys y Berto** por apoyarme en todo momento, más que mis padres fueron mis amigos en todo momento.*

*A mi hermana **Mily** por estar ahí conmigo en los momentos más difíciles.*

AGRADECIMIENTOS

A mis tutores **Yudiel la Rosa, Yosvany Arrastia y Yunier Emilio Tejeda** por el apoyo incondicional que siempre me brindaron y la confianza que depositaron en mí, aun cuando todo el mundo creía que el desarrollo de esta tesis era imposible.

A mi esposo **Felipe** por su comprensión y cariño en todos los momentos estresantes. Te agradezco además por darme el mejor regalo de estos 5 años, nuestro hijo.

A mis más grandes amigas **Ili y Yayi** que me soportaron en todos estos años y me brindaron su apoyo en los momentos más difíciles.

A toda mi familia porque aunque están lejos, siempre estuvieron apoyándome en especial a mis tías **Vivian y Nürka**, mi prima **Yusneidis** y mi cuñado **Yoel**.

A toda la familia de mi esposo por acogerme y permitirme ser parte de esa familia, en especial a mis suegros **Adela y Felipe**.

A todos los profesores que me formaron en el transcurso de todos estos años, en especial a **Yordanis, Neysis, Elenis, Marisel, Zayli, Yuliet, Yuned, Adisley y Garnache**. Agradecer además a todos los profesores del laboratorio 103 que en algún momento aportaron al desarrollo de esta investigación.

A los profesores de tribunal **Yunet González, Irina Collazo, Roberto Tellez** y a la oponente **Johanna Carbonell** que me permitieron crecer como investigadora.

A todos mis compañeros de aula en el paso por esta universidad y a todos los que de una forma u otra aportaron a mi formación y al desarrollo de este trabajo, en especial a **Tahimí, Laura, Ernesto, Luis Enrique, Alex, Frank y Andy**.

A todas aquellas personas que de una forma u otra, participaron o colaboraron en mi formación como persona y profesional, a todos ustedes gracias.

RESUMEN

El presente trabajo consiste en el desarrollo de un componente para análisis Biodemográficos para el sistema SEEGEN-R. Durante la investigación se realizó el estudio de los diferentes software que realizan análisis estadísticos en estudios Biodemográficos identificándose que los mismos por si solos no son suficientes para realizar análisis demográficos de acuerdo a las características que estos presentan. Se utilizó como metodología de desarrollo RUP, metodología establecida en la línea arquitectónica definida para el desarrollo de componentes para SEEGEN-R. Con el desarrollo del componente para la aplicación SEEGEN-R se obtuvo una herramienta que posibilita a los genetistas del Centro Nacional de Genética Médica (CNGM) los cálculos estadísticos de estudios Biodemográficos centralizados en un solo sistema.

Palabras claves: análisis estadísticos, Biodemografía, Centro Nacional de Genética Médica, estudios Biodemográficos.

ABSTRACT

This work involves the development of a component for biodemographic analysis for the SEEGEN-R system. During the investigation the study of different software that performs statistical analysis was performed. Biodemographic studies identified that they alone are insufficient for demographic analysis according to the features they present. It was used as RUP development methodology, methodology established in architectural line defined for the development of components for SEEGEN-R. With the development of components for application-R SEEGEN a tool that enables geneticists at the National Center of Medical Genetics (CNGM) statistical calculations biodemographic studies centralized in a single system was obtained.

Keywords: statistical analysis, Bio-demographics, National Center of medical genetics, bio-demographic studies.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO I: Fundamentación teórica sobre los análisis Biodemográficos en Cuba.	4
Introducción.....	4
1.1 Estudios Biodemográficos	4
1.1.1 Estudios Biodemográficos en Cuba.	5
1.2 Software que permiten realizar cálculos estadísticos.....	5
1.3 Metodología de desarrollo de software	7
1.3.1 Rational Unified Process (RUP)	7
1.4 Tecnologías y Herramientas.....	8
1.4.1 Lenguaje Unificado de Modelado (UML)	8
1.4.2 Visual Paradigm v8.0	8
1.4.3 Lenguaje de programación R 3.2.0	9
1.4.4 Lenguaje de programación Java 1.6	9
1.4.5 Entorno de Desarrollo Integrado	10
1.4.6 Biblioteca de clases	11
Conclusiones parciales.....	11
CAPÍTULO II: Análisis y diseño del componente de Análisis Biodemográficos para el sistema SEEGEN-R.	12
Introducción.....	12
2.1 Modelos de negocio	12
2.1.1 Actores del negocio.....	12
2.1.2 Trabajadores del negocio.....	12

2.1.3 Diagrama de casos de uso del negocio (CUN).....	13
2.1.4 Descripción textual del caso de uso del negocio.....	13
2.1.5 Diagrama de actividades.....	14
2.1.6 Modelo de objetos del negocio.....	15
2.1.7 Reglas del negocio.....	15
2.2 Especificación de los requisitos del componente informático.....	15
2.2.1 Requisitos Funcionales.....	16
2.2.2 Requisitos No Funcionales.....	17
2.3 Definición de los casos de uso del sistema	18
2.3.1 Actores del sistema.....	18
2.3.2. Paquetes del sistema.....	19
2.3.3. Casos de uso del sistema	19
2.3.4 Descripción de casos de uso del sistema.....	20
2.4 Patrones arquitectónicos	32
2.4.1 Arquitectura en N capas.....	32
2.4.2. Patrones arquitectónicos aplicados.....	33
2.4.3 Vista lógica del sistema.....	33
2.5 Diagrama de clases del diseño.....	36
2.7 Diagrama de secuencias	38
2.8 Patrones de diseño.....	39
2.8.1 Patrones GRASP.....	39
2.8.2 Patrones GoF.....	41
Conclusiones parciales.....	42

CAPÍTULO III: Implementación y pruebas del componente de Análisis Biodemográficos para el sistema SEEGEN-R.....	43
Introducción.....	43
3.1 Diagrama de componentes.....	43
3.2 Estándares de codificación.....	45
3.3 Fragmentos de código.....	45
3.3.1 Fragmento de código R.....	46
3.3.2 Fragmento de código Java.....	48
3.4 Pruebas.....	50
3.4.1 Diseño de casos de prueba.....	53
3.4.2 Resultados de las pruebas aplicadas.....	55
Conclusiones parciales.....	58
CONCLUSIONES GENERALES.....	59
RECOMENDACIONES.....	60
REFERENCIAS BIBLIOGRÁFICAS.....	61
BIBLIOGRAFÍA.....	64
GLOSARIO DE TÉRMINOS.....	65

ÍNDICE DE FIGURAS

Fig. 1 Diagrama de CUN: Realizar estudios de análisis Biodemográficos.....	13
Fig. 2 Diagrama de actividades del negocio.....	14
Fig. 3 Modelo de objetos del negocio.....	15
Fig. 4 Diagrama de paquetes del sistema.....	19
Fig. 5 Diagrama de CUS: Paquete de Estudios Biodemográficos.....	19
Fig. 6 Vista lógica del sistema.....	34
Fig. 7 Diagrama de clases del diseño.....	36
Fig. 8 Diagrama de secuencia: CUS Realizar estudios de tablas de frecuencias de apellidos.....	39
Fig. 9 Ejemplo del patrón Experto.....	40
Fig. 10 Ejemplo del patrón Bajo Acoplamiento.....	40
Fig. 11 Ejemplo del patrón Creador.....	41
Fig. 12 Ejemplo del patrón Alta Cohesión.....	41
Fig. 13 Ejemplo del patrón Singleton.....	42
Fig. 14 Diagrama de componentes.....	44
Fig. 15 Fragmentos de código R de la función Fst.....	46
Fig. 16 Fragmentos de código R de la función hedrick.....	46
Fig. 17 Fragmentos de código R de la función lasker.....	47
Fig. 18 Fragmentos de código R de la función uri.....	47
Fig. 19 Fragmentos de código R de la función rri.....	47
Fig. 20 Fragmentos de código R de la función rel.cond.....	48
Fig. 21 Fragmentos de código R de la función rel.phi.....	48
Fig. 22 Fragmentos de código java de la función fst.....	48
Fig. 23 Fragmentos de código de la función Hedrick.....	49
Fig. 24 Fragmentos de código java de la función lasker.....	49
Fig. 25 Fragmentos de código java de la función uri.....	49
Fig. 26 Fragmentos de código java de la función rri.....	50
Fig. 27 Fragmentos de código java de la función rel_cond.....	50
Fig. 28 Fragmentos de código java de la función rel_phi.....	50
Fig. 29 Registro de No Conformidades.....	55

Fig. 30 Integración del componente al sistema SEEGEN-R.....56

Fig. 31 Resultados obtenidos en el sistema SEEGEN-R.....57

Fig. 32 Resultados obtenidos en R.57

ÍNDICE DE TABLAS

Tabla. 1 Actores del negocio.....	12
Tabla. 2 Trabajadores del negocio.....	12
Tabla. 3 Descripción textual del caso de uso del negocio	13
Tabla. 4 Actores del sistema.....	18
Tabla. 5 Descripción CUS: Realizar estudios de tablas de frecuencias de apellidos.....	20
Tabla. 6 DCD: AnalysisSurnames.java	37
Tabla. 7 DCD: RSurnames.java	37
Tabla. 8 CP: Realizar estudios de tablas de frecuencias de apellidos.....	53
Tabla. 9 Descripción de variables	53
Tabla. 10 Matriz de datos del CU: Realizar estudios de frecuencias de apellidos	54

INTRODUCCIÓN

La Genética es una disciplina de la Medicina surgida a mediados de los años 80 que trata los genes y los mecanismos que regulan la transmisión de los caracteres hereditarios. En los últimos años ha tenido un desarrollo notable gracias a la Biología molecular y al despliegue de sus marcadores genéticos, además de los avances de la tecnología en esta rama.

Actualmente, los avances en este campo de investigación se encuentran en plena expansión, ofreciéndose diversos programas educativos y de investigación por todo el mundo. La Sociedad Internacional de Genética suma cada año nuevos miembros, evidenciando así su evolución como ciencia a partir del desarrollo y el uso de dicha ciencia en los últimos estudios genéticos.

La diversidad Genética puede aparecer espacialmente estructurada a diferentes escalas: población, subpoblación o en un grupo de individuos próximos (Escudero, y otros, 2003), y su distribución se ve influenciada, no sólo por la acción de fuerzas evolutivas tales como la selección natural y migración, sino también, por procesos demográficos y sucesos históricos, que pueden influir sobre el grado de expansión, contracción y fragmentación de las poblaciones. (Strorz, y otros, 2002)

La Biodemografía es el estudio cuantitativo de la Demografía y la Genética de las poblaciones humanas. Cuenta con fuentes de información y métodos de análisis que proporcionan datos valiosos sobre las poblaciones, los cuales son muy útiles para interpretar la diversidad humana en términos genéticos y en el eje espacio/tiempo. (Strorz, y otros, 2002)

Un aspecto de gran interés, y en gran parte responsable de la actual diversidad humana, son los procesos migratorios ocurridos en nuestra especie desde su aparición. Teniendo efecto sobre la estructura Genética de las poblaciones actuales y la consiguiente dificultad de delimitación Genética de los grupos humanos. Esto ha dado como resultado la aparición de grupos híbridos en áreas geográficas muy alejadas de las zonas de origen.

Los estudios sobre análisis Biodemográficos en Cuba se realizan utilizando diferentes *software* estadísticos, los cuales no cubren muchas de las funcionalidades demandadas por los especialistas y en muchos casos son herramientas propietarias, requiriendo el pago de licencias en cada una de las instituciones donde se vayan a utilizar, lo que constituye un gasto considerable para el país. Al mismo tiempo, estas herramientas no son pertinentes para la realización de estos estudios y no arrojan un resultado completo sobre los mismos, teniendo los especialistas que recurrir a cálculos manuales de fórmulas matemáticas complejas para completar la investigación iniciada.

El Sistema Estadístico de Epidemiología Genética (SEEGEN-R) es una solución informática desarrollada para el análisis estadístico en estudios de Epidemiología Genética, desarrollado en la Universidad de las Ciencias Informáticas en coordinación con el Centro Nacional de Genética Médica. En la actualidad el sistema SEEGEN-R carece de funcionalidades que le permita la realización de estudios Biodemográficos mediante métodos estadísticos.

Se define como **problema científico de la investigación**: ¿Cómo contribuir a la realización de estudios Biodemográficos utilizando análisis estadísticos en el sistema SEEGEN-R?

Se define como **objeto de estudio**: análisis estadísticos en estudios Biodemográficos mediante sistemas informáticos, enmarcado en el **campo de acción**: los análisis estadísticos en estudios Biodemográficos para el sistema SEEGEN-R.

Para darle solución al problema antes identificado se plantea como **objetivo general**: Desarrollar un componente para la realización de análisis estadísticos de estudios Biodemográficos en el sistema SEEGEN-R.

Para dar cumplimiento al objetivo general se definen las siguientes **tareas de investigación**:

- Caracterización de las aplicaciones existentes que realizan análisis Biodemográficos para determinar su alcance e impacto.
- Caracterización del sistema SEEGEN-R para el desarrollo del componente informático de análisis Biodemográficos.
- Selección de la metodología, tecnologías y herramientas para el desarrollo del componente informático de análisis Biodemográficos.
- Identificación de requisitos para el componente informático de análisis Biodemográficos.
- Análisis del componente informático de análisis Biodemográficos.
- Diseño del componente informático de análisis Biodemográficos.
- Implementación de las funcionalidades diseñadas para el componente informático de análisis Biodemográficos.
- Validación del componente informático de análisis Biodemográficos para garantizar el cumplimiento de los requisitos mediante pruebas de software.

Para darle solución al problema en cuestión, fueron empleados los siguientes **métodos de investigación**:

Métodos lógicos

1. Analítico-Sintético: división del todo en partes, con el objeto de estudiarlas y analizarlas por separado. Reunión racional de varios elementos en una nueva totalidad. Se realiza sobre la base de los resultados del análisis. (Barchini, 2005)
 - Se proponen un grupo de tareas de investigación con el propósito de obtener un componente informático para los análisis Biodemográficos.
2. Sistémico: estudio de los sistemas en su totalidad, complejidad y dinámica propia. (Barchini, 2005)
 - Se estudian los temas relacionados con los análisis Biodemográficos y se aplican de acuerdo a las necesidades del sistema.

Métodos empíricos

3. Método Investigación- Acción: incluye diagnóstico del problema, intervención de acción y aprendizaje reflexivo. (Barchini, 2005)
 - Se realiza un estudio de los principales temas relacionados con los análisis Biodemográficos para dar solución al problema de investigación.
4. Estudio de Casos: involucran una persona y muchas observaciones. (Barchini, 2005)
 - Con el objetivo de realizar observaciones de casos anteriores donde se haya hecho uso de los análisis Biodemográficos.

La presente investigación cuenta con la siguiente estructura capitular:

Capítulo 1: Fundamentación teórica sobre los análisis Biodemográficos en Cuba. En este capítulo se presenta una breve descripción de los estudios sobre análisis Biodemográficos en Cuba. Se definen las herramientas, tecnologías y metodología a utilizar para el desarrollo del componente informático.

Capítulo 2: Análisis y diseño del componente de Análisis Biodemográficos para el sistema SEEGEN-R. En este capítulo se exponen los requisitos funcionales y no funcionales del componente identificándose así las funcionalidades del mismo, representado por los diferentes actores.

Capítulo 3: Implementación y pruebas del componente de Análisis Biodemográficos para el sistema SEEGEN-R. En este capítulo se desarrolla el modelo de implementación y se aplican las pruebas de sistema, integración y unidad una vez concluida la implementación del componente para validar las funcionalidades de dicho componente.

CAPÍTULO I: Fundamentación teórica sobre los análisis Biodemográficos en Cuba.

Introducción

En este capítulo se presenta una descripción de los estudios sobre análisis Biodemográficos en Cuba, para lograr un mayor entendimiento del contenido de la presente investigación. Se explican brevemente las herramientas existentes para el desarrollo de análisis estadísticos en estudios Biodemográficos. Además se definen las herramientas, tecnologías y la metodología a utilizar para el desarrollo del componente informático.

1.1 Estudios Biodemográficos

Para comprender los estudios Biodemográficos se hace necesario entender los siguientes conceptos:

Demografía: “*estudio estadístico de una colectividad humana, referido a un determinado momento o a su evolución.*” (Real Academia, 2015)

Genética de las poblaciones: la Genética de las poblaciones estudia la composición Genética de las poblaciones y los cambios en estas poblaciones, que pueden estar influenciados por factores evolutivos. Permite explicar la variabilidad natural en las poblaciones. Estudia entre individuos de una misma población la base Genética de la variación fenotípica. (Ecured, 2015)

Biodemografía: estudia la integración de la Demografía y la Genética de las poblaciones. Es una nueva rama de la Demografía humana que ocupa la comprensión de los determinantes biológicos y demográficos complementarios. Incluye, por otra parte, las interacciones entre los procesos de nacimiento y muerte que dan forma a las personas, las cohortes y poblaciones. El componente biológico trae Demografía humana en el marco teórico unificador de la evolución, y el componente demográfico ofrece una base analítica para muchos de los principios sobre los que descansa la teoría de la evolución como la aptitud, la selección, la estructura y el cambio. (Biodemografía, 2015)

Biodemografía complementa, involucra e informa a la investigación sobre el envejecimiento humano; mediante modelos matemáticos, estadísticos y pruebas de hipótesis utilizando métodos experimentales. (Biodemografía, 2015)

1.1.1 Estudios Biodemográficos en Cuba.

Los estudios Biodemográficos en Cuba están asociados al estudio de las poblaciones considerado desde un punto de vista cuantitativo y estadístico. Partiendo de tratamiento de factores como la dimensión de las poblaciones, estructura y características generales, donde se tienen en cuenta las particularidades geográficas y socioeconómicas de la población que se estudia. (Vázquez Sánchez, 2014)

1.2 Software que permiten realizar cálculos estadísticos

En la actualidad existen diversos *software* estadísticos encargados del análisis de datos, con el propósito de dar solución a problemas que manualmente son complejos de resolver. A continuación se describen las características de los principales *software* encargados de realizar estos estudios en el campo de la genética y la Biodemografía.

IBM SPSS Statistics

IBM SPSS Statistics es una familia de *software* estadístico integrada que se centra en el completo proceso analítico, desde la planificación a la colección de datos y al análisis, presentación de informe y despliegue. Cuenta con módulos integrados que permiten encontrar las capacidades especializadas para conducir la investigación y tomar decisiones. SPSS Statistics es un *software* propietario por lo que resulta altamente costoso adquirir su licencia. (SPSS, 2015) Entre los métodos estadísticos que presenta no se encuentran relacionados los métodos necesarios para la realización de estudios Biodemográficos que se mencionan a continuación:

- Estudios sobre modelos de migración Malecot.
- Estudios sobre tablas de frecuencia de apellidos.
- Estudios sobre datos de matrimonios.
- Estudio sobre matrices.

InfoStat

InfoStat es un *software* estadístico desarrollado por un equipo de trabajo conformado por docentes-investigadores de Estadística, Biometría y Diseño de Experimentos de la Universidad Nacional de Córdoba. Tiene como característica su conexión con R de dos maneras: mediante un intérprete integrado el cual le permite ejecutar un script de R sin salir del ambiente de trabajo de InfoStat y mediante

el desarrollo de aplicaciones utilizando el motor de cálculo de R pero con su interfaz. Es un software propietario que está en constante evolución, por lo que sería un gasto considerable para el país la adquisición de la licencia para mantenerlo activo. (InfoStat, 2015) Además presenta como deficiencia que en ninguna de sus actualizaciones han incluido los métodos de estudios Biodemográficos que son necesarios para los genetistas del CNGM que se listan a continuación:

- Estudios sobre modelos de migración Malecot.
- Estudios sobre tablas de frecuencia de apellidos.
- Estudios sobre datos de matrimonios.
- Estudio sobre matrices.

Epidat

Epidat es un software de distribución libre para el análisis epidemiológico y estadístico que permite importar datos desde archivos en formato dBase, Excel o Access. Es de libre distribución y está compuesto por 12 módulos adaptados a su propia entrada de datos. Estos módulos abarcan una amplia variedad de técnicas estadísticas. En la versión 3.0 se incluye el módulo de Demografía, el cual permite la construcción de pirámides de población y tablas de mortalidad. Además permite el cálculo de años potenciales de vida perdidos y años de esperanza de vida perdido. Aun cuando este *software* permita realizar estudios sobre demografía, estos están enfocados a análisis epidemiológicos, por lo que no incluye los estudios específicos de demografía vinculados a la genética de poblaciones. (Epidat, 2014)

Mx

Mx es un *software* que combina un intérprete de álgebra matricial y un optimizador numérico. Este sistema permite la exploración de álgebra matricial a través de funciones dentro de las que se encuentran el modelado de ecuaciones estructurales y modelado estadístico de datos. Permite definir por parte del usuario sus propias funciones de ajuste y optimización pero no incluye los cálculos con matrices asociados a cálculo de valores de isonimia, funciones sobre matrices estocásticas y tablas de frecuencia de apellidos. (Mx, 2015)

SEEGEN-R

El Sistema Estadístico de Epidemiología Genética (SEEGEN-R) es un sistema realizado con el objetivo de agrupar los diferentes estudios en la Genética Médica. La arquitectura basada en extensiones de

SEEGEN-R permite dejar abierto el alcance del sistema, brindando la posibilidad de mejorar las extensiones existentes o de desarrollar nuevas extensiones para otros estudios. Actualmente el sistema cuenta con módulos asociados a los estudios de Genética Poblacional, Epidemiología Genética y Epidemiología Tradicional. (La Rosa González, y otros, 2014)

Pese a todas las funcionalidades que el sistema SEEGEN-R le provee a los especialistas del CNGM, este carece de alcance en cuanto a los estudios Biodemográficos para suplir las necesidades de los especialistas. No cuenta con un módulo que le permita realizar los análisis que se listan a continuación:

- Estudios sobre modelos de migración Malecot.
- Estudios sobre tablas de frecuencia de apellidos.
- Estudios sobre datos de matrimonios.
- Estudio sobre matrices.

1.3 Metodología de desarrollo de software

Una metodología de desarrollo de *software* es utilizada para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información. Actualmente existen diversos tipos de metodologías, donde se seleccionan en dependencia de las características del sistema. (Cendejas Valdéz, 2014)

1.3.1 Rational Unified Process (RUP)

La metodología de desarrollo RUP proporciona una forma estructurada al crear el ciclo de vida del *software* dado que proporciona un plan específico para cada paso del proceso de desarrollo, ayuda a prevenir el desperdicio de recursos y reduce los costes de desarrollo inesperados. Identifica quién hace qué, cuándo y cómo en un proyecto de *software*. Es incremental y posee una implementación de desarrollo en espiral. Está guiada por casos de uso y centrada en la arquitectura del proyecto. RUP divide el proceso de desarrollo de *software* en 4 fases involucrando el modelado de negocio, análisis, diseño, implementación, prueba y despliegue. (RUP, 2015)

Se decide adoptar como metodología por estar basada en las buenas prácticas que se han probado en el campo de la Ingeniería de Software. Es la metodología definida en el proyecto SEEGEN-R, por lo que el componente de análisis Biodemográficos aprovecha las facilidades que RUP ofrece para un mayor

entendimiento y generar una amplia documentación, en correspondencia con los artefactos anteriormente desarrollados.

1.4 Tecnologías y Herramientas

Las tecnologías permiten a los desarrolladores aplicar un grupo de conocimientos necesarios con el objetivo de crear sistemas informáticos que satisfagan un problema dado. Por su parte las herramientas son un conjunto de programas utilizados por los desarrolladores y diseñadores para construir sistemas. (PRESSMAN, 2002)

1.4.1 Lenguaje Unificado de Modelado (UML)

El lenguaje UML v2.0 es un estándar diseñado para visualizar, especificar, construir y documentar un *software*. (Booch, y otros, 2014) En este caso, este lenguaje se centra en la representación gráfica de un sistema. Permite expresar mediante diagramas de una forma gráfica los elementos de forma tal que se puedan entender, así como las características de un sistema antes de su construcción. (Hernández Orallo, 2015) [15] Tiene como ventajas principales:

- Unificar distintas notaciones.
- Permite realizar una verificación y validación del modelo realizado.
- Permite generar código a partir de modelos y viceversa
- Corrección de errores viables en todas las etapas.

1.4.2 Visual Paradigm v8.0

Visual Paradigm v8.0 es una herramienta que posee un diseño centrado en casos de uso y enfocado al negocio de forma tal que genera un *software* con mayor calidad. Además, soporta los últimos estándares de la notación UML. Es una herramienta multiplataforma por lo que es operable en cualquier sistema operativo. Permite realizar los diagramas y artefactos que se generan durante todo el desarrollo del *software*. Permite producir informes en formato PDF, Word o en formato HTML. (VP, 2015)

1.4.3 Lenguaje de programación R 3.2.0

R es un lenguaje y entorno de programación para el análisis estadístico y gráfico. Permite el almacenamiento y la manipulación efectiva de los datos, contiene operadores para cálculos sobre variables de tipo *Array*. Presenta una amplia, coherente e integrada colección de herramientas permitiendo el análisis de datos, además de posibilidades gráficas para análisis de datos. Incluye un conjunto de operadores para cálculos matriciales. (Zamora Araya, y otros, 2012)

R ofrece una amplia variedad de técnicas de análisis de series de tiempo lineal y modelado no lineal, pruebas estadísticas clásicas, clasificación y agrupamiento. Está disponible como *software* libre bajo los términos de la Licencia Pública General de GNU de la *Free Software Foundation* en forma de código fuente. Puede ser compilado y ejecutado en Linux y Windows. (Venables, y otros, 2015.)

1.4.4 Lenguaje de programación Java 1.6

El lenguaje de programación Java permite la modularidad por lo que se pueden hacer rutinas individuales que sean usadas por más de una aplicación. Permite el desarrollo de aplicaciones bajo el esquema de Cliente Servidor, como de aplicaciones distribuidas, lo que lo hace capaz de conectar dos o más computadoras u ordenadores, ejecutando tareas simultáneamente. (IBM, 2015)

Estudiando la arquitectura del lenguaje Java se pueden definir entre sus características principales:

- **Multiplataforma**

Es multiplataforma lo que permite desarrollar programas que se ejecuten sin problemas en sistemas operativos como Windows, Linux, Mac, Unix. (IBM, 2015)

- **Orientado a objetos**

Los objetos agrupan en estructuras encapsuladas tanto sus datos como los métodos (o funciones) que manipulan esos datos. Soporta las características propias de la programación orientada a objetos: clase, objeto, herencia, encapsulamiento y polimorfismo. (IBM, 2015)

- **Distribuido**

Proporciona una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas. (IBM, 2015)

- **Interpretado y compilado a la vez**

Es compilado, en la medida en que su código fuente se transforma en una especie de código máquina, los *bytecodes*, semejantes a las instrucciones de ensamblador. Por otra parte, es interpretado, ya que los *bytecodes* se pueden ejecutar directamente sobre cualquier máquina a la cual se hayan portado el intérprete y el sistema de ejecución en tiempo real. (IBM, 2015)

- **Robusto**

Fue diseñado para crear *software* altamente fiable. Para ello proporciona numerosas comprobaciones en compilación y en tiempo de ejecución. Sus características de memoria liberan a los programadores de una familia entera de errores (la aritmética de punteros), ya que se ha prescindido por completo de los punteros, y la recolección de basura elimina la necesidad de liberación explícita de memoria. (IBM, 2015)

1.4.5 Entorno de Desarrollo Integrado

Un entorno de desarrollo integrado (IDE, es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios. Consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). (Netbeans, 2015)

Netbeans v8.0

Netbeans es un IDE que permite el desarrollo rápido y fácil de aplicaciones Java para escritorio, aplicaciones para móviles y aplicaciones web haciendo uso de las tecnologías HTML 5, javascript y CCS. Presenta una gran cantidad de herramientas para desarrolladores PHP y C++. Es un entorno de código abierto y libre que cuenta con una gran comunidad de desarrolladores y usuarios en todo el mundo. Posee un editor extensible lo que permite instalarle soporte para otros lenguajes mediante *plugins*. (Netbeans, 2015)

1.4.6 Biblioteca de clases

Una biblioteca de clases es un conjunto de clases de programación orientada a objetos. Esas clases contienen métodos que son útiles para los programadores. En el caso de Java cuando se descarga el JDK se obtiene la biblioteca de clases API. El uso de las clases y métodos de las APIs de Java reduce el tiempo de desarrollo de los programas. También, existen diversas bibliotecas de clases desarrollados por terceros que contienen componentes reutilizables de *software*. (Satman, 2014)

Rserve 1.8-1

Rserve es la implementación de cliente de servicios de fondo para REngine para acceder a R desde Java. Rserve actúa como un servidor de socket (TCP/IP o tomas de locales) que permite peticiones binarias a R. Cada conexión tiene un espacio de trabajo separado y directorio de trabajo. Las implementaciones del lado del cliente están disponibles para lenguajes populares como C/C++ y Java, permitiendo cualquier solicitud de uso de instalaciones de R sin la necesidad de vincular código R. Rserve soporta conexión remota, transferencia de archivos y autenticación de usuarios. (Urbanek, 2015) En SEEGEN-R se utiliza la biblioteca de clases Rserve 1.8-1 para hacer el llamado de las funciones de R, permitiendo la conexión del lenguaje Java a R.

Conclusiones parciales

En el desarrollo de este capítulo se abordaron conceptos fundamentales referentes a los estudios Biodemográficos en Cuba. Se analizaron los principales *software* existentes para el desarrollo de cálculos estadísticos vinculados con los estudios Biodemográficos, determinando que ninguno de los sistemas anteriores satisface las necesidades de los especialistas; evidenciándose la necesidad de realizar el componente de análisis Biodemográficos para SEEGEN-R. Además la metodología seleccionada para el desarrollo es RUP, establecida en la línea base de la arquitectura del sistema para la creación de nuevos componentes. Como lenguaje de modelado se adoptó UML 2.0 y como herramienta Visual Paradigm 8.0. Los lenguajes de programación seleccionados son R 3.2 y Java 1.6 utilizando como IDE Netbeans 8.0 e integrándose mediante la biblioteca de clases Rserve 1.8-1..

CAPÍTULO II: Análisis y diseño del componente de Análisis Biodemográficos para el sistema SEEGEN-R.

Introducción

En este capítulo se define lo que debe hacer el componente informático a través de la caracterización del negocio identificándose los actores, casos de uso del negocio y sus descripciones. Se exponen los procesos a informatizar identificándose los requisitos funcionales así como los requisitos no funcionales con los que debe cumplir el componente. Además se describen los actores y casos de uso que intervienen en el sistema a desarrollar. Por otra parte se realiza el diseño de los diagramas de clases del componente identificándose los patrones de diseño utilizados en el mismo.

2.1 Modelos de negocio

El modelo de negocio es la caracterización de la organización con el propósito de conocer la estructura de la organización, identificar los requisitos del sistema para conocer lo que debe hacer el nuevo sistema. (Tecnología, 2011)

2.1.1 Actores del negocio

Tabla. 1 Actores del negocio

Actor	Descripción
Consejo científico	Inicia la investigación y recibe los resultados de la misma.

2.1.2 Trabajadores del negocio

Tabla. 2 Trabajadores del negocio

Trabajador	Descripción
Especialista	Especialista en Genética encargado de realizar estudios Biodemográficos.

2.1.3 Diagrama de casos de uso del negocio (CUN)

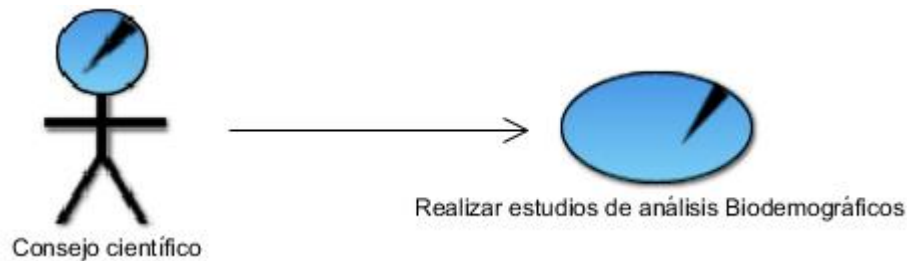


Fig. 1 Diagrama de CUN: Realizar estudios de análisis Biodemográficos

2.1.4 Descripción textual del caso de uso del negocio

Tabla. 3 Descripción textual del caso de uso del negocio

Caso de Uso del negocio	Realizar estudios de análisis Biodemográficos
Actores	Consejo científico (inicia)
Resumen	El caso de uso se inicia cuando el Consejo científico inicia la investigación. El especialista realiza los estudios Biodemográficos correspondientes, comunica los resultados al Consejo científico terminando así el caso de uso.
Acción del actor	Respuesta del proceso de negocio
1- El Consejo Científico inicia la investigación.	2- El especialista aplica instrumento de recolección de datos.
	3- El especialista realiza los estudios Biodemográficos.
	4- El especialista obtiene los resultados del estudio.
	5- El especialista informa al Consejo Científico los resultados.
6- El Consejo Científico recibe los resultados sobre los estudios Biodemográficos.	
Precondiciones	Deben estar cargados en el módulo base los datos necesarios para realizar el estudio.

2.1.5 Diagrama de actividades

Un diagrama de actividades es un proceso de negocio o un proceso de *software* que representa un flujo de trabajo a través de una serie de acciones. Estas acciones las pueden llevar a cabo personas, componentes de *software* o equipos. Utilizan símbolos con significados definidos que representan los pasos del algoritmo, y representan el flujo de ejecución mediante flechas que conectan los puntos de inicio y de fin de proceso. (Developer, 2015)

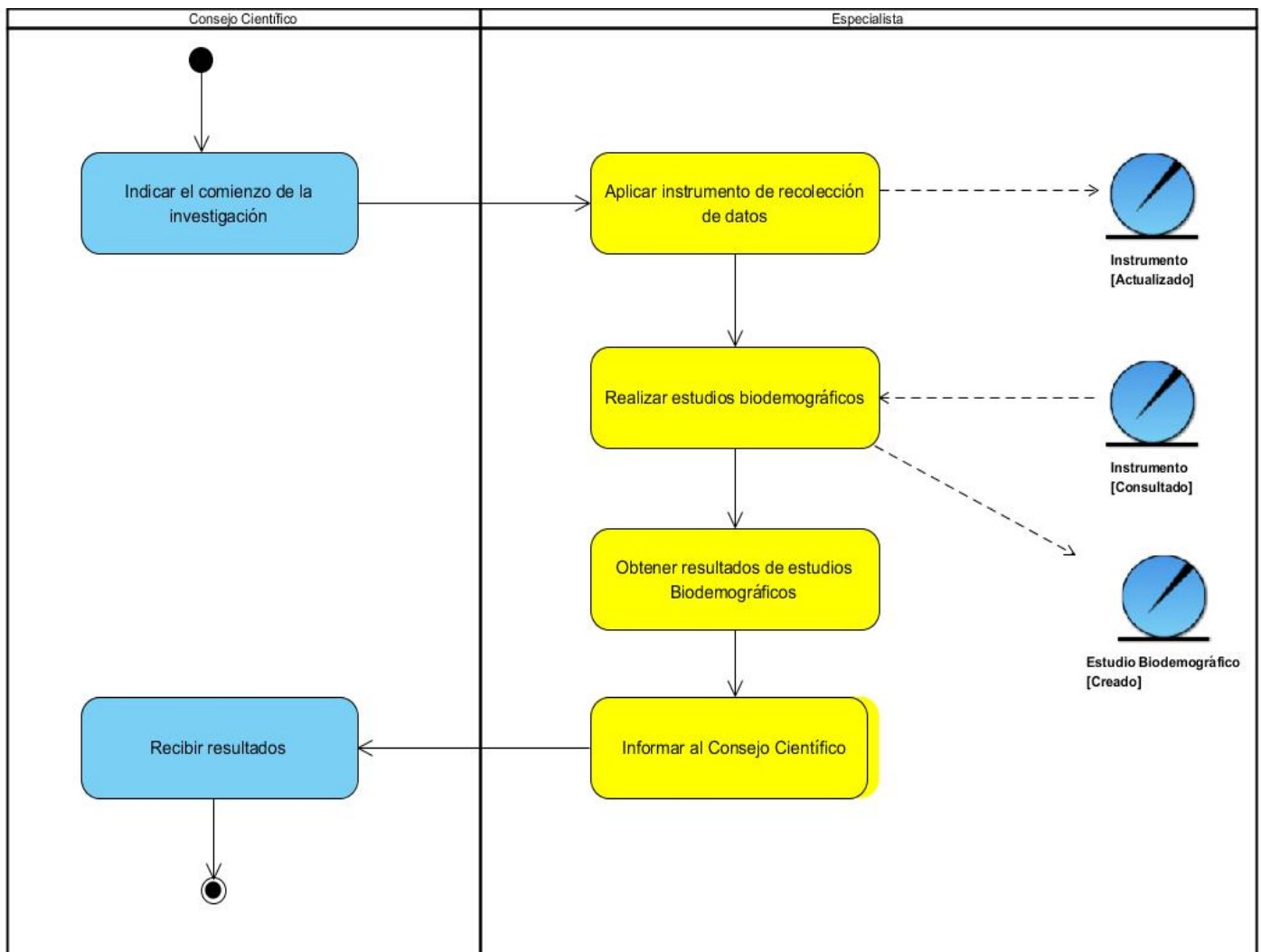


Fig. 2 Diagrama de actividades del negocio

2.1.6 Modelo de objetos del negocio

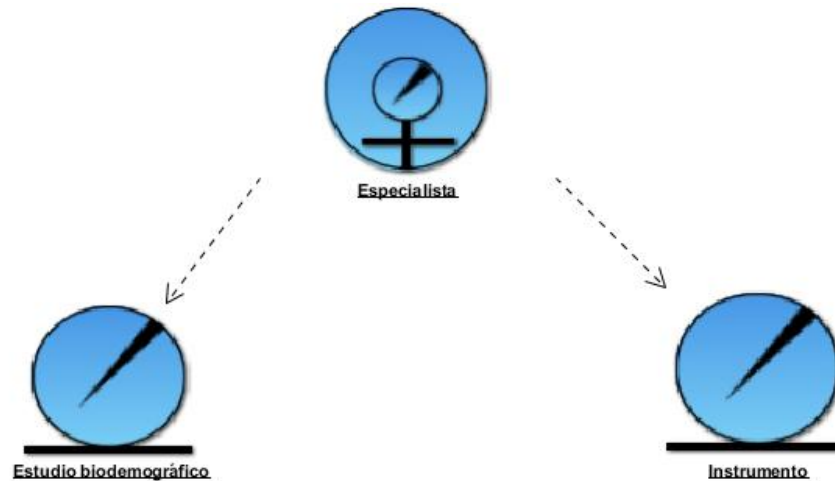


Fig. 3 Modelo de objetos del negocio

2.1.7 Reglas del negocio

Las reglas del negocio se basan en la extracción de información originada de las políticas, reglas y regulaciones del negocio de la descripción del flujo.

1. Solo el Consejo Científico puede indicar el inicio de una investigación.
2. El valor del parámetro n para el cálculo de n-potencia de una matriz debe ser un valor entero mayor que cero.
3. La longitud del vector efectivo de la población es de 12 elementos.
4. La matriz de migración estocástica columna, debe ser una matriz de 12 poblaciones.
5. La matriz de presión sistémica debe ser una matriz simétrica cuadrada de dimensiones 12 x12 donde los elementos fuera de la diagonal principal tienen valor cero.

2.2 Especificación de los requisitos del componente informático.

Para el modelado del sistema se realizó la captura y especificación de los requisitos funcionales y no funcionales con los que el mismo debe cumplir.

2.2.1 Requisitos Funcionales

Los requisitos funcionales representan servicios que proporcionan el sistema, cómo reacciona a entradas particulares y cómo se comporta en situaciones particulares. (Wieggers, 2003) En el sistema se identificaron 18 requisitos funcionales, los cuales se listan a continuación;

RF1: Crear juegos de datos

Estudios de tablas de frecuencia de apellidos: los estudios de tablas de frecuencia de apellidos abarcan los análisis de subdivisiones geográficas a partir de matrices de parentesco condicionales y un formulario de datos de apellidos. Por otra parte relaciona los cálculos de coeficiente de una matriz de parentesco estandarizada y de una matriz de parentesco cuadrada simétrica a partir de frecuencias de apellidos.

RF2: Calcular el coeficiente de parentesco estandarizado Hedrick.

RF3: Calcular el coeficiente de parentesco Lasker.

RF4: Calcular una matriz de parentesco condicional de datos isonimia.

RF5: Calcular una matriz de parentesco a 'priori' de datos isonimia.

RF6: Calcular una estimación objetiva de Isonimia aleatoria.

RF7: Calcular el coeficiente isonimia aleatoria imparcial.

RF8: Calcular el F_{st} de la matriz de parentesco condicional

Estudios sobre el modelo de migración Malecot: los estudios sobre el modelo de migración Malecot incluyen el cálculo de la generación asintótica para el modelo Malecot. Partiendo de este cálculo el estudio es capaz de calcular la matriz de parentesco respectiva. Por otra parte este modelo permite obtener la matriz estocástica columna simétrica.

RF9: Calcular la generación asintótica para el modelo Malecot.

RF10: Calcular una matriz de parentesco con el modelo migratorio Malecot.

RF11: Convertir una matriz estocástica columna asimétrica en una matriz estocástica columna simétrica.

Estudios sobre matrices: los estudios sobre matrices abarcan los cálculos para la obtención de matrices de parentesco condicional a partir de vectores efectivos de la población y matrices de parentesco cuadrada y simétrica. Por otra parte, permite el cálculo de la potencia n-ésima de una matriz y convierte una matriz de migración en una matriz estocástica.

RF12: Convertir una matriz de migración en una matriz estocástico columna.

RF13: Calcular una matriz de parentesco condicional a partir de una matriz de parentesco.

RF14: Calcular la potencia n-ésima de una matriz.

Estudios de conjuntos de datos de matrimonios en bruto: los estudios de datos de matrimonio en bruto abarcan relacionan los cálculos de coeficientes de pares repetidos y tablas de frecuencia de apellidos. Además, permite realizar cálculos de estados civiles de isonimia y coeficientes de consanguinidad.

RF15: Calcular coeficientes observados y pares al azar.

RF16: Calcular tablas de frecuencia apellidos.

RF17: Calcular los coeficientes de endogamia.

RF18: Calcular los coeficientes civil de isonimia.

2.2.2 Requisitos No Funcionales

Los requisitos no funcionales son restricciones de los servicios o funciones ofrecidas por el sistema. (Wieggers, 2003)

- **Apariencia e interfaz**

RNF1: Debe mostrar los resultados de los estudios en la misma posición del escritorio, posibilitando mayor rapidez en la ubicación de los resultados.

RNF2: Los elementos de las clases de presentación deben poseer ToolTips que muestren información de la acción que realizan.

- **Usabilidad**

RNF3: La aplicación informática debe garantizar un acceso rápido y fácil, contando con un menú que satisfaga las necesidades de los usuarios.

RNF4: La aplicación informática consta como máximo de tres niveles que le facilita al especialista ir directamente al tipo de estudio que desea realizar al dar un clic sobre una de las opciones del menú.

- **Software**

RNF5: Se requiere para el correcto funcionamiento de la aplicación el sistema operativo Windows o Linux.

RNF6: Máquina Virtual de Java versión 1.6 o superior.

RNF7: Entorno de R 3.2.0.

- **Hardware**

RNF8: Para el funcionamiento de la aplicación se necesitará: 256 Megabytes (MB) de memoria RAM como mínimo y 200 MB de capacidad en el disco duro.

- **Requisitos de licencia**

RNF9: La aplicación está basada en tecnologías y herramientas de licencia de *software* GNU/GPL.

2.3 Definición de los casos de uso del sistema

Los casos de uso son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un caso de uso especifica la secuencia de acciones que el sistema realiza interactuando con sus actores, incluyendo alternativas dentro de la secuencia de acciones. (IBM, 2015)

2.3.1 Actores del sistema

Tabla. 4 Actores del sistema

Actor	Descripción
Especialista	Especialista en Genética que interactúa con el sistema, es el encargado de

	realizar los estudios y cálculos estadísticos sobre Biodemografía.
--	--

2.3.2. Paquetes del sistema

El diagrama de paquetes del sistema muestra la división del sistema en agrupaciones lógicas. El paquete de análisis Biodemográficos utiliza el paquete SEEGEN-R Módulo Base, para la funcionalidad de **Cargar** un juego de datos, así como mostrar resultados para los casos de uso del sistema.



Fig. 4 Diagrama de paquetes del sistema

2.3.3. Casos de uso del sistema

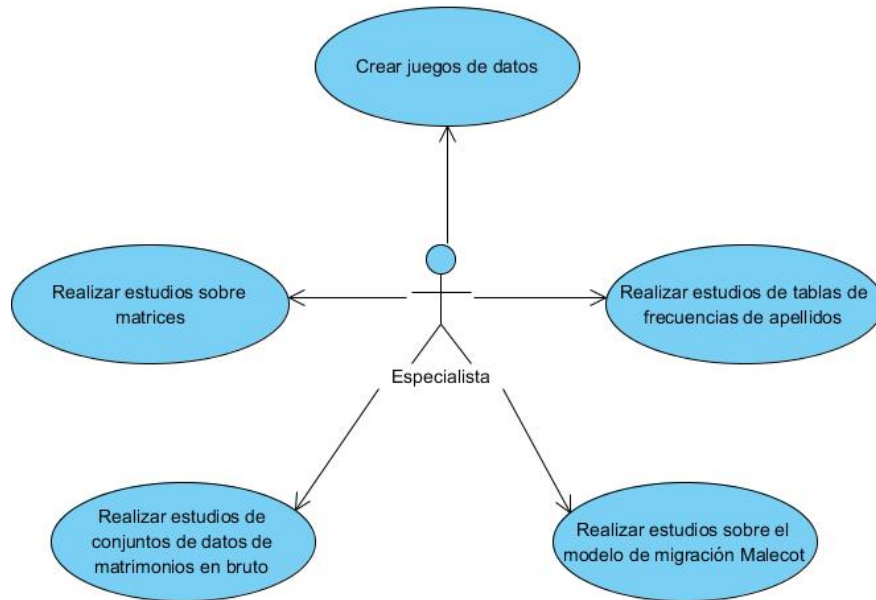


Fig. 5 Diagrama de CUS: Paquete de Estudios Biodemográficos

*CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS
BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R*

2.3.4 Descripción de casos de uso del sistema.

Tabla. 5 Descripción CUS: Realizar estudios de tablas de frecuencias de apellidos

Caso de Uso	Realizar estudios de tablas de frecuencias de apellidos	
Actores	Especialista	
Propósito	Este caso de uso se lleva a cabo con el objetivo de realizar estudios de tablas de frecuencias de apellidos asociados a los estudios Biodemográficos	
Resumen	El caso de uso inicia cuando el especialista desea realizar algunos de los estudios siguientes: calcular el coeficiente de parentesco estandarizado Hedrick, calcular el coeficiente de parentesco Lasker, calcular una matriz de parentesco condicional de datos isonimia, calcular una matriz de parentesco "a priori" de datos isonimia, calcular una estimación objetiva de Isonimia aleatoria, calcular el coeficiente isonimia aleatoria imparcial y calcular el Fst de la matriz de parentesco condicional. El caso de uso termina cuando han sido completados los estudios seleccionados.	
Complejidad	Alta	
Precondiciones	Deben estar cargados los datos del módulo base.	
Flujo Normal de Eventos		
Acciones del Actor	Respuesta del Sistema	
1. El especialista selecciona la opción <i>Paquetes</i> , luego <i>Biodem</i> en el Menú principal de SEEGEN-R. (Ver Interfaz 1.1)	2. El sistema muestra una interfaz Realizar estudios de tablas de frecuencia de apellidos (Ver Interfaz 1.2) que contiene: <ul style="list-style-type: none"> • El título del estudio. • La tabla de frecuencias con la que se realizará el estudio. • El vector efectivo de la población con la que se realizará el estudio. • El tipo de método a utilizarse en la realización del estudio. Si el especialista desea calcular el coeficiente de	

*CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS
BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R*

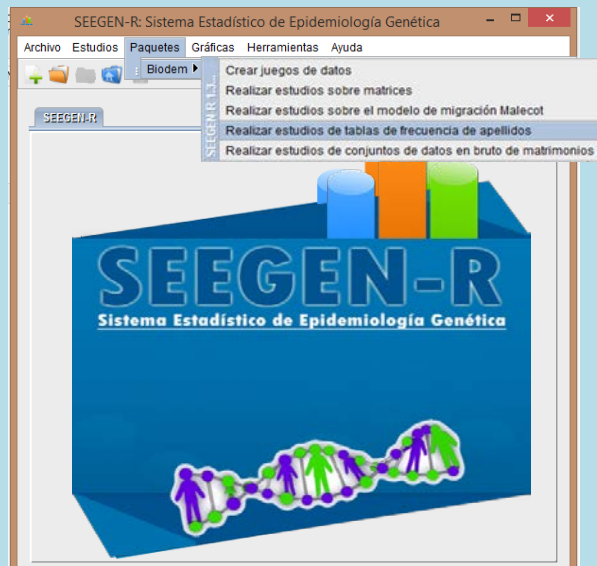
	<p>parentesco estandarizado Hedrick, ir a la sección “Calcular el coeficiente de parentesco estandarizado Hedrick.”.</p> <p>Si el especialista desea calcular el coeficiente de parentesco Lasker, ir a la sección “Calcular el coeficiente de parentesco Lasker”.</p> <p>Si el especialista desea calcular una matriz de parentesco condicional de datos isonimia, ir a la sección “Calcular una matriz de parentesco condicional de datos isonimia”.</p> <p>Si el especialista desea calcular una matriz de parentesco "a priori" de datos isonimia, ir a la sección “Calcular una matriz de parentesco "a priori" de datos isonimia.”</p> <p>Si el especialista desea calcular una estimación objetiva de Isonimia aleatoria, ir a la sección “Calcular una estimación objetiva de Isonimia regional aleatoria”</p> <p>Si el especialista desea calcular el coeficiente isonimia aleatoria imparcial, ir a la sección “Calcular el coeficiente isonimia aleatoria imparcial”</p> <p>Si el especialista desea calcular el Fst de la matriz de parentesco condicional, ir a la sección “Calcular el Fst de la matriz de parentesco condicional”</p>
--	--

Prototipo de Interfaz

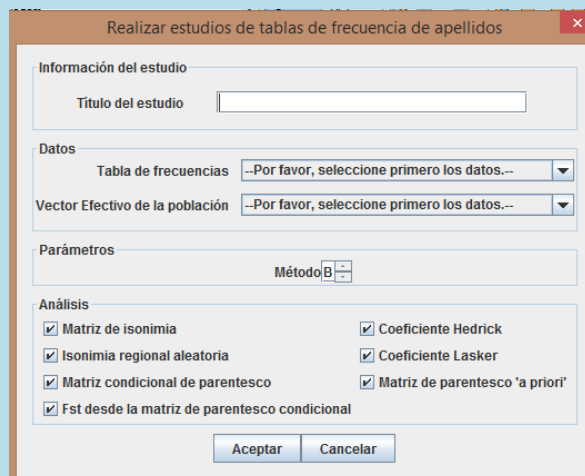


CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Interfaz 1.1



Interfaz 1.2



Sección “Calcular el coeficiente de parentesco estandarizado Hedrick.”

Acciones del Actor	Respuesta del Sistema
1. El especialista marca la opción <i>Calcular el coeficiente de parentesco estandarizado Hedrick</i> .	2. El sistema activa la selección para los siguientes parámetros: (Ver interfaz 1.3). <ul style="list-style-type: none"> • Una tabla de frecuencia de apellidos.

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

	<ul style="list-style-type: none"> Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio. (Ver interfaz 1.4)

Flujo Alternativo

3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.5). Se realiza el paso 3 de la Sección “Calcular el coeficiente de parentesco estandarizado Hedrick”.
---	---

Prototipo de Interfaz

Interfaz 1.3

Interfaz 1.4

```

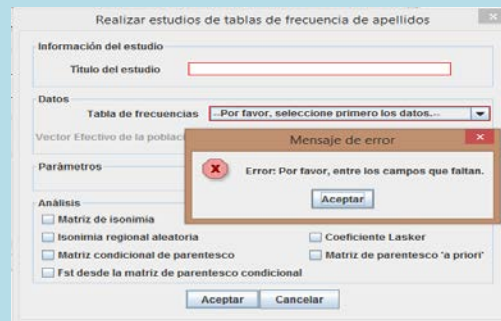
Resultado del estudio
Titulo: ejemplo
Autor: My Computer
Estudio: Realizar estudios de tablas de frecuencia de apellidos

Coeficiente Hedrick
      Due      Tre      Uno
Due 1.0000000 0.3606394 0.5545896
Tre 0.3606394 1.0000000 0.3812424
Uno 0.5545896 0.3812424 1.0000000

Análisis realizado en SEEGEN-R (Sistema Estadístico de Epidemiología Genética)
    
```

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Interfaz 1.5



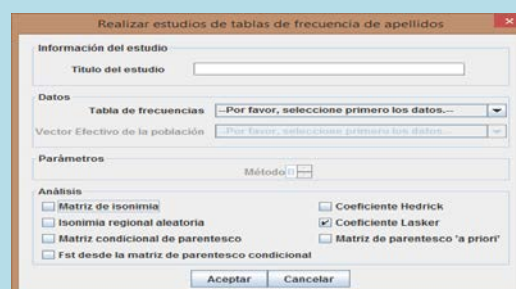
Sección “Calcular el coeficiente de parentesco Lasker”

Acciones del Actor	Respuesta del Sistema
1. El especialista marca la opción <i>Calcular el coeficiente de parentesco Lasker</i> .	2. El sistema activa la selección para los siguientes parámetros: (Ver interfaz 1.6). <ul style="list-style-type: none"> • Una tabla de frecuencia de apellidos. • Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio.(Ver interfaz 1.7)

Flujos Alternos

3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.8). Se realiza el paso 3 de la Sección “Calcular el coeficiente de parentesco Lasker”
---	---

Interfaz 1.6



CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Interfaz 1.7

```

Resultado del estudio
Titulo: ejemplo
Autor: My Computer
Estudio: Realizar estudios de tablas de frecuencia de apellidos

Coeficiente Lasker
      Due      Tre      Uno
Due 0.10997732 0.04166667 0.06390977
Tre 0.04166667 0.12109375 0.04605263
Uno 0.06390977 0.04605263 0.12049861

Análisis realizado en SEEGEN-R (Sistema Estadístico de Epidemiología Genética)
    
```

Interfaz 1.8

The screenshot shows a window titled "Realizar estudios de tablas de frecuencia de apellidos". It contains several sections: "Información del estudio" with a text field for "Titulo del estudio"; "Datos" with a dropdown menu for "Tabla de frecuencias" (currently showing "--Por favor, seleccione primero los datos.--"); "Vector Efectivo de la población"; "Parámetros"; and "Análisis" with several checkboxes: "Matriz de isonimia", "isonimia regional aleatoria", "Matriz condicional de parentesco", "Fat desde la matriz de parentesco condicional", "Coeficiente Lasker" (checked), and "Matriz de parentesco 'a priori'". A modal error message box is overlaid on the form, titled "Mensaje de error", with the text "Error: Por favor, entre los campos que fallan." and an "Aceptar" button.

Sección "Calcular una matriz de parentesco condicional de datos isonimia"

Acciones del Actor	Respuesta del Sistema
1. El especialista marca la opción <i>Calcular una matriz de parentesco condicional de datos isonimia</i> .	2. El sistema activa la selección para los siguientes parámetros. (Ver interfaz 1.9) <ul style="list-style-type: none"> • Método. • Una tabla de frecuencia de apellidos. • Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio. (Ver interfaz 1.10)

Flujos Alternos

3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.11). Se realiza el paso 3 de la Sección "Calcular una matriz de parentesco
---	--

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

condicional de datos isonimia”

Interfaz 1.9

Interfaz 1.10

```

Resultado del estudio
Título: ejemplo
Autor: My Computer
Estudio: Realizar estudios de tablas de frecuencia de apellidos

Matriz condicional de parentesco
      [,1]      [,2]      [,3]
[1,] 0.0143246630 -0.01376432 -0.009637322
[2,] -0.0137643249 0.01740782 -0.0112402648
[3,] -0.009637322 -0.01124026 0.0194691387

Análisis realizado en SEEGEN-R (Sistema Estadístico de Epidemiología Genética)
    
```

Interfaz 1.11

Sección “Calcular una matriz de parentesco ‘a priori’ de datos isonimia”

Acciones del Actor	Respuesta del Sistema
<p>1. El especialista marca la <i>Calcular una matriz de parentesco ‘a priori’ de datos isonimia</i>.</p>	<p>2. El sistema activa la selección para los siguientes parámetros: (Ver interfaz 1.12).</p> <ul style="list-style-type: none"> • Una tabla de frecuencia de apellidos. • Método.

*CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS
BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R*

	<ul style="list-style-type: none"> Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio.(Ver interfaz 1.13)

Flujos Alternos

3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.14). Se realiza el paso 3 de la Sección “Calcular una matriz de parentesco ‘a priori’ de datos isonimia”
---	--

Interfaz 1.12

Interfaz 1.13

```

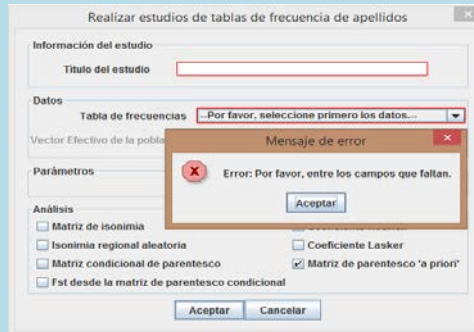
Resultado del estudio
-----
Titulo: ejemplo
Autor: My Computer
Estudio: Realizar estudios de tablas de frecuencia de apellidos

Matriz de parentesco 'a priori'
      [,1]      [,2]      [,3]
[1,] 0.04664733 0.01947925 0.03186008
[2,] 0.01947925 0.04962919 0.02192054
[3,] 0.03186008 0.02192054 0.05162291

Análisis realizado en SEEGEN-R (Sistema Estadístico de Epidemiología Genética)
    
```

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Interfaz 1.14



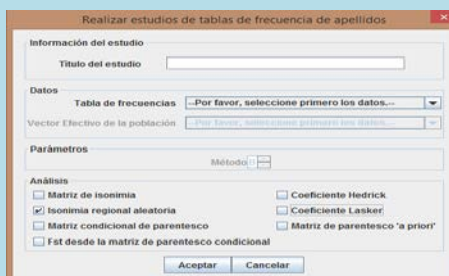
Sección “Calcular una estimación objetiva de Isonimia regional aleatoria”

Acciones del Actor	Respuesta del Sistema
1. El especialista marca la opción <i>Calcular una estimación objetiva de Isonimia regional aleatoria</i>	2. El sistema activa la selección para los siguientes parámetros: (Ver interfaz 1.15). <ul style="list-style-type: none"> • Tabla de frecuencia de apellidos • Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio. (Ver interfaz 1.16)

Flujos Alternos

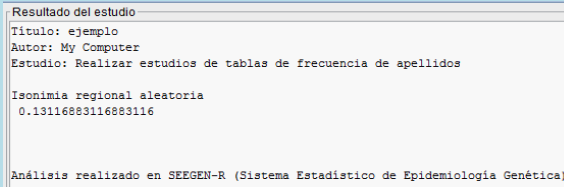
3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.17). Se realiza el paso 3 de la Sección “Calcular una estimación objetiva de Isonimia regional aleatoria”
---	---

Interfaz 1.15

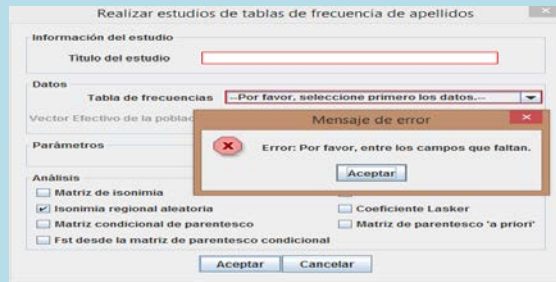


CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Interfaz 1.16



Interfaz 1.17



Sección “Calcular el coeficiente isonimia aleatoria imparcial”

Acciones del Actor	Respuesta del Sistema
1. El especialista marca la opción <i>Calcular el coeficiente isonimia aleatoria imparcial</i>	2. El sistema activa la selección para los siguientes parámetros: (Ver Interfaz 1.18) <ul style="list-style-type: none"> • Tabla de frecuencia de apellidos • Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio. (Ver Interfaz 1.19)

Flujos Alternos

3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.20). Se realiza el paso 3 de la Sección “Calcular el coeficiente isonimia aleatoria imparcial”
---	--

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Interfaz 1.18

Realizar estudios de tablas de frecuencia de apellidos

Información del estudio
Título del estudio:

Datos
Tabla de frecuencias:
Vector Efectivo de la población:

Parámetros
Método:

Análisis
 Matriz de isonimia
 Isonimia regional aleatoria
 Matriz condicional de parentesco
 Fat desde la matriz de parentesco condicional
 Coeficiente Hedrick
 Coeficiente Lasker
 Matriz de parentesco 'a priori'

Interfaz 1.19

```

Resultado del estudio
Titulo: ejemplo
Autor: My Computer
Estudio: Realizar estudios de tablas de frecuencia de apellidos

Matriz de isonimia
      [,1] [,2] [,3]
[1,] 0.18095235 0.08333333 0.12781955
[2,] 0.08333333 0.19166667 0.09210526
[3,] 0.12781955 0.09210526 0.19883041

Análisis realizado en SEEGEN-R (Sistema Estadístico de Epidemiología Genética)
    
```

Interfaz 1.20

Realizar estudios de tablas de frecuencia de apellidos

Información del estudio
Título del estudio:

Datos
Tabla de frecuencias:
Vector Efectivo de la población:

Parámetros
Método:

Análisis
 Matriz de isonimia
 Isonimia regional aleatoria
 Matriz condicional de parentesco
 Fat desde la matriz de parentesco condicional
 Coeficiente Hedrick
 Coeficiente Lasker
 Matriz de parentesco 'a priori'

Mensaje de error

Error: Por favor, entre los campos que faltan.

Sección “Calcular el Fst de la matriz de parentesco condicional”

Acciones del Actor	Respuesta del Sistema
<p>1. El especialista marca la opción <i>Calcular el Fst de la matriz de parentesco condicional</i>.</p>	<p>2. El sistema activa la selección para los siguientes parámetros: (Ver interfaz 1.21).</p> <ul style="list-style-type: none"> • Vector efectivo de la población.

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

	<ul style="list-style-type: none"> Las opciones de Aceptar y Cancelar.
3. El especialista llena los campos correspondientes al tipo de estudio solicitado y acepta el estudio.	4. El caso de uso termina cuando el sistema muestra los resultados del estudio. (Ver Interfaz 1.22)

Flujos Alternos

3.1 El especialista no introduce alguno de los datos necesarios para el análisis.	3.2 El sistema indica que el especialista debe llenar los campos necesarios para completar el estudio (Ver interfaz 1.23). Se realiza el paso 3 de la Sección “Calcular el Fst de la matriz de parentesco condicional”
---	--

Interfaz 1.21

Realizar estudios de tablas de frecuencia de apellidos

Información del estudio
Título del estudio:

Datos
Tabla de frecuencias: --Por favor, seleccione primero los datos--
Vector Efectivo de la población: --Por favor, seleccione primero los datos--

Parámetros
Método: B

Análisis
 Matriz de isonimia
 Isonimia regional aleatoria
 Matriz condicional de parentesco
 Fst desde la matriz de parentesco condicional
 Coeficiente Hedrick
 Coeficiente Lasker
 Matriz de parentesco 'a priori'

Aceptar Cancelar

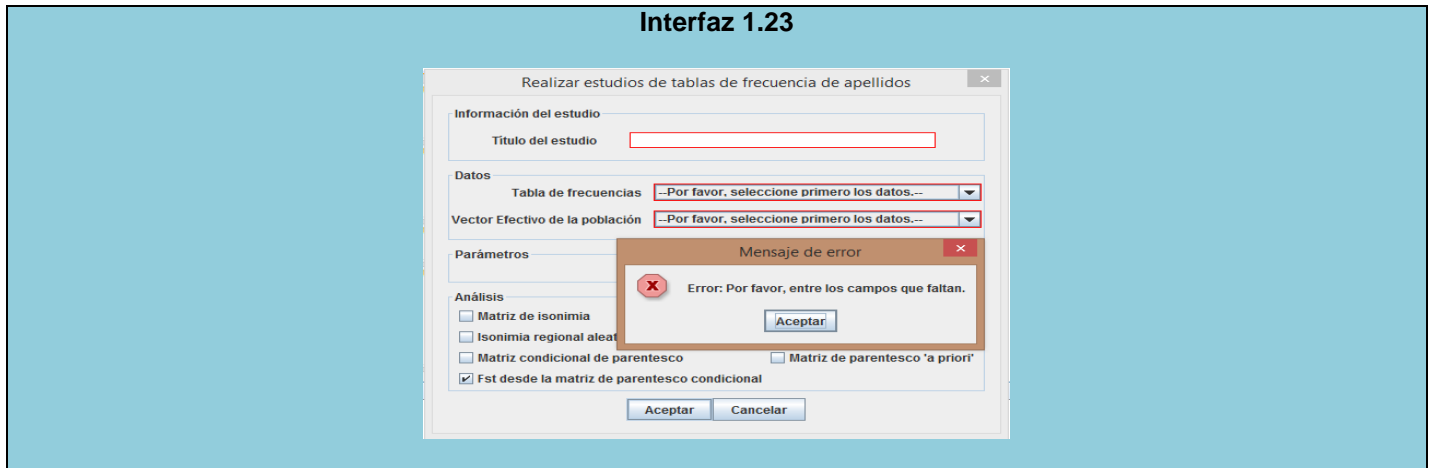
Interfaz 1.22

```

Resultado del estudio
-----
Titulo: ejemplo
Autor: My Computer
Estudio: Realizar estudios de tablas de frecuencia de apellidos

Fst desde la matriz de parentesco condicional
0.01522696

Análisis realizado en SEEGEN-R (Sistema Estadístico de Epidemiología Genética)
    
```



2.4 Patrones arquitectónicos

Los patrones arquitectónicos definen la estructura de un sistema, los cuales a su vez se componen de subsistemas con sus responsabilidades. Poseen varias directivas para organizar los componentes del sistema. Facilita las tareas de diseño de un sistema. Proporciona un conjunto de sub-sistemas predefinidos, especificando sus responsabilidades, reglas y directrices que determinan la organización, comunicación, interacción y relaciones entre ellos. (Gamma, 2002)

2.4.1 Arquitectura en *N* capas

La programación por capas es una arquitectura en la cual el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño. En toda arquitectura de capas los elementos agrupados en una misma capa pueden comunicarse entre sí. (Gamma, 2002)

➤ **Arquitectura en tres capas**

En la arquitectura en 3 capas se define cómo organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, lo cual quiere decir que los componentes de una capa sólo pueden hacer referencia a componentes en capas inmediatamente inferiores. Simplifica la comprensión y la organización del desarrollo de sistemas complejos. (Gamma, 2002)

- **Capa de presentación**

Es la encargada de presentarle el sistema al usuario, comunicar y capturar la información del usuario (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. (Gamma, 2002)

- **Capa de negocio**

Esta capa interactúa con la capa de presentación para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al manejador de base de datos que realice una operación de almacenamiento, edición, eliminación, consulta de datos u otra. (Gamma, 2002)

- **Capa de datos**

Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio. (Gamma, 2002)

2.4.2. Patrones arquitectónicos aplicados

El componente desarrollado presenta internamente una arquitectura N capas. Según las características de la aplicación se definen solo dos capas, la capa de presentación y la capa de negocio, no identificándose la capa de datos pues para el componente no es necesaria la capa de acceso a datos ya que no se requiere de la persistencia de datos.

2.4.3 Vista lógica del sistema

Muestra el diseño de la funcionalidad del sistema en sus dos aspectos esenciales: los componentes que lo integran y su comportamiento, expresado en términos de la dinámica de interacción de dichos componentes. (IBM, 2015) En la figura 6 se muestra la vista lógica del diseño.

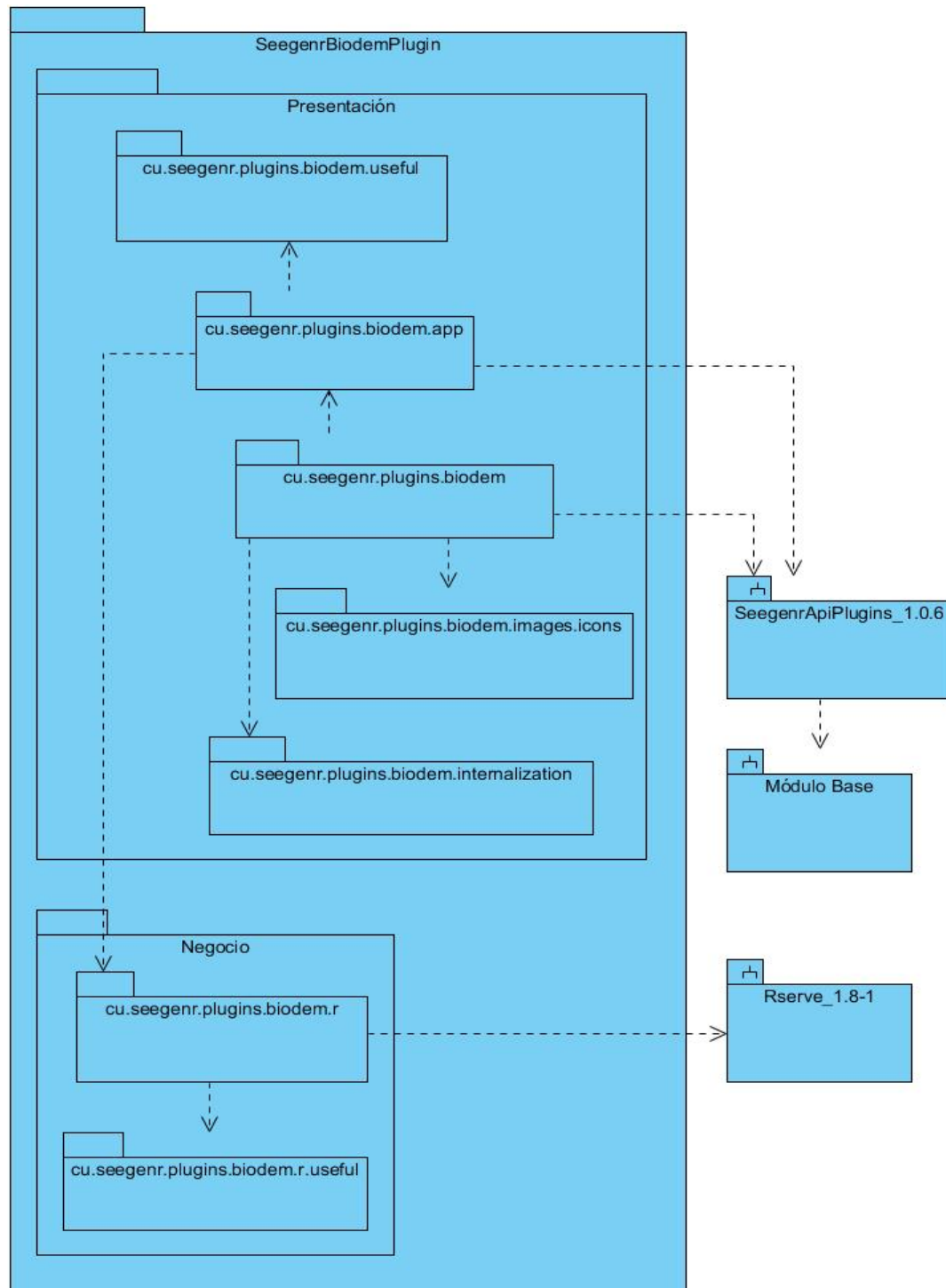


Fig. 6 Vista lógica del sistema

CAPÍTULO II: ANÁLISIS Y DISEÑO DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

En la vista lógica del sistema se presentan las dos capas del sistema en relación con la arquitectura empleada.

Presentación: la capa de presentación contiene las clases de presentación que serán visualizadas por los especialistas. Se detallan cada uno de los paquetes en los cuales se separaron.

cu.seegenr.plugin.biodem: contiene la clase que crea el menú principal del componente.

cu.seegenr.plugin.biodem.app: contiene las clases de presentación que serán visualizadas por los especialistas.

cu.seegenr.plugin.biodem.internacionalizacion: es un paquete auxiliar usado para la internacionalización del sistema.

cu.seegenr.plugin.biodem.useful: es un paquete que contiene las clases auxiliares de las clases de presentación necesarias para realizar los estudios Biodemográficos.

Negocio: la capa de negocio se comunica con la capa de presentación, ejecutando la lógica de la aplicación. Se detallan cada uno de los paquetes en los cuales fue separado.

cu.seegenr.biodem.r: contiene las clases de ejecución de código R para cada estudio en relación con la interfaz que le corresponde.

cu.seegenr.biodem.r.useful: es un paquete que contiene las clases auxiliares de las clases de ejecución de código R.

Bibliotecas de clases: las bibliotecas de clases son las encargadas de realizar la comunicación del lenguaje Java con el lenguaje R.

SeegenrApiPlugins_1.0.6: es la biblioteca de clases encargada de comunicar el componente con el sistema SEEGEN-R a través del Módulo Base. Mediante esta biblioteca se cargan los datos necesarios del módulo base para realizar los diferentes estudios.

Rserve_1.8-1: es la biblioteca de clases encargada de conectar el lenguaje de programación R con el lenguaje Java.

2.5 Diagrama de clases del diseño

El diagrama de clases del diseño representa las relaciones entre clases que mantienen la información manipulada por el sistema, representan además la interacción de las clases del diseño y sus objetos en la realización de los casos de uso de forma que de cada caso de uso se desprende un diagrama de clases. (PRESSMAN, 2002) En la figura se muestra el diagrama de clases evidenciándose las relaciones de dependencia, agregación y composición.

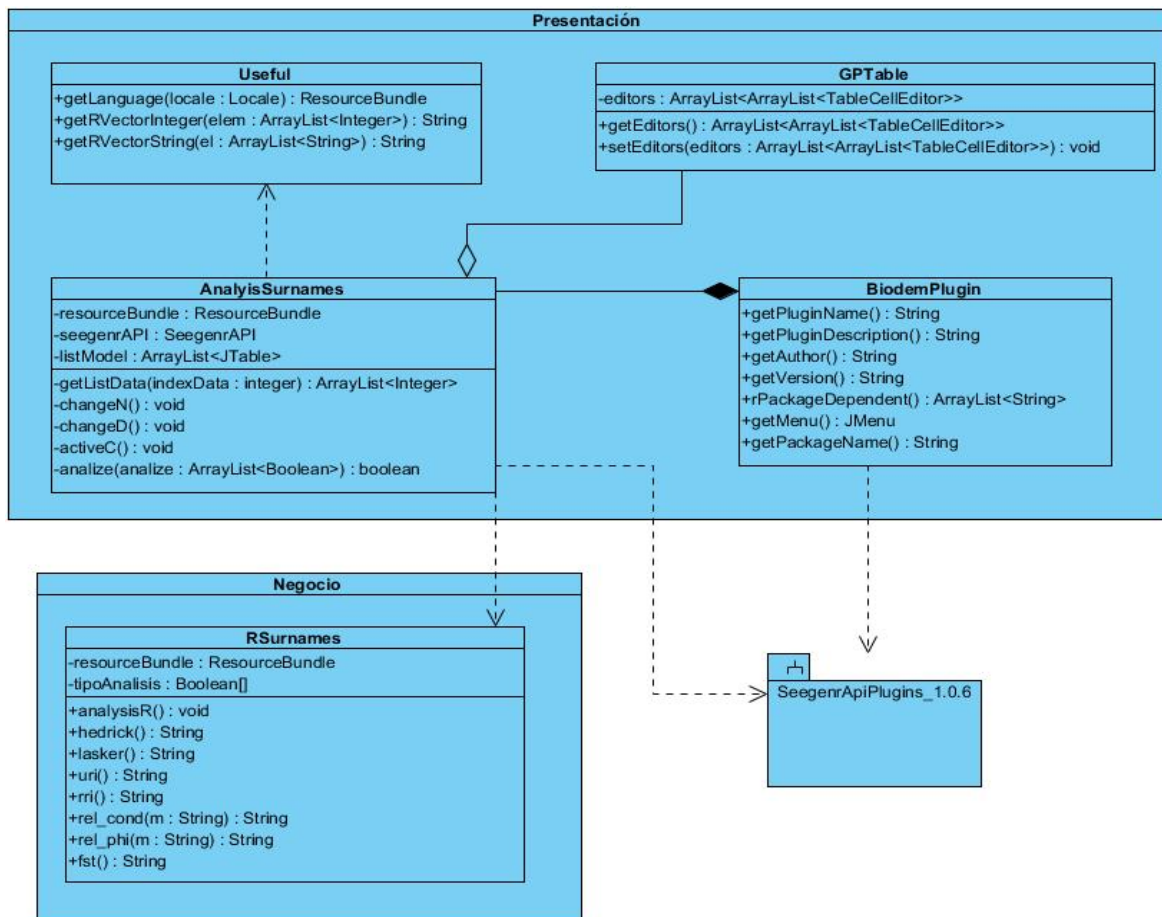


Fig. 7 Diagrama de clases del diseño

La clase *BiodemPlugin* es la encargada de crear un menú con todas las vistas. El *SeegenAPIPlugin* captura el menú creado y lo muestra en el menú del módulo base. A partir de este menú se ejecuta la vista que se seleccione, mostrando los textos en el idioma correspondiente obtenido de la clase *Useful* y se cargan los datos abiertos del módulo base. Al completar la petición de la vista, los datos y funciones

seleccionadas por el especialista son enviados a la clase de abstracción de código R para que realice el estudio. Al finalizar la ejecución del código, los resultados son enviados al Módulo Base a través de SeegenAPIPlugin, el cual es el encargado de mostrárselo al usuario.

2.6 Descripción de las clases del diseño.

A continuación se muestran las tablas de descripción de las clases del diseño (DCD) que intervienen en el caso de uso que se está analizando.

Tabla. 6 DCD: AnalysisSurnames.java

Nombre: AnalysisSurnames	
Tipo de clase: Interfaz de usuario	
Responsabilidades: Interfaz de usuario asociada al CU: Realizar estudios de tablas de frecuencia de apellidos	
Nombre:	GetListData(int indexData)
Descripción:	Obtiene del módulo base los datos que están cargados.
Nombre:	CreateTable()
Descripción:	Crea una tabla con la lista de datos obtenidos del GetListData(), la que será devuelta al Módulo Base el cual es el encargado de mostrárselo al usuario.
Nombre:	analize(ArrayList<Boolean> analize)
Descripción:	Es el encargado de indicar a la clase de abstracción de código R el estudio que desea realizar el especialista.

Tabla. 7 DCD: RSurnames.java

Nombre: RSurnames	
Tipo de clase: Abstracción	
Responsabilidades: Clase de abstracción de código R necesaria para realizar estudios de tablas de frecuencias de	

apellidos	
Nombre:	analysisR(String m)
Descripción:	Encargado de indicar cuál es el estudio que desea realizar el especialista
Nombre:	fst()
Descripción:	Realiza el cálculo del Fst de la matriz de la matriz de parentesco condicional
Nombre:	hedrick()
Descripción:	Realiza el cálculo del coeficiente estandarizado hedrick
Nombre:	lasker()
Descripción:	Realiza el cálculo del coeficiente estandarizado lasker.
Nombre:	uri()
Descripción:	Realiza el cálculo de la matriz de isonimia
Nombre:	rri()
Descripción:	Realiza el cálculo del valor de la isonimia regional aleatoria
Nombre:	rel_cond(String m)
Descripción:	Realiza el cálculo de la matriz condicional de parentesco
Nombre:	rel_phi(String m)
Descripción:	Realiza el cálculo de la matriz de parentesco 'a priori'

2.7 Diagrama de secuencias

Representa la secuencia de mensajes entre las instancias de clases, componentes, subsistemas o actores. El tiempo fluye hacia abajo en el diagrama y muestra el flujo de control de un participante a otro. Contienen detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el

escenario y mensajes intercambiados entre ellos. Estos diagramas destacan el orden temporal de los mensajes. (PRESSMAN, 2002)

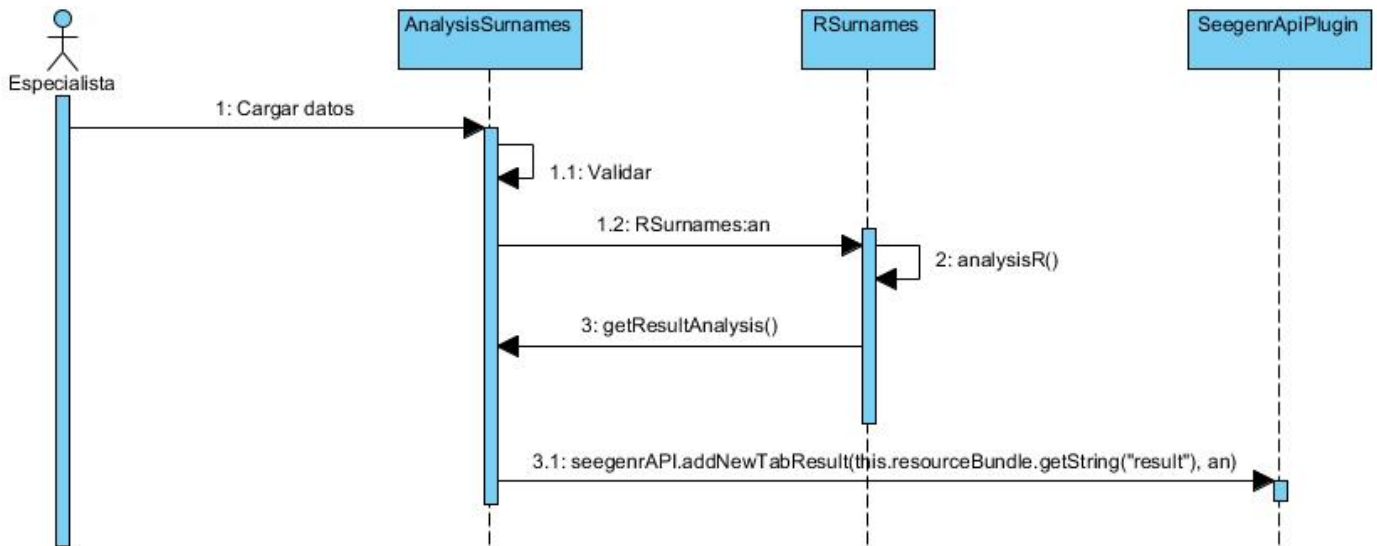


Fig. 8 Diagrama de secuencia: CUS Realizar estudios de tablas de frecuencias de apellidos

2.8 Patrones de diseño

Los patrones de diseño brindan una solución ya probada y documentada a problemas de desarrollo de *software* que están sujetos a contextos similares. Los patrones están concebidos como diseños orientados a objetos, donde se identifican clases e instancias, sus roles y colaboraciones y se describe la distribución de responsabilidades. (PRESSMAN, 2002)

2.8.1 Patrones GRASP

Dentro de los patrones GRASP más utilizados se encuentran los siguientes: Experto, Creador, Alta Cohesión y Bajo Acoplamiento pues se refieren a cuestiones muy básicas, comunes y a aspectos fundamentales del diseño.

- **Patrones GRASP aplicados**

Experto: La intencionalidad de este patrón es asignar una responsabilidad a la clase que contiene la información necesaria para cumplirla. (PRESSMAN, 2002) En el sistema se evidencia este patrón en las

clases de presentación, las cuales son las clases expertas en presentar los estudios al usuario y capturar los parámetros necesarios para cada estudio.

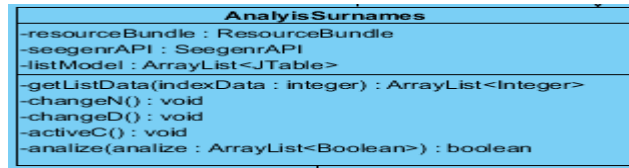


Fig. 9 Ejemplo del patrón Experto

Bajo acoplamiento: Es un patrón evaluativo pues permite que el diseño de clases sea más independiente. Reduce el impacto de los cambios y aumenta la reutilización. Es la medida en cuanto una clase depende de la otra. (PRESSMAN, 2002) Este patrón se pone en práctica en la clase RSPN donde se los métodos necesarios para realizar los estudios sobre el modelo de migración Malecot se encuentran en esta clase, por lo que depende lo menos posible de otras clases para ejecutarse.

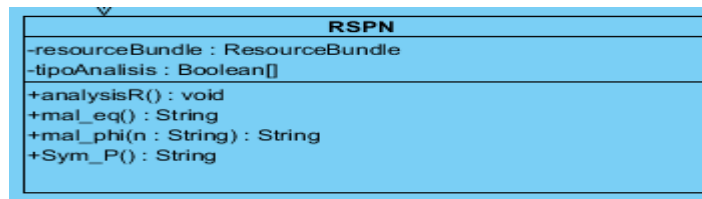


Fig. 10 Ejemplo del patrón Bajo Acoplamiento

Creador: El patrón creador guía la asignación de responsabilidades relacionadas con la creación de los objetos, tarea muy frecuente en los sistemas orientados a expertos. El propósito fundamental de este patrón es encontrar un creador para conectar con el objeto producido en cualquier evento. Al escoger el patrón de creador se da soporte al bajo acoplamiento. (PRESSMAN, 2002) En el sistema se evidencia en la clase interfaz AnalysisSurnames, la cual crea la clase RSurnames para ejecutar el código R referente a esa clase interfaz.

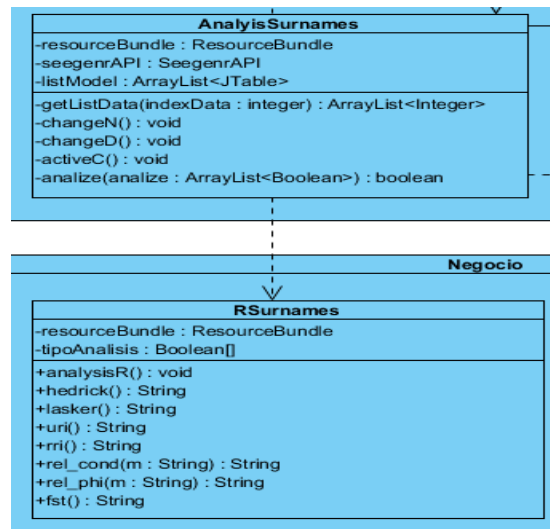


Fig. 11 Ejemplo del patrón Creador

Alta cohesión: Es un patrón evaluativo, entre más alta cohesión, más fácil de entender, cambiar y reutilizar. Mejoran la calidad y facilidad con que se entiende el diseño, se simplifica el mantenimiento y las mejoras de funcionalidad. (PRESSMAN, 2002) Este patrón se pone en práctica en la clase RUseful, ya que todas las clases de ejecución de código R dependen de ella para organizar los datos del lenguaje R.

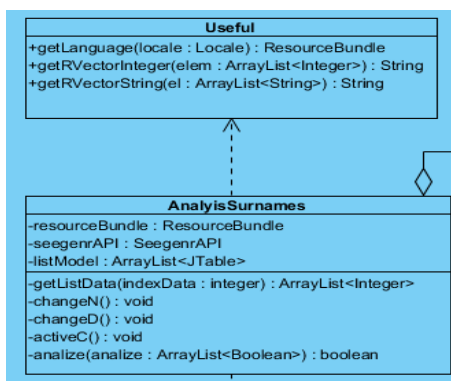


Fig. 12 Ejemplo del patrón Alta Cohesión

2.8.2 Patrones GoF

Los patrones GoF son la base para la búsqueda de soluciones concretas a problemas comunes en el desarrollo de *software* y otros ámbitos referentes al diseño de interacción o interfaces. Indican resoluciones

técnicas basadas en la Programación Orientada a Objetos y ayudan a construir *software* basado en la reutilización de código.

- **Patrones GOF aplicados**

Singleton: Garantiza que una clase solo tenga una instancia y proporciona un punto global de acceso a ella. (Welicki, 2015) Este patrón se aplica a la clase SeegenAPIPlugin, de la cual se utiliza una instancia que proporciona el Módulo Base.

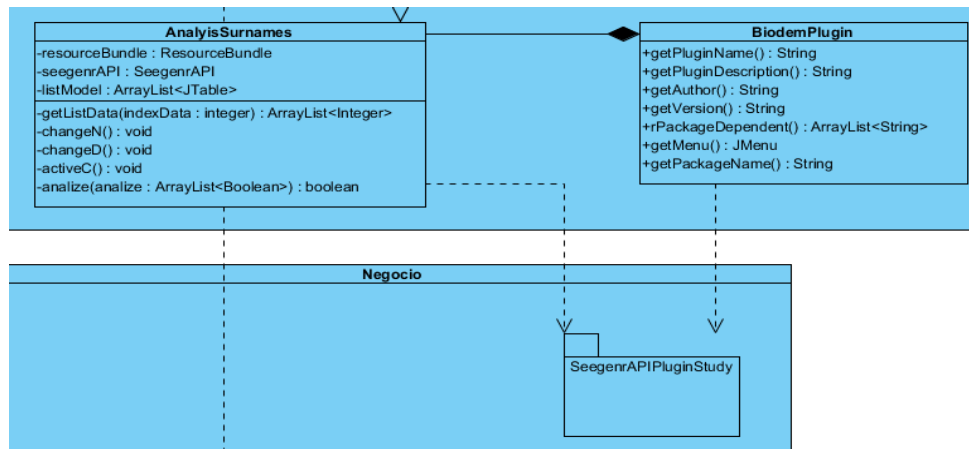


Fig. 13 Ejemplo del patrón Singleton

Conclusiones parciales

En este capítulo se definió el modelo de negocio con el objetivo de conocer la estructura de la organización para conocer las funcionalidades del nuevo sistema, identificándose 18 RF agrupados en 5 casos de uso del sistema y 9-RNF. Para representar el flujo de acciones entre el especialista y el consejo científico se diseñó el diagrama de actividades. Se evidenciaron las clases del diseño y las relaciones existentes entre ellas mediante los diagramas de clases del diseño haciendo uso de los patrones arquitectónicos y de diseño. Se realizó la vista lógica del sistema quedando reflejada la arquitectura N capas, definiéndose las capas de negocio y presentación. Para representa la secuencia de mensajes entre las instancias de clases, componentes, subsistemas y actores se diseñaron los diagramas de secuencia del sistema

CAPÍTULO III: Implementación y pruebas del componente de Análisis Biodemográficos para el sistema SEEGEN-R.

Introducción

En el presente capítulo se desarrolla el modelo de implementación. Para ello se muestra el diagrama de componentes mostrando la organización del sistema, así como fragmentos de código de las clases *Rurnames.java* y *AnalysisSurnames.java* implementadas utilizándose el estándar de codificación definido. Además se muestran los resultados al aplicar las pruebas de sistema e integración mostrando la calidad del componente.

3.1 Diagrama de componentes

El diagrama de componentes muestra la organización y dependencias lógicas entre componentes de *software*, sean estos ficheros de código fuente, binarios o ejecutables. Los elementos de modelado que lo conforman son los componentes y paquetes mostrando la estructura del sistema en términos de implementación a un alto nivel. (PRESSMAN, 2005) En la figura que se muestra a continuación se evidencia el diagrama de componentes, para un mayor entendimiento del sistema se relacionan y explican cada uno de los componentes que interactúan.

cu.seegenr.plugins.biodem: contiene la clase que crea el menú principal del componente.

cu.seegenr.plugins.biodem.app: contiene las clases de presentación que serán visualizadas por los especialistas.

cu.seegenr.plugins.biodem.internalization: es un paquete auxiliar usado para la internacionalización del sistema.

cu.seegenr.plugins.biodem.useful: este paquete contiene las clases auxiliares de la clase *RUseful.java* de ejecución de código R.

cu.seegenr.plugins.biodem.r: este paquete contiene las clases de abstracción del paquete R necesarias para cada una de las clases de presentación del estudio.

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

cu.seegenr.biodem.r.useful: es un paquete que contiene las clases auxiliares de las clases de ejecución de código R.

SeegenrApiPlugin 1.0.6: es la biblioteca encargada de conectar al Módulo Base de la aplicación con la solución propuesta.

Rserve 1.8-1: es la biblioteca de clases encargada de conectar el lenguaje de programación R con el lenguaje Java.

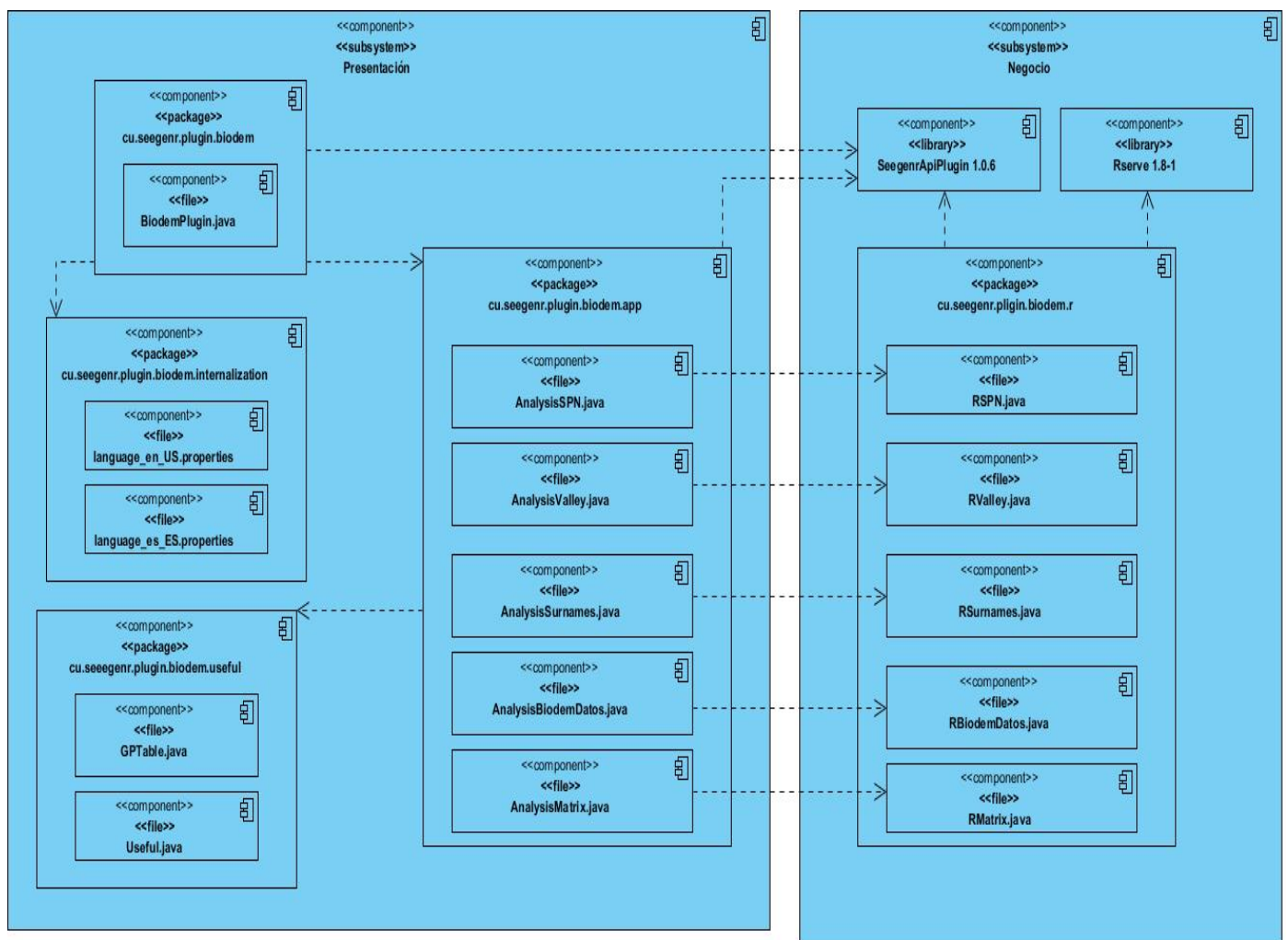


Fig. 14 Diagrama de componentes

3.2 Estándares de codificación

Los estándares de codificación describen un conjunto de reglas que los programadores deben seguir para escribir el código fuente de un sistema. Este debe ser escrito de la misma forma por todos los desarrolladores permitiendo mayor flexibilidad y organización en el trabajo. Mejoran la lectura del *software* permitiendo entender el código con más facilidad y agilidad.

- Los nombres de las clases deben ser sustantivos, cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúsculas. Intentar mantener los nombres de las clases simples y descriptivas.
- En los comentarios de las clases debe aparecer el autor de esta, el objetivo de la misma.
- Los nombres de las variables y métodos tienen que ser lo más descriptivos posibles, procurando que sean palabras en minúsculas con significado claro. Si necesita más de una palabra, deben ponerse juntas, poniendo la inicial de cada palabra en mayúscula siempre que no sea la primera, manteniéndolas tan breve como sea posible.
- Inicializar las variables locales donde se declaran. La única razón para no inicializar una variable donde se declara es si el valor inicial depende de algunos cálculos que deben ocurrir.
- Utilizar nombres en plural para arreglos, listas o matrices de objetos.
- Usar paréntesis en expresiones que implican distintos operadores para evitar problemas con el orden de precedencia de los operadores. Incluso si parece claro el orden de precedencia de los operadores, podría no ser así para otros, no se debe asumir que otros programadores conozcan el orden de precedencia.
- Las líneas en blanco mejoran la facilidad de lectura separando secciones de código que están lógicamente relacionadas por lo que se debe usar siempre una línea en blanco entre métodos.

3.3 Fragmentos de código

A continuación se muestran fragmentos de código fuente en R que pertenecen a varios componentes de paquetes representados a partir de la relevancia de estos en el módulo. También se muestra fragmentos de código java en el cual se realizan las funciones que interactúan con la clase *AnalysisSurnames.java* perteneciente a la clase *RSurnames.java*.

3.3.1 Fragmento de código R

La función **Fst** que se muestra calcula la medida de la heterogeneidad Genética entre subdivisión, a partir de **rval** (matriz de parentesco condicional) y **N** (vector de tamaño efectivo de la población) retornando un valor numérico. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **Fst** del paquete **Biodem**.

```
> library(Biodem)
Warning message:
package 'Biodem' was built under R version 3.1.3
> data(valley);
> tot <- sur.freq(valley, PAR, SURM, SURF);
> iso.matrix <- uri(tot);
> reg <- rri(tot);
> kin.cond <- rel.cond(iso.matrix, reg);
> N <- colSums(tot);
> fst<-Fst(kin.cond,N);
> fst;
           [,1]
[1,] 0.01512128
> |
```

Fig. 15 Fragmentos de código R de la función Fst

La función **hedrick** calcula el coeficiente estandarizado de parentesco hedrick, a partir de **x** (tabla de frecuencia de apellidos); retornando una matriz de parentesco estandarizada simétrica cuadrada. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **hedrick** del paquete **Biodem**.

```
> library(Biodem)
Warning message:
package 'Biodem' was built under R version 3.1.3
> data(surnames);
> hed.kin <- hedrick(surnames);
> hed.kin;
           Due      Tre      Uno
Due 1.0000000 0.3606394 0.5545896
Tre 0.3606394 1.0000000 0.3812424
Uno 0.5545896 0.3812424 1.0000000
> |
```

Fig. 16 Fragmentos de código R de la función hedrick

La función **lasker** calcula el coeficiente de parentesco lasker, a partir de **x** (tabla de frecuencia de apellidos); retornando una matriz de parentesco simétrica cuadrada. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **lasker** del paquete **Biodem**.

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

```
> data(surnames);
> lask.kin <-lasker(surnames);
> lask.kin;
      Due      Tre      Uno
Due 0.10997732 0.04166667 0.06390977
Tre 0.04166667 0.12109375 0.04605263
Uno 0.06390977 0.04605263 0.12049861
> |
```

Fig. 17 Fragmentos de código R de la función lasker

La función **uri** que se muestra calcula el coeficiente de isonimia aleatorio imparcial, a partir de **x** (tabla de frecuencia de apellidos); devolviendo una matriz cuadrada simétrica imparcial de isonimia. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **uri** del paquete **Biodem**.

```
> data(surnames);
> iso.matrix <- uri(surnames);
> iso.matrix
      [,1]      [,2]      [,3]
[1,] 0.18095238 0.08333333 0.12781955
[2,] 0.08333333 0.19166667 0.09210526
[3,] 0.12781955 0.09210526 0.19883041
> |
```

Fig. 18 Fragmentos de código R de la función uri

La función **rri** que se muestra calcula una estimación objetiva de isonimia regional aleatoria, a partir de **x** (tabla de frecuencia de apellidos); devolviendo un valor numérico. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **rri** del paquete **Biodem**.

```
> data(surnames);
> reg <- rri(surnames);
> reg;
[1] 0.1311688
> |
```

Fig. 19 Fragmentos de código R de la función rri

La función **rel.cond** que se muestra calcula una matriz de parentesco condicional de datos isonimia, a partir de **x** (matriz imparcial aleatoria cuadrada de datos de isonimia), **R** (estimación objetiva de isonimia regional aleatoria), **method** (método a utilizar para el cálculo de coeficientes); devolviendo una matriz de parentesco condicional simétrica cuadrada. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **rel.cond** del paquete **Biodem**.

```
> data(surnames);
> iso.matrix <- uri(surnames);
> reg <- rri(surnames);
> kin.cond <- rel.cond(iso.matrix, reg);
> kin.cond;
      [,1]      [,2]      [,3]
[1,] 0.0128678527 -0.01236433 -0.0008657091
[2,] -0.0123643281 0.01563724 -0.0100969948
[3,] -0.0008657091 -0.01009699 0.0174888940
> |
```

Fig. 20 Fragmentos de código R de la función rel.cond

La función **rel.phi** que se muestra calcula una matriz de parentesco de datos isonimia 'a priori', a partir de **x** (matriz imparcial aleatoria cuadrada de isonimia), **R** (estimación objetiva de isonimia regional aleatoria), **method** (método a utilizar para el cálculo de coeficientes); devolviendo una matriz de parentesco simétrica cuadrada 'a priori'. Se muestra un fragmento de código R que muestra un ejemplo de uso de la función **rel.phi** del paquete **Biodem**.

```
> data(surnames);
> iso.matrix <- uri(surnames);
> reg <- rri(surnames);
> kin <- rel.phi(iso.matrix, reg);
> kin;
      [,1]      [,2]      [,3]
[1,] 0.04523810 0.02083333 0.03195489
[2,] 0.02083333 0.04791667 0.02302632
[3,] 0.03195489 0.02302632 0.04970760
> |
```

Fig. 21 Fragmentos de código R de la función rel.phi

3.3.2 Fragmento de código Java

A continuación se muestra el método **fst** implementado en la clase **Rurnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos del **fst** de la matriz de parentesco condicional.

```
public String fst() throws RserveException, REXPMismatchException {
    c.eval("iso.matrix <- uri(d)");
    c.eval("reg <- rri(d)");
    c.eval("N<-colSums(d)");
    c.eval("kind.cond <- rel.cond(iso.matrix, reg)");
    c.eval("salida <- Fst(kind.cond, dN)");
    String[] s = c.eval("capture.output(salida)").asStrings();
    String salida = ResourceBundle.getString("fst") + "\n";
    for (String item : s) {
        if(!item.contains("[,1]"))
            salida += item + "\n";
    }
    salida = salida.replace("[,1]", "");
    return salida + "\n";
}
```

Fig. 22 Fragmentos de código java de la función fst

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Se muestra el método **hedrick** implementado en la clase **Rurnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos del coeficiente de parentesco estandarizada **Hedrick**.

```
public String hedrick() throws RserveException, REXPMismatchException {
    String[] s = c.eval("capture.output(hedrick(d))").asStrings();
    String salida = resourceBundle.getString("hedrick") + "\n";
    return salida + String.join("\n", s) + "\n\n";
}
```

Fig. 23 Fragmentos de código de la función Hedrick

Se muestra el método **lasker** implementado en la clase **Rurnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos del coeficiente de parentesco **Lasker**.

```
public String lasker() throws RserveException, REXPMismatchException {
    String[] s = c.eval("capture.output(lasker(d))").asStrings();
    String salida = resourceBundle.getString("lasker") + "\n";
    return salida + String.join("\n", s) + "\n\n";
}
```

Fig. 24 Fragmentos de código java de la función lasker

Se muestra el método **uri** implementado en la clase **Rurnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos del coeficiente isonimia aleatoria imparcial.

```
public String uri() throws RserveException, REXPMismatchException {
    String[] s = c.eval("capture.output(uri(d2))").asStrings();
    String salida = resourceBundle.getString("uri") + "\n";
    return salida + String.join("\n", s) + "\n\n";
}
```

Fig. 25 Fragmentos de código java de la función uri

Se muestra el método **rri** implementado en la clase **Rurnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos de estimación objetiva de regional isonimia aleatorio.

```
public String rri() throws RserveException, REXPMismatchException {
    return resourceBundle.getString("rri") + "\n " + c.eval("rri(d2)").asDouble() + "\n\n";
}
```

Fig. 26 Fragmentos de código java de la función rri

Se muestra el método **rel_cond** implementado en la clase **R surnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos de matriz de parentesco condicional de datos isonimia.

```
public String rel_cond(String m) throws RserveException, REXPMismatchException {
    c.eval("iso.matrix <- uri(d)");
    c.eval("reg <- rri(d)");
    String[] s = c.eval("capture.output(rel_cond(iso.matrix,reg,method=\"\" + m + "\"))").asStrings();
    String salida = resourceBundle.getString("relCond") + "\n";
    return salida + String.join("\n", s) + "\n\n";
}
```

Fig. 27 Fragmentos de código java de la función rel_cond

Se muestra el método **rel_phi** implementado en la clase **R surnames.java**. Este método es el encargado de conectar el resultado de la ejecución de código R con la interfaz de Java para los cálculos de matriz de parentesco de datos isonimia 'a priori'.

```
public String rel_phi(String m) throws RserveException, REXPMismatchException {
    c.eval("iso.matrix <- uri(d)");
    c.eval("reg <- rri(d)");
    String[] s = c.eval("capture.output(rel_phi(iso.matrix,reg,method=\"\" + m + "\"))").asStrings();
    String salida = resourceBundle.getString("relPhi") + "\n";
    return salida + String.join("\n", s) + "\n\n";
}
```

Fig. 28 Fragmentos de código java de la función rel_phi

3.4 Pruebas

“El proceso de pruebas se centra en los procesos lógicos internos del software, asegurando que todas las sentencias se han comprobado, y en los procesos externos funcionales; es decir, realizar las pruebas para la detección de errores y asegurar que la entrada definida produce resultados reales de acuerdo con los resultados requeridos” (PRESSMAN, 2005)

Cada prueba contendrá un mínimo de información que se muestra a continuación.

- Objetivo de la prueba
- Descripción de la prueba

- Técnica
- Criterio de completitud
- Consideraciones especiales

Entre los principios que tienen las pruebas se pueden evidenciar los que a continuación se relacionan:

- En la pruebas se debe poder realizar un seguimiento de los requisitos del cliente. El énfasis aquí debe estar centrado en la capacidad que tiene el sistema para cumplir con los requisitos que plantea el cliente.
- Las pruebas que se le van a realizar al *software* se deben ser planificadas con antelación. Esta planificación se puede realizar incluso al finalizar la etapa de requisitos, posibilitando que una vez concluida la fase de análisis del sistema se pueda comenzar una detallada definición de los casos de pruebas (CP).
- Las pruebas deben comenzar por las funcionalidades independientes e ir progresando hasta probar el *software* como un todo. Este principio se enfoca en que las primeras pruebas realizadas al sistema están centrada en el funcionamiento individual de los módulos, una vez probados los módulos independientes se procede a probar el funcionamiento de varios módulos en conjunto, hasta llegar al funcionamiento del sistema en general.

Las pruebas están estructuradas por tipos de pruebas y técnicas de pruebas.

Los pruebas de *software* entre sus disimiles clasificaciones se encuentran estructuradas en métodos y niveles de pruebas. En el desarrollo de la presente investigación se utiliza el método de **pruebas de caja negra** el cual se centra en los requisitos funcionales del *software*, permitiendo obtener un conjunto de condiciones de entrada que permitan la ejecución completa de todos los requisitos funcionales de un programa. Estas pruebas tienen como objetivo encontrar errores, entre los que se destacan: funciones incorrectas o ausente, errores en las interfaces de usuarios, errores en los datos, errores de rendimientos, errores de inicialización y terminación. El método de caja negra se aplica mediante las **técnicas de partición de equivalencia**, la cual se basa en la división de los datos de entradas en clases a partir de las cuales se puedan derivar los CP.

En la etapa de pruebas en el desarrollo del Componente de análisis Biodemográficos para el sistema SEEGEN-R se utilizaron los siguientes niveles de pruebas:

1. Pruebas de integración

- **Objetivo de la prueba:** la prueba de integración se encarga de identificar errores introducidos mediante la combinación de programas probados de forma unitaria.
- **Descripción:** verifica que las interfaces de usuarios de los componentes a integrar mantienen un correcto funcionamiento. Determina además que los datos de prueba del sistema se carguen de forma correcta. Determina el camino para avanzar de una interfaz de usuario a otra entre los componentes a integrar. Permite la toma de decisiones y acciones a ejecutar ante cualquier problema detectado.
- **Técnicas utilizadas:** se comparan los resultados obtenidos, con los resultados esperados. Se realiza la técnica Top-Down que consiste en ejecución desde el módulo principal hasta los módulos más bajo. En el caso del componente informático de la presente investigación se hace pruebas a la integración del *plugin* con el sistema SEEGEN-R.

2. Pruebas de sistema.

- **Objetivo:** asegurar el correcto funcionamiento del sistema, una adecuada carga de datos, ejecución del sistema y la devolución correcta de los resultados obtenidos.
- **Descripción:** las pruebas del sistema están enfocadas a los requisitos del usuario agrupados en los CU, así como a las reglas y funciones que pueda tener el *software*. Como objetivo principal de estas pruebas, está el correcto funcionamiento del sistema en cuanto a la carga, ejecución y muestra de resultados, además valida una correcta implementación de las reglas del negocio.
- **Técnica:** ejecuta cada CU, flujo básico o función utilizando datos válidos e inválidos, para verificar que:
 - Los resultados esperados ocurren cuando se utiliza un dato válido.

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

- Los mensajes de error o de advertencia aparecen en el momento adecuado, cuando se utiliza un dato inválido.
- Cada regla de negocios es aplicada adecuadamente.

3.4.1 Diseño de casos de prueba

A partir de la descripción de los casos de uso del sistema se diseñó un CP para cada CU. Para desglosar el CU se divide en secciones, y estas a su vez en escenarios, con el objetivo de obtener mejores resultados en las pruebas aplicadas. Se muestra la tabla con el CP: Realizar estudios de tablas de frecuencias de apellidos.

Tabla. 8 CP: Realizar estudios de tablas de frecuencias de apellidos

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC 1: Realizar estudios de tablas de frecuencia de apellidos	EC 1.1: El especialista inserta los datos correctamente.	En este escenario el especialista inserta los datos.
	EC 1.2: El especialista inserta los datos incorrectamente	Este escenario sigue la misma funcionalidad que el anterior pero verifica que todos los datos estén insertados correctamente.
	EC 1.3: El especialista no inserta todos los datos necesarios	Este escenario valida que todos los datos estén insertados.

Partiendo de esta descripción se detallan las variables que se encuentran en la interfaz asociada al CU al que pertenece el CP.

Tabla. 9 Descripción de variables

No	Nombre del campo	Clasificación	Valor nulo	Descripción
1	Título del estudio	texto	No	Se introduce el título del estudio a realizar.
2	Tabla de frecuencias	lista desplegable	No	Se escoge la tabla de frecuencias a utilizar para el desarrollo del estudio.
3	Vector efectivo de la	lista	No	Se escoge el vector efectivo de la

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

	población	desplegable		población a utilizar para el desarrollo del estudio.
4	Método	texto	No	Se escoge el método a utilizar para el cálculo de coeficientes en el desarrollo del estudio

A continuación se describe la matriz de datos del CU Realizar estudios sobre tablas de frecuencias de apellidos.

Tabla. 10 Matriz de datos del CU: Realizar estudios de frecuencias de apellidos

ID del escenario	Escenario	Tabla de frecuencia de apellidos	Vector efectivo	Respuesta del Sistema
EC 1.1	El especialista inserta los datos correctamente	V	V	El especialista selecciona los datos cargados en el módulo base con los que desea realizar el estudio y presiona el botón "Aceptar". El sistema valida que los datos seleccionados por el especialista sean correctos en cuantos a formato, si estos son correctos devuelve los resultados del estudio.
EC 1.2	El especialista inserta los datos incorrectamente.	I	V	Si los datos cargados por el especialista no cumplen con el formato el sistema muestran el mensaje de error indicándole al usuario el tipo de dato necesario para el estudio. El sistema da la posibilidad de volverse a introducir los datos.

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

EC 1.3	El especialista no inserta todos los datos necesarios	I	I	Si el especialista no carga los datos necesarios para el análisis, el sistema devuelve el mensaje de error: "Por favor, entre los campos que faltan". El sistema permite introducir los datos nuevamente.
--------	---	---	---	---

3.4.2 Resultados de las pruebas aplicadas

Una vez aplicadas las pruebas de sistema en 3 iteraciones, se identificaron un total de 10 No Conformidades (NC) siendo todas significativas. En una primera iteración se encontraron 6 NC, en la segunda iteración se detectaron 4 NC, tanto las NC de la primera iteración, como la de la segunda fueron resueltas satisfactoriamente, lo que hizo posible que en una tercera iteración no se detectaran NC.

A continuación se visualiza una gráfica con la relación de las NC y su clasificación por iteraciones.

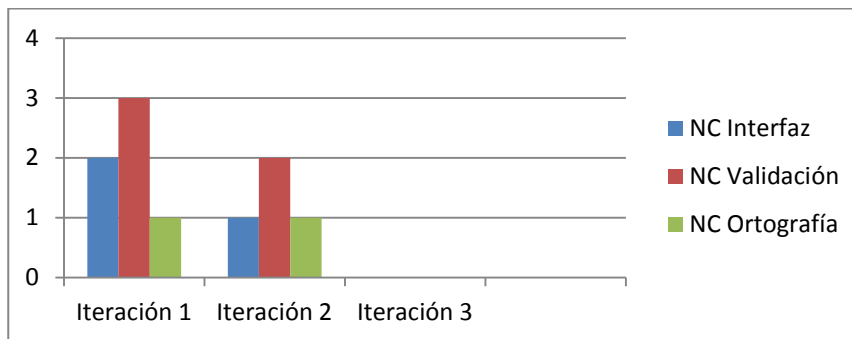


Fig. 29 Registro de No Conformidades

Para llevar a cabo la prueba de integración del componente al sistema SEEGEN-R se utilizó la clase *BiodemPlugin.java* encargada de visualizar el menú e integrar las clases visuales con las clases de abstracción del código R.

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

Como resultado de la integración se muestra la siguiente figura.



Fig. 30 Integración del componente al sistema SEEGEN-R

La pestaña "Paquetes" muestra los paquetes integrados al sistema, ejemplo el paquete "Biodem" que contiene los estudios relacionados con el componente integrado, a partir del cual el usuario podrá realizar cualquiera de los estudios una vez que los seleccione.

Para comprobar el correcto funcionamiento de la integración realizada, se muestran los resultados obtenidos en el sistema para las funcionalidades: calcular el coeficiente estandarizado Hedrick, calcular la matriz de isonimia, el valor de la isonimia regional aleatoria y el cálculo de la matriz de parentesco 'a priori' comparándolos con los resultados arrojados en el lenguaje de programación R, quedando demostrado que ambos resultados coinciden.

CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBAS DEL COMPONENTE DE ANÁLISIS BIODEMOGRÁFICOS PARA EL SISTEMA SEEGEN-R

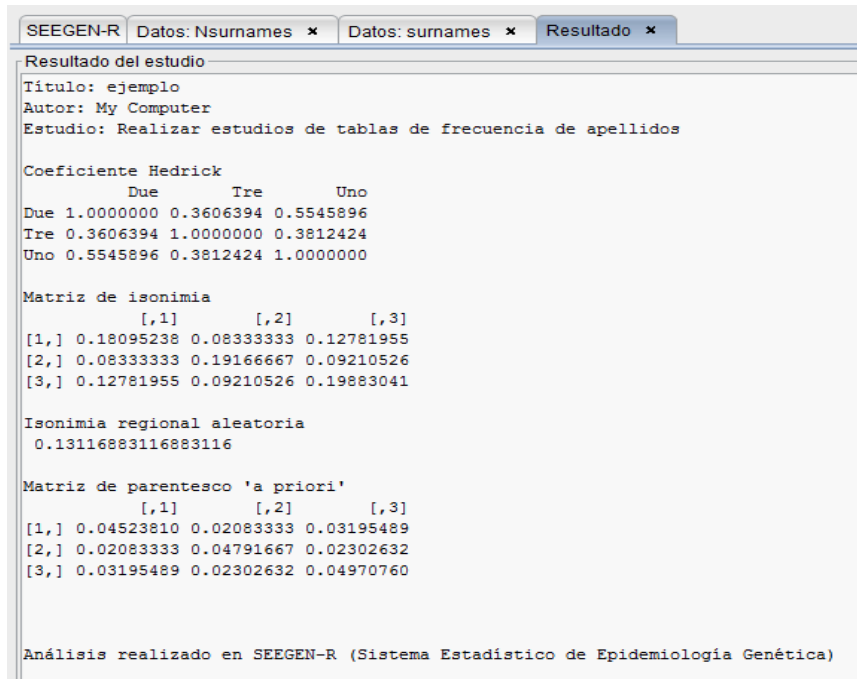


Fig. 31 Resultados obtenidos en el sistema SEEGEN-R.

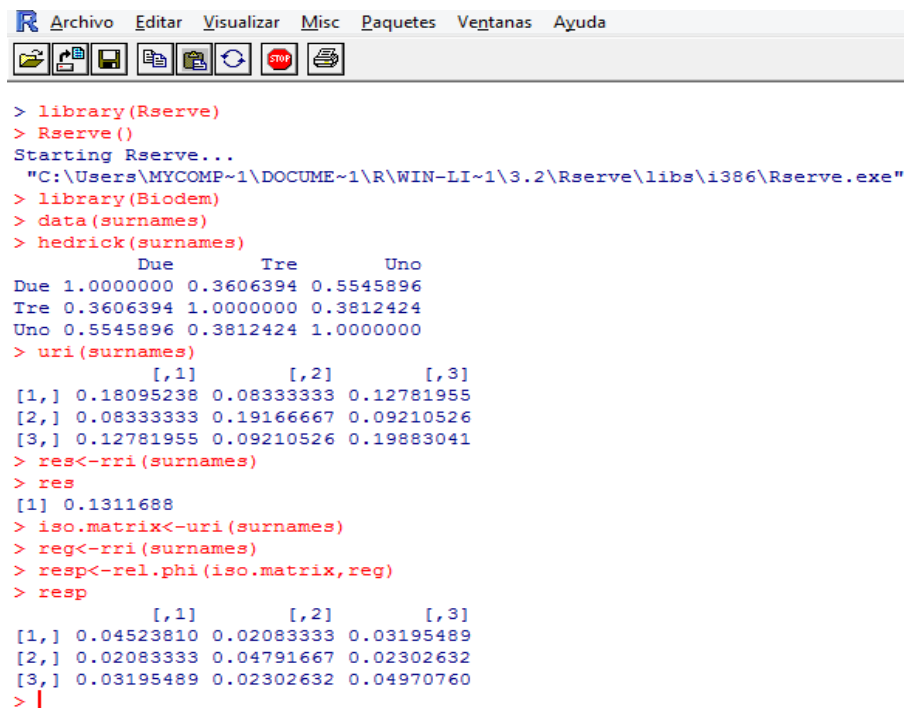


Fig. 32 Resultados obtenidos en R.

Conclusiones parciales

En este capítulo se realizó la implementación del componente de estudios Biodemográficos, evidenciándose la relación entre los componentes del sistema, mediante el diagrama de componentes. Se definieron los estándares de codificación y una breve descripción de los métodos implementados. Se realizaron las pruebas a nivel de sistema a través del método de caja negra mediante la técnica de partición de equivalencia, permitiendo encontrar 10 NC, los cuales fueron corregidos posteriormente, obteniendo un producto libre de errores. Además se realizó la integración del componente al sistema SEEGEN-R cumpliendo con los objetivos del mismo.

CONCLUSIONES GENERALES

En el desarrollo del presente trabajo de diploma se obtuvo como resultado el desarrollo de un componente que permite la realización de análisis estadísticos de estudios Biodemográficos en el sistema SEEGEN-R, para ello:

- Se realizó el estudio de *software* que realizan análisis estadísticos en estudios Biodemográficos identificándose que los mismos por si solos no son suficientes para realizar estos análisis a partir de las necesidades de los genetistas en el CNGM.
- La generación de los artefactos posibilitó un mejor entendimiento del negocio garantizando una correcta implementación de las funcionalidades del componente.
- Las pruebas diseñadas y aplicadas al componente a nivel de sistema garantizaron el correcto funcionamiento del componente desarrollado.
- Con el desarrollo del componente para la aplicación SEEGEN-R se obtuvo una herramienta que posibilita a los genetistas del CNGM los cálculos estadísticos de estudios Biodemográficos centralizados en un solo sistema.

RECOMENDACIONES

Se recomienda que el componente de análisis Biodemográficos:

- Sea implantado debido a su importancia y el aporte que le puede dar a los especialistas del CNGM en sus investigaciones.
- Extender la internacionalización del componente a otros idiomas para de esta forma facilitar su uso hacia todos los especialistas de otras lenguas.

REFERENCIAS BIBLIOGRÁFICAS

- Barchini, Graciela Elisa. 2005.** *Métodos "I + D" de la Informática.* Universidad Nacional de Santiago del Estero : s.n., 2005.
- Biodemografía. 2015.** [En línea] 2015.
<http://docsetools.com/revista-digital-educacion-tecnologia-educativa/contenido-17113.html>.
- Booch, Grady, Rumbaugh, Jim y Jacobson, Ivar. 2014.** *UML: El Lenguaje Unificado de Modelado.* 2014.
- Cendejas Valdéz, José Luis. 2014.** *Implantación del modelo integral colaborativo (MDSIC) como fuente de innovación para el desarrollo ágil de software en las empresas de la zona centro-occidente en México.* Estado de Puebla : s.n., 2014.
- Developer, Network. 2015.** [En línea] 2015. <http://msdn.microsoft.com/es-es/library/dd409360.aspx>.
- Ecured. 2015.** Genética de las poblaciones. [En línea] 2015.
http://www.ecured.cu/index.php/Gen%C3%A9tica_de_poblaciones.
- Epidat. 2014.** Epidat 4: Ayuda de Demografía. [En línea] Octubre de 2014.
http://www.sergas.es/gal/documentacionTecnica/docs/SaudePublica/Apli/Epidat4/Ayuda/Ayuda_Epidat_4_Demografia_Octubre2014.pdf.
- Escudero, Adrián, Iriondo, José M y Torres, M. Elena. 2003.** *Spatial analysis of genetic diversity as a tool for plant conservation.* España : s.n., 2003.
- Gamma, Erich. 2002.** *Design Patterns. Elements of Reusable Object-Oriented Software.* Addison-Wesley Professional. Londres : s.n., 2002. ISBN: 9780201633610.
- Hernández Orallo, Enrique. 2015.** El Lenguaje Unificado de Modelado (UML). [En línea] 2015.
<http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
- IBM. 2015.** IBM Developer works. [En línea] 2015. <http://www.ibm.com/developerworks/ssa/>.
- . 2015. IBM Knowledge Center . [En línea] 2015.
http://www-01.ibm.com/support/knowledgecenter/SSBLQQ_8.6.0/com.ibm.rational.rit.ref.doc/topics/c_ritref_the_logical_view.html?lang=es.
- . 2015. Knowledge Center. [En línea] 2015.
http://www-01.ibm.com/support/knowledgecenter/SSWSR9_11.0.0/com.ibm.pim.dev.doc/pim_tsk_arc_definingusecases.html?lang=es.
- InfoStat. 2015.** InfoStat Software estadístico. [En línea] 2015. <http://www.infostat.com.ar/>.

- La Rosa González, Yudiel, y otros. 2014.** *SEEGEN-R: Sistema Estadístico de Epidemiología Genética*. Ecuador : s.n., 2014.
- Mx. 2015.** Sitio oficial Mx. [En línea] 2015. <http://www.vcu.edu/mx/about-mx.html>.
- Netbeans. 2015.** Sitio Oficial . [En línea] 2015. <https://netbeans.org/features/>.
- PRESSMAN, R.S. 2002.** *Ingeniería de Software. Un enfoque práctico*. 2002.
- . **2005.** *Ingeniería de Software. Un enfoque práctico. Quinta Edición*. 2005.
- Real Academia, Española. 2015.** [En línea] 2015. <http://lema.rae.es/drae/srv/search?key=demograf%C3%ADa>.
- RUP. 2015.** Rational Unified Process. Best Practices for Software Development Teams. [En línea] 2015. https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf.
- Satman, M. Hakan. 2014.** *RCaller: A Software Library for Calling R from Java*. Istanbul University. Department of Econometrics. Beyazit, Istanbul. Turkey : s.n., 2014.
- SPSS. 2015.** IBM SPSS Statistics. [En línea] 2015. <http://www-01.ibm.com/software/analytics/spss/products/statistics/>.
- Storz, Jay F y Beaumont, Mark. 2002.** 2. *Testing for Genetic Evidence of Population Expansion and Contraction: An Empirical Analysis of Microsatellite DNA Variation Using a Hierarchical Bayesian Model*. Lincoln : s.n., 2002.
- . **2002.** *Testing for Genetic Evidence of Population Expansion and Contraction: An Empirical Analysis of Microsatellite DNA Variation Using a Hierarchical Bayesian Model*. Lincoln : s.n., 2002.
- Tecnología, Modelo de negocio innovador Fundación Premio Nacional de. 2011.** [En línea] 2011. http://www.fpnt.org.mx/boletin/Marzo_2011/Pdf/Modelo_de_Negocio_Innovador.pdf.
- Urbanek, Simon. 2015.** Binary R Server. Version 1.7-3. [En línea] 2015. <http://cran.r-project.org/web/packages/Rserve/Rserve.pdf>.
- Vázquez Sánchez, Vanessa. 2014.** *La biodemografía en Cuba: eficacia biológica de población residente en el Municipio Plaza de la Revolución*. Museo Antropológico Montané. Facultad de Biología. Cuba : s.n., 2014.
- Venables, W. N y Smith, D. M. 2015..** 17. Venables, W. N; Smith D. M. and the R Core Team. An Introduction to R. Notes on R: A Programming Environment for Data Analysis and Graphics. Version 3.2.0. [En línea] 2015.

VP. 2015. Visual Paradigm for UML Standard 8.0. [En línea] 2015.
<http://visual-paradigm-for-uml-standard.soft112.com/>.

Welicki, León. 2015. El Patrón Singleton. [En línea] 2015.
<https://msdn.microsoft.com/es-es/library/bb972272.aspx#m21>.

Wieggers, Karl. 2003. *Software Requirements, Second Edition*. 2003.

Zamora Araya, José Andrey y Vallejos Brenes, Rosibel Tatiana. 2012. Aprendiendo estadística con R. Sede Chorotega, Universidad Nacional, Liberia. [En línea] 2012.

BIBLIOGRAFÍA

1. Boattini, Alessio; Calboli, Federico; Canto, Vincente; Maechler, Martin. Package 'Biodem': Biodemography functions. Version 0.3. 2015.
2. Aguirre Fey, Daniel. Demografía y Genética de las Poblaciones de *Zamia loddigesii* Miq. Veracruz. México. 2004.
3. Booch, Grady; Rumbaugh, Jim; Jacobson, Ivar. UML: El Lenguaje Unificado de Modelado.
4. Gamma, Erich. Design Patterns. Elements of Reusable Object-Oriented Software. Londres: Addison-Wesley Professional, 2002. ISBN: 9780201633610.
5. IBM Developer works [En línea] [Citado el: 20 de Abril de 2015] .Disponible en: <http://www.ibm.com/developerworks/ssa>
6. Instituto de Estadística y Matemáticas de la WU Viena. What is R? The R Project for statistical computing. [En línea] [Citado el: 20 de Abril de 2015] .Disponible en: <http://www.r-project.org/about.html>.
7. PRESSMAN, R.S. Ingeniería de Software. Un enfoque práctico. 2002.
8. Satman, M. Hakan. RCaller: A Software Library for Calling R from Java. Istanbul University. Department of Econometrics. Beyazit, Istanbul, Turkey. 2014.
9. Storz, Jay F. Beaumont, Mark. Testing for Genetic Evidence of Population Expansion and Contraction: An Empirical Analysis of Microsatellite DNA Variation Using a Hierarchical Bayesian Model. University of Nebraska. Lincoln. 2002.
10. Urbanek, Simon. Binary R Server. Version 1.7-3. 2015 [En línea] [Citado el 8 de Mayo de 2015] Disponible en: <http://cran.r-project.org/web/packages/Rserve/Rserve.pdf>
11. Vázquez Sánchez, Vanessa. La biodemografía en Cuba: eficacia biológica de población residente en el Municipio Plaza de la Revolución. Museo Antropológico Montané. Facultad de Biología. Universidad de la Habana. Cuba.
12. Venables, W. N; Smith D. M. and the R Core Team. An Introduction to R. Notes on R: A Programming Environment for Data Analysis and Graphics. Version 3.2.0. 2015.
13. Visual Paradigm for UML Standard 8.0. [En línea] [Citado el: 19 de Abril de 2015]. Disponible en: <http://visual-paradigm-for-uml-standard.soft112.com/>

GLOSARIO DE TÉRMINOS

CNGM: Centro Nacional de Genética Médica.

SEEGEN-R: Sistema Estadístico de Epidemiología Genética- basado en R.

Modelo de Migración Malecot: se define como la probabilidad de que dos gametos, tomados cada uno de un individuo sean idénticos por descendencia. Si estos individuos aparean su coeficiente de parentesco será el coeficiente de consanguinidad de sus hijos.

Deriva Genética: es un proceso micro-evolutivo de tipo aleatorio o estocástico, admite un tratamiento matemático descrito en término de probabilidades.

Estocástica: La Real Academia Española define el término estocástica a perteneciente o relativo al azar. Teoría estadística de los procesos cuya evolución en el tiempo es aleatoria.

RUP: Rational Unified Process.

UML: Lenguaje Unificado de Modelado.

GUI: Interfaz Gráfica de Usuario.

IDE: Entorno de Desarrollo Integrado.

Isonimia: estudio de la frecuencia y distribución de apellidos en poblaciones humanas, mediante el cual pueden establecerse relaciones de parentesco y origen.