

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 6



CENTRO DE INGENIERÍA GENÉTICA Y BIOTECNOLOGÍA



SISTEMA DE ANÁLISIS DE SECUENCIAS MRNA PARA EL DISEÑO DE SIRNA

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE
INGENIERO EN CIENCIAS INFORMÁTICAS**

Autores:

Yanileyky Correa Fernández

Yoendris Carlos Diaz Garcia

Tutores:

Ing. Vladir Antonio Parrado Cruz.

Dr. Ricardo Bringas Perez.

Ciudad de la Habana, Cuba, Julio del 2007

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la facultad 6 de la Universidad de las Ciencias Informáticas y al grupo de especialistas del área de Bioinformática del Centro de Ingeniería Genética y Biotecnología, que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmamos la presente a los ____ días del mes de Julio del año 2007.

Nombre completo del primer autor

Nombre completo del segundo autor

Nombre completo del primer tutor

Nombre completo del segundo tutor

OPINIÓN DE LOS TUTORES SOBRE EL TRABAJO DE DIPLOMA

Título: Sistema de Análisis de Secuencias mRNA para el Diseño de siRNA.

Autores:

Yanileyky Correa Fernández

Yoendris Carlos Diaz Garcia

Los tutores del presente Trabajo de Diploma consideramos que durante su ejecución los estudiantes mostraron las cualidades que a continuación se detallan.

Los autores demostraron durante el desarrollo del trabajo de diploma un conjunto de cualidades valiosas en su futuro desempeño como profesionales en el campo de la Ingeniería Informática. Consideramos que alcanzaron las siguientes calificaciones: la independencia muy alta, originalidad muy alta, creatividad muy alta, laboriosidad muy alta y la responsabilidad muy alta.

El presente trabajo de diploma presenta magnífica calidad en el aspecto científico técnico, utilizando correctamente la metodología de la investigación y el proceso de desarrollo de software. Además constituye una documentación muy valiosa en lo que a diseño de siRNA respecta, así como de los algoritmos que se emplean en para el mismo. Los resultados obtenidos tendrán un elevado valor para el CIGB primeramente y luego para todo el polo científico.

Por todo lo anteriormente expresado consideramos que los estudiantes están aptos para ejercer como Ingenieros Informáticos; y proponemos que se le otorgue al Trabajo de Diploma la calificación de 5 puntos. Además consideramos que los resultados poseen valor para ser publicados.

Ing. Vladir Antonio Parrado Cruz.

Nombre completo del primer tutor

Dr. Ricardo Bringas Pérez

Nombre completo del segundo tutor

AGRADECIMIENTOS

Especialmente agradezco:

A la Revolución que sin ella nada de esto fuera posible.

A mi papá (papi chino) que esté donde esté, siempre estará a mi lado y en mi corazón, además es el ser que ilumina mi camino y quien me da fuerzas para luchar en la vida por mis metas.

A mi mamá Sofía por su apoyo constante en mi vida y principalmente en mis estudios, por todo el amor que he recibido de ella y porque es el ser más maravilloso que existe y lo mas grande que tengo.

A Carlitín por su inmenso apoyo en mi carrera y en mi vida.

A mi tutor Vladir por su ayuda constante e incondicional durante toda la tesis.

A mis hermanas Betty y Tania por confiar siempre en su hermanita más chiquita.

A mi fiel amigo Yendris por su apoyo, consejos y sincera amistad.

A mi amiga Arianna por su compañía inseparable en todo momento.

A toda mi familia, amigos y amigas por esperar de mí lo mejor.

Yanileyky Correa Fernández

Agradezco:

A la Revolución por haberme dado la posibilidad de estudiar esta carrera.

A mi mamá Elda C. (negra) por haberme ayudado siempre y apoyado en todo, durante mi vida estudiantil, así como darme los mejores consejos.

A mi papa Carlos M. por haberse preocupado todo el tiempo por mí.

A mis hermanas Yennis y Yaritza por haberme ayudado en mi carrera y apoyado siempre.

A Caridad Rodríguez (cachita) por el cariño y los consejos de coger el camino correcto, aunque ya no este físicamente, siempre la tendré en mi corazón.

A mis tíos y tías por ayudarme en mi carrera, especialmente (José, Elba y Dayami)

A mi familia completa por confiar en mí y siempre esperar lo mejor de mí.

A mi tutor Vladimir por la preocupación en cada paso de la tesis.

A mis compañeros de aula, a mis amigos de la escuela y fuera de la escuela.

A los Colaboradores por parte del CIGB (Bringas, Laya y Alexander).

Yoendris Carlos Díaz García

DEDICATORIA

A mi familia por todo el tiempo que les he robado especialmente a:

Mi mamá Sofía

Mis hermanas Betania y Tania

Mis sobrinos Francisco Rafael, José Carlos, Ernesto David y Leidý Laura

Yanileyky Correa Fernández

A mi familia por dedicarme todo el tiempo que necesité de ellos especialmente a:

Mis padres Elda C. y Carlos M.

Mis hermanas Yaritza y Yennis

Yoendris Carlos Diaz Garcia

Resumen

El silenciamiento de genes por interferencia de RNA (iRNA) es una tecnología reciente y de alto impacto en las investigaciones biológicas. Un componente fundamental de esta tecnología lo constituyen las herramientas informáticas para el diseño de los RNA pequeños de interferencia (siRNA). El Centro de Ingeniería Genética y Biotecnología (CIGB) es una de las instituciones cubanas pioneras en el uso en la investigación de esta nueva tecnología. A pesar de la existencia en Internet de aplicaciones que permiten realizar este diseño, el CIGB ha decidido desarrollar una herramienta propia con estos fines, que le permita desarrollar estas predicciones en un servidor local y agregar facilidades a la misma a partir de las recomendaciones y exigencias derivadas de la experiencia experimental de los investigadores. La ejecución de este proyecto se desarrollará de forma conjunta por el Centro de Ingeniería Genética y Biotecnología (CIGB) y la Universidad de las Ciencias Informáticas (UCI). Como un primer paso de éste proyecto se presenta éste trabajo, donde se propuso desarrollar una aplicación con una arquitectura cliente-servidor y con interfaz web amigable. Este trabajo también incluye el diseño y llenado de una base de datos que contiene información de los genes identificados en el genoma humano así como su focalización cromosómica. Además para garantizar la portabilidad de la aplicación se desarrollará bajo ambientes multiplataformas, fundamentalmente con herramientas de software libre.

Índice.

Introducción	12
Capítulo 1 Fundamentación teórica.....	16
<i>Introducción.</i>	16
1.1 <i>La informática, relación con las ciencias biológicas.....</i>	16
1.2 <i>El diseño de siRNA.</i>	17
1.2.1 <i>Qué es mRNA.</i>	17
1.2.2 <i>Progreso de los mecanismos de diseño de siRNA.</i>	18
1.3 <i>Sistemas existentes vinculados al campo de acción.</i>	18
1.3.1 <i>Target Finder SiRNA design tools from Ambion.</i>	19
1.3.2 <i>SiRNA Design Software.</i>	19
1.3.3 <i>Dharmacon siDESIGN Center.</i>	20
1.3.4 <i>siDirect.</i>	20
1.3.5 <i>DEQOR.....</i>	21
1.3.6 <i>IDT's RNAi Design.....</i>	22
1.4 <i>Limitaciones generales de los sistemas existentes.</i>	22
1.5 <i>Importancia del Desarrollo de la Aplicación.....</i>	23
1.6 <i>Tendencias, técnicas, metodologías, tecnologías actuales y software.....</i>	24
1.6.1 <i>Tecnología Java Server Page (JSP).....</i>	25
1.6.2 <i>Lenguaje JAVA</i>	25
1.6.3 <i>Gestor de bases de datos. PostgreSQL.</i>	26
1.6.4 <i>Web Services. Tomcat 5.0.12</i>	27

1.6.5 Herramienta de desarrollo.....	27
1.6.6 Metodología de Desarrollo de Software.....	28
1.6.7 UML (Unified Modeling Language)	31
1.6.8 Herramienta CASE.....	32
<i>1.7 Fundamentación de la tecnología a utilizar.....</i>	<i>33</i>
<i>1.8 Conclusiones</i>	<i>34</i>
Capítulo 2 Características del sistema.	35
<i>Introducción.</i>	<i>35</i>
<i>2.1 Objeto de estudio.</i>	<i>35</i>
2.1.1 Objetivos estratégicos de la organización.....	35
2.1.2 Flujo actual de los procesos.....	36
2.1.3 Análisis crítico de la ejecución de los procesos	36
<i>2.2 Procesos objeto de automatización.</i>	<i>37</i>
<i>2.3 Propuesta del sistema.</i>	<i>39</i>
2.3.1 Descripción de la solución propuesta del sistema.	39
2.3.2 Análisis comparativo de otras soluciones existentes con la propuesta.	40
<i>2.4 Modelo de Dominio.</i>	<i>41</i>
2.4.1 Reglas del negocio.....	41
2.4.2 Definición de las entidades y los conceptos principales.	41
2.4.3 Diagrama Modelo del Dominio.....	42
<i>2.5 Requerimientos Funcionales</i>	<i>42</i>
<i>2.6 Requerimientos No Funcionales.</i>	<i>44</i>
<i>2.7 Definición de los actores y casos de uso del sistema a automatizar.</i>	<i>45</i>

2.7.1 Actores	46
2.7.2 Casos de Uso Definidos.....	46
2.7.3 Diagrama de casos de uso del sistema a automatizar.	47
2.7.4 Descripción de casos de usos expandidos.	47
2.8 Conclusiones	55
Capítulo 3 Análisis y diseño del sistema.....	56
Introducción.	56
3.1 Modelo de clases de análisis del sistema.	56
3.2 Propósitos del diseño.....	56
3.3 Principios de diseño.	57
3.4 Diagramas de Clases Web.	58
3.5 Descripción de las clases.....	64
3.6 Diagramas de Interacción.	74
3.8 Descripción de las Tablas de la Base de Datos.....	78
3.9 Diagrama de Despliegue.	80
3.10 Tratamiento de errores.....	81
3.11 Definiciones de diseño que se aplica.....	81
3.12 Seguridad.....	81
3.13 Interfaz.	82
3.14 Concepción de la ayuda.	82
3.15 Conclusiones.	82
Capítulo 4 Implementación y prueba.	83
Introducción.	83

4.1 Diagrama de Componentes.....	83
4.1.1 Diagrama de Componentes General.....	84
4.1.2 Diagrama de Componentes para el módulo “LOGIN”.....	85
4.1.3 Diagrama de Componentes para el módulo “NOSOTROS”.....	85
4.1.4 Diagrama de Componentes para el módulo “DISEÑO siRNA”.....	86
4.2 Modelo de Prueba.....	87
4.3 Conclusiones.....	106
Conclusiones.....	107
Recomendaciones.....	108
Referencias bibliográficas.....	109
Bibliografía.....	111
Anexos.....	113
<i>Anexo 1 “Diagramas de secuencia”.....</i>	<i>113</i>
Glosario de Términos.....	120

Introducción

El silenciamiento de genes mediante los ARN pequeños de interferencia (siRNA) es un tema novedoso y de mucha actualidad, hace aproximadamente 8 años en estudios de otra índole, fue donde se descubrió la existencia de posibles genes que permitirían el silenciamiento de casi cualquier tipo de gen, lo que sería de gran utilidad para el desarrollo de la Biología Molecular, incorporando beneficios a la humanidad. En estudios más profundos se dedujo, que la creación de forma artificial de un gen como el siRNA, sería una de las principales herramientas experimentales hoy en día para combatir y buscar soluciones a problemas de enfermedades en animales, plantas y el genoma humano, siendo éste último el centro de grandes investigaciones.

El objetivo de los siRNA es desactivar secuencias genéticas específicas, previamente seleccionadas, donde las moléculas son capaces de modificar y silenciar la expresión genética de un gen determinado, lo que ha causado una revolución en la comunidad científica.

Existen hoy en día, gran número de científicos y especialistas que realizan estudios sobre el diseño de siRNA y su aplicación en el silenciamiento de genes. No obstante, en la actualidad todas las investigaciones y estudios llevados a cabo por los especialistas son realizados sobre la base de herramientas online publicadas en Internet, las mismas están encaminadas a resolver parte de los problemas que se presentan en cuanto al tema en cuestión.

Gestionar el diseño de siRNA y su aplicación en el silenciamiento génico es una tarea un poco difícil de desarrollar para los especialistas. Lo cual viene dado por la siguiente **situación problémica**: no se cuenta con una herramienta capaz de dar solución al tema tratado sobre el diseño de siRNA, razón por la que realizan todas las investigaciones y estudios a través de Internet. Esto provoca que haya ineficiencia en los resultados alcanzados, puesto que no toda la información que necesitan está siempre disponible en los servidores de Internet, además trae consigo otros problemas como la demora y pérdida de tiempo por problemas de velocidad de conexión para alcanzar las solicitudes realizadas; tampoco existe una unidad en las ideas expuestas en estos sitios, razón que provoca que haya diversidad sobre el mismo tema, y los usuarios no pueden estar al tanto de todas las informaciones que se presenten. Se torna difícil lograr que existan condiciones imprescindibles como la confidencialidad, la disponibilidad, pues la información debe estar en el momento que el usuario requiera de ella y como existe una dependencia de Internet puede

haber una publicación hoy pero al día siguiente no estar; otra e importante condición necesaria es la privacidad de la información que requiere que la misma sea vista y manipulada únicamente por quienes tienen el derecho o la autoridad de hacerlo, cosa que no ocurre cuando se navega por estos sitios; no se preocupan por la integridad , donde la información debe ser consistente, fiable y no propensa a alteraciones no deseadas. Otra cuestión relevante es que no existe un software libre que reúna todas las condiciones y cumpla con lo requisitos de diseño que desean los especialistas y además que le brinde al usuario la posibilidad de solicitar cualquier servicio que necesite sobre la base de las investigaciones con una alta eficiencia, obteniendo una respuesta concreta y precisa de sus necesidades, así como; que sea posible que el cliente pueda evaluar el trabajo finalizado ya, para que de esta forma se pueda conocer el nivel de satisfacción de los especialistas en los estudios e investigaciones sobre el tema abordado, por lo que concretamente se ha propuesto resolver el **problema** de ¿Cómo automatizar el proceso de diseño de siRNA?

Por todo esto, se hace necesario desarrollar un sistema que permita darle solución a los problemas aquí mencionados, y de esta forma eliminar el engorroso y difícil trabajo que hasta ahora se realiza.

Para ello se creará el **Sistema de Análisis de Secuencias mRNA para el diseño de siRNA** el cual será capaz de satisfacer todas las necesidades anteriormente planteadas, aportando avances en los proyectos investigativos que se realicen, permitiéndoles mayor confiabilidad y seguridad de la información, además reducir el tiempo de investigación y maximizar los resultados haciéndolos cada vez más eficientes.

Para la implementación del mismo se ha decidido desarrollar una herramienta Cliente – Servidor con interfaz web, debido a su fácil acceso, y además siempre estará disponible para los especialistas que la necesiten para sus estudios.

La obtención artificial de un gen constituye la forma más novedosa de aplicación de técnicas y algoritmos para automatizar la función de diseñar genes con características físicas- químicas deseadas. Es en esta dirección en la que se centra esta investigación por lo que el **objeto de estudio** de este trabajo está enmarcado en lo referente al desarrollo de aplicaciones informáticas Cliente-Servidor con interfaz Web teniendo como **campo de acción** las aplicaciones informáticas Cliente-Servidor con interfaz Web para el diseño de siRNA. Para dar solución al **problema** planteado se trazó como **objetivo general** desarrollar una herramienta Cliente–Servidor con interfaz web que permita el análisis de secuencia mRNA para el diseño de siRNA.

Para cumplir el objetivo general del trabajo se trazaron los siguientes **objetivos específicos**:

- Análisis del sistema que se propone.
- Diseño del sistema que se propone.
- Implementación del sistema propuesto.
- Validar la solución de software implementado.

Para cumplir con dichos objetivos, se trazaron las siguientes **tareas**:

- Estudio de las diferentes herramientas que se emplean para el diseño de siRNA.
- Intercambio con especialistas para determinar los requerimientos del sistema y mostrar prototipos para garantizar que el producto se adapte a las condiciones y requerimientos de los clientes.
- Estudio de las tecnologías necesarias y factibles para la construcción del sistema que se propone.
- Confección del modelo de dominio.
- Describir detalladamente los casos de uso del sistema.
- Confección de diagrama de caso de uso del sistema.
- Confección de diagramas de clases e interacción del análisis.
- Estudio de los principios de diseño y confección de los diagramas de clases web para cada caso de uso y los diagramas de interacción por cada escenario de los mismos.
- Realización del diseño de la base de datos.
- Confección de diagramas de componentes.
- Realización de pruebas al sistema.

Se pretende finalmente obtener un producto de software a la altura de las exigencias actuales de producción de software en la sociedad, con los principios de software libre, acorde con los estándares internacionales de catalogación y los estándares de diseño y presentación de aplicaciones informáticas Cliente-Servidor con interfaz web.

El documento ha sido estructurado de la siguiente manera:

Capítulo 1: Fundamentación Teórica, éste capítulo constituye la base teórica del presente trabajo. En él se describen los principales conceptos y lineamientos, como resultado de la investigación realizada para la concepción del Sistema de Análisis de Secuencias mRNA para el diseño de siRNA, teniendo en cuenta los acontecimientos nacionales e internacionales.

Además se muestran una serie de aplicaciones existentes en la actualidad vinculadas al diseño de siRNA y análisis de secuencias mRNA, para realizar una comparación de las mismas y realizar un estudio crítico-valorativo, también se expondrán las ventajas que la herramienta puede proporcionar.

Se describen las tendencias, técnicas, tecnologías actuales, metodologías y lenguajes a utilizar para la implementación del sistema, también se justifica el por qué del uso de las mismas para el proyecto a realizar.

Capítulo 2: Características del sistema, se describe el objeto de estudio, problemas existentes, se especifican los detalles de la construcción de la herramienta, propuesta del sistema, requerimientos funcionales y no funcionales y los casos de uso del sistema.

Capítulo 3: Análisis y diseño del sistema, se representa parte del análisis y del diseño del sistema que se propone, mostrándose la expansión de los casos de uso del sistema, los diagramas de interacción del sistema, y los diagramas de clases Web, así como la descripción detallada de las clases. Además se detalla el diagrama de modelo de datos, se plantean los detalles relacionados con el propósito del sistema, se describe el tratamiento de errores, los principios en que se basa la ayuda general y el sistema de seguridad. También se especifican principios tenidos en cuenta para la implementación y diseño de interfaz de la herramienta propuesta.

Capítulo 4: Implementación y prueba, se detalla el diagrama de componentes, y se exponen las pruebas que se le realizaron al sistema principalmente sobre la interfaz de la aplicación, es decir, se llevó a cabo la realización de las pruebas de caja negra para comprobar la presencia de errores en el sistema en caso de tenerlos.

Introducción.

En este capítulo se analiza brevemente la relación de la informática con las ciencias biológicas; así como la importancia del desarrollo de ésta aplicación. Se presentan además una serie de aplicaciones vinculadas con el tema de diseño de siRNA. Se fundamentan las tendencias y tecnologías actuales utilizadas para el logro de una herramienta o producto con calidad.

1.1 La informática, relación con las ciencias biológicas.

La informática en la actualidad es una innegable solución a muchos de los problemas que hace algunos años parecían imposibles sus respuestas. Resaltando su papel en diferentes áreas o ciencias, pero una de las más notable ha sido la Biología Molecular. Hoy en día las ciencias en su totalidad se hacen reales por los tantos logros que contribuyen al bienestar de la sociedad, evidenciando esto el avance del desarrollo tecnológico en importantes esferas de la vida.

Cualquier progreso en genómica, proteómica y otros campos biomédicos requiere un sólido apoyo de la informática, ya que es necesario procesar y manejar de un modo eficiente la abundante información que estas técnicas masivas generan. La informática pretende brindar una herramienta al biólogo a través de la cual él pueda afrontar la integración coherente de grandes cantidades de datos, al permitir la organización del conocimiento en si mismo, con lo cual se afirma que durante los últimos años se han realizado grandes avances en el desarrollo de las herramientas web para el análisis masivo de estos datos.

La producción de software es uno de los aspectos importante a tener en cuenta cuando se habla de avances y logros en la informática, lo cual se ha generalizado y abarca todas las esferas de la vida.

Es importante mencionar que se han creado herramientas principalmente publicadas en sitios online para el diseño de siRNA. Las cuales han sido fuentes utilizables en el presente trabajo como guía base para adquirir experiencia sobre las técnicas de diseño que estos emplean y apreciar cómo se obtienen resultados según las especificaciones de las propiedades, reglas o algoritmos que emplean los

desarrolladores en el análisis de secuencias. Las mismas constituyen una valiosa herramienta por su facilidad de uso, pero con la desventaja notable de la dependencia de la red global (Internet).

1.2 El diseño de siRNA.

Para alcanzar un avance notable en el desarrollo de la Biología Molecular se necesita entre otros aspectos obtener resultados de una forma totalmente exitosa desde el punto de vista científico de la composición genética, forma o estructura y funcionalidad de los genes, con la presencia y apoyo de las novedosas tecnologías. Después del descubrimiento del siRNA, varios estudios procuraron identificar las características óptimas requeridas para el diseño de las mismas. Estas técnicas permiten el rápido diseño y la posibilidad de realizar análisis de la función de los genes a través de un silenciamiento génico estable y heredable. Por lo tanto, podría llegar a utilizarse con fines terapéuticos, lo que constituye una necesidad para las expectativas de progreso en la Biotecnología y así poder cambiar infinidad de obstáculos a los que hoy en día se enfrenta. Los beneficios que se obtengan tienen que ver de una forma muy íntegra con el diseño del RNA pequeño de interferencia el mismo puede no ser en su totalidad perfecto, pero asegura resultados más potenciales y satisfactorios que las otras técnicas aplicadas para silenciar la expresión de un gen específico.

La definición de diseño se basa en la aplicación de tecnologías, técnicas y algoritmos de alto rendimiento de procesamiento para facilitar estudios en la genómica a grandes escala. En este caso el diseño de los RNA pequeños de interferencia, pone de manifiesto la necesidad intrínseca de ser empleada para lograr el silenciamiento de casi cualquier gen y la vital importancia de los mismos.

Diseñar es la tarea principal de la herramienta propuesta donde se transforman datos abstractos y fenómenos complejos de la realidad en una valiosa técnica, además no es sólo un resultado de técnicas aplicadas, incluye también un análisis detallado de los estudios e investigaciones de los especialistas que estudian las reglas para que el diseño sea llevado a cabo y se pueda obtener la información precisa para una investigación posterior de cualquier secuencia de ARN mensajero (mRNA).

1.2.1 Qué es mRNA.

El ARN mensajero es el ácido ribonucleico que contiene la información genética procedente del ADN para utilizarse en la síntesis de proteínas, es decir, determina el orden en que se unirán los aminoácidos.

Las secuencias de mRNA son el primer objetivo para realizar el diseño de los siRNA, puesto que es necesario llevar a cabo un análisis previo de estas secuencias obteniendo como resultado la información de una serie de sitios blancos (en inglés targets sites) los cuales se llevarán a prueba para determinar una eficiente aplicación de los diseños obtenidos en el proceso de silenciamiento de genes.

1.2.2 Progreso de los mecanismos de diseño de siRNA.

El progreso de los mecanismos de diseño de siRNA y, en lógica consecuencia, del silenciamiento de genes, se han relacionado estrechamente con el avance y desarrollo de disciplinas tales como la bioquímica y muy particularmente con la Biología Molecular, la Biotecnología y la Bioinformática. En el desarrollo de una nueva técnica para lograr el buen diseño y regular la expresión genética hay que considerar que la investigación ha de ir unida a una simultánea toma de decisiones. La interacción de todas estas disciplinas y tecnologías ha incrementado la eficacia en el silenciamiento de genes específicos mediante el diseño obtenido según reglas y algoritmos aplicados al análisis de secuencias mRNA, de potencial ayuda para obtener los resultados esperados en esta rama de la Biotecnológica, cuya consecuencia es el silenciamiento directo de genes y el rápido análisis de la función de los mismos, así como la generación más eficiente de nuevas estructuras en los genes. Para obtener un nuevo diseño aplicable de forma óptima, se necesita conocer primeramente la secuencia de nucleótidos y especificar las reglas de cumplimiento para el análisis, para obtener resultados satisfactorios y así predecir de una manera adecuada y efectiva las actividades biológicas en las que estarían implicados los ARN pequeños de interferencia.

Actualmente existen varias formas de silenciar genes, podrían citarse: mediante mutaciones de genes; que permiten obtener de manera general las nuevas características, funciones y estructuras químico-físicas, biológicas o ambas del gen analizado. Y también están los siRNA que realiza funciones muy parecidas a la del mecanismo de mutación pero se realizan de forma artificial y prometen resultados con mayor eficiencia y rapidez.

1.3 Sistemas existentes vinculados al campo de acción.

Actualmente las herramientas existentes sobre el diseño de siRNA constituyen la puesta en práctica de un conjunto de teorías e investigaciones sobre el tema, es decir, son el resultado de ingenieros y especialistas médicos, los cuales ocupan un importante papel en el progreso tecnológico e investigativo.

Existen ciertas herramientas online dedicadas al diseño de siRNA, de las cuales se presenta una muestra de las más conocidas a nivel internacional, porque a nivel nacional y particularmente en el Centro de Ingeniería Genética y Biotecnología no existen software desarrollados con estos fines, por lo que se considera que la herramienta que se desarrollará sobre el análisis de secuencias mRNA para el diseño de siRNA va a constituir el primer trabajo en Cuba sobre el tema tratado.

1.3.1 Target Finder SiRNA design tools from Ambion.

Es una herramienta creada por la compañía Ambion para encontrar los sitios blancos en una secuencia de mRNA de interés para ser utilizada en diseñar un siRNA. Se hizo uso de recomendaciones para el diseño de siRNA de otras fuentes y software. Los científicos de Ambion encuentran generalmente que el 50% de siRNAs diseñados usando esta herramienta reducirán la expresión de gen blanco por el 50%. [11]

Funciones Básicas de Target Finder siRNA.

- Pegar secuencia del mRNA en la ventana.
- Seleccionar el contenido máximo de G/C (%).
- Especificar la condición de parada para el siRNA.
- Seleccionar la condición que se debe evitar para que el siRNA sea químicamente sintetizado.

1.3.2 SiRNA Design Software.

SDS (software para el diseño de siRNA) es una herramienta que ayuda a diseñar los siRNAs para silenciar la expresión del gen. La herramienta toma una secuencia del mRNA y hacen uso de las herramientas de diseño existentes para hacer salir un sistema de candidatos. Filtran a estos candidatos basados en la estructura secundaria. [12]

El SDS es un software con las características siguientes:

- Proporciona una plataforma unificada para usar software existente.
- Compara la salida del software existente.

Funciones Básicas de siRNA Design Software.

- Entrar el nombre de la secuencia del mRNA (opcional).

- Entrar secuencia del mRNA que va a ser analizada.
- Especificar una posición de comienzo y fin para analizar el segmento que comprenda.
- Entrar el número de secuencia.
- Seleccionar el Organismo por ejemplo (Humano, ratón).
- Especificar los requisitos de los siRNAs de la salida como por ejemplo: longitud del sitio blanco del siRNA (ejemplo 21 nucleótidos).

1.3.3 Dharmacon siDESIGN Center.

Es una herramienta de diseño con notables ventajas y con una interfaz de usuario amigable, con significativas mejoras en el proceso de identificar siRNA comparándolo con otras herramientas e incorpora otros criterios de diseño bien evaluados por los científicos de Dharmacon. También ofrece flexibilidad al definir los sitios blancos específicos, ajustando los criterios de diseño, y utilizando el algoritmo BLAST. Las listas de secuencias de siRNA candidatas son proporcionadas todo el tiempo a partir de las secuencias o genes obtenidos en el diseño de siRNA. Para lograr una herramienta más potente se hizo uso de tecnologías como Dharmacon's *SMART*selection y *SMART* pool®. [13]

Funciones Básicas de Dharmacon siDESIGN Center.

- Especificar la especie por ejemplo, Human-Homo sapiens.
- Identificar la secuencia de nucleótidos de mRNA. Entrar número o NCBI.
- Entrar secuencia de nucleótidos (máximo 10,000).
- Seleccionar criterios, por ejemplo contenido G/C (%).
- Analizar el gen o la secuencia.
- Mostrar la secuencia o gen en la lista de siRNA candidatos.

1.3.4 siDirect.

Es un software para diseñar los RNA pequeños de interferencia (siRNA) se tienen en cuenta reglas mas adecuadas y eficaces para lograr una máxima especificidad en la interferencia mamífera del RNA (RNAi).

El sistema acepta una secuencia como entrada y vuelve una lista de los candidatos de siRNA. El algoritmo del siDirect incorpora pautas para favorecer RNAi mamífero y obtener éxito en los resultados. Investiga todos los candidatos de siRNA para evitar que haya una hibridación en los sitios blancos del gen que impone efectos en el silenciamiento. [14]

Funciones Básicas de Dharmacon siDirect.

- Entrar número de secuencia o pegar la secuencia de nucleótidos.
- Seleccionar una opción de diseñar pautas.
- Buscar sitios blancos.
- Especificar Organismo.
- Especificar en la secuencia la posición para diseñar el siRNA.
- Especificar contenido de G/C.
- Tener en cuenta reglas a evitar para el diseño de siRNA.

1.3.5 DEQOR

Es una herramienta que ayuda en el diseño y control de calidad de RNA pequeño de interferencia (siRNA) para interferencia del RNA (RNAi) y silenciar el gen. Evalúa la potencia inhibidora de las secuencias potenciales de siRNA así como identificar las regiones del gen que tiene un alto poder de silenciamiento. DEQOR puede predecir la probabilidad que un fragmento del mRNA cruzara-reaccionará con otros genes en la célula y ayudará que investigadores diseñen experimentos para probar la especificidad de siRNAs o químicamente diseñará siRNAs.

Funciones Básicas de DEQOR.

- Introducir la secuencia en FASTA o el formato crudo (5kb es el límite).
- Incorporar alternativamente el número de secuencia o NCBI.
- Seleccionar el genoma.
- Seleccionar criterios para ser mostrados, como por ejemplo: mostrar el porcentaje de la CROMATOGRAFÍA GASEOSA y la cantidad de Guanosines en una fila.

1.3.6 IDT's RNAi Design.

Es una herramienta que permite diseñar siRNA de una manera más completa, empleando una serie de criterios, que la hacen más poderosa en su función. La misma contiene 3 páginas [15]. El diseño comienza a partir de la página de la START, donde se realiza:

- Entrar en un nombre de la secuencia la cual debe tener como máximo 30 caracteres,
- Entrar NCBI ID #.
- Conseguir la secuencia o pegar secuencia (Limite 10,000 Bases).
- Entrar número de Oligos por secuencia (limite 20).
- Escoger la regla para el criterio de diseño.

Luego se pasa a la sección de TARGET.

- Editar la longitud para el RNAi doble.
- Seleccionar el contenido de G/C.
- Seleccionar los motivos usados para seleccionar los RNAi candidatos.

La tercera y última sección es OVERHANG.

- Permite personalizar cualquier proyección de los RNAi candidato.

El desarrollo de una herramienta para el diseño de siRNA es un proceso largo y un poco engorroso, su objetivo principal es diseñar un algoritmo para la predicción de estos genes e investigar profundamente sobre su función, y la obtención de nuevas características de ellos, etc., en los diferentes organismos incluyendo el genoma humano (Homo Sapiens).

1.4 Limitaciones generales de los sistemas existentes.

Los sistemas existentes anteriormente mencionados como DEQOR, Target Finder SiRNA design tools y muchos otros, solo son aplicaciones online y versiones de las mismas para probar sus funcionalidades, las cuales pueden cambiar muy a menudo, por lo que se puede afirmar que los estudios o trabajos investigativos no se pueden realizar de manera confidencial ni sistemática.

Además el proceso de obtención de la cadena de mRNA a analizar es verdaderamente engorroso porque la información de esta cadena que se debe analizar hay que conformarla de una serie de datos que están almacenados en una Base de Datos que solamente para su conformación sería necesario realizar una nueva aplicación. La mayoría de estos software solo permiten darle como entrada una cadena de mRNA ya conformada y los que brindan la posibilidad de formar la cadena, no ofrecen seguridad en cuanto a los resultados obtenidos debido a que nada prueba la veracidad de estos y nunca se conocen los algoritmos que se han utilizado para hacer el análisis de secuencias mRNA para la predicción de los siRNA, cuyos algoritmos son muy importantes en este proceso para tener conocimiento de lo que se hace y cómo; además otra desventaja es que no dan la posibilidad de hacer un análisis con el por ciento de exactitud que se estime conveniente de acuerdo al estudio que se esté realizando.

Es importante mencionar que las aplicaciones potentes que puedan existir no están a disponibilidad de todos los usuarios, es decir, pertenecen a compañías privadas.

Por las diversas limitaciones expuestas anteriormente es que se hace necesario cada vez más desarrollar una herramienta capaz de resolver estos problemas, además traer consigo muchos logros esperados y una buena aplicación para los diseños de siRNA.

1.5 Importancia del Desarrollo de la Aplicación

Después de haber analizado las herramientas que tienen similitud con nuestros requerimientos y necesidades, se puede afirmar que tienen varias características que de cierta manera se asemejan a la situación tratada, pero que no satisfacen en su totalidad el proceso propuesto. **Dharmacon siDESIGN Center** gestiona lo referente a la predicción de siRNA, pero solamente analiza las secuencias siguiendo escasas reglas de diseño, también define sitios blancos específicos del gen; pero como es una aplicación online, imposibilita que se puedan comprobar o verificar los resultados, además no se cuenta con la seguridad de que los análisis se realicen con la certitud requerida, por lo que se puede decir que no cumple con las exigencias de los clientes, que es la principal funcionalidad del sistema que se requiere. **Target Finder siRNA** por su parte permite también el análisis de secuencias para el diseño de siRNA pero se restringe a que la misma debe ser entrada, es decir, no brinda la posibilidad de poder extraer la información de una base de datos a partir de una especificación dada, además solo se limita a encontrar sitios blancos en la secuencia, imposibilitando la realización de otras técnicas para un análisis profundo y

lograr un buen diseño de los RNA pequeño de interferencia (siRNA), mediante ésta vía de las herramientas Web. Estos sistemas no brindan la posibilidad de poder diseñar un siRNA que reúna las mejores reglas establecidas por los clientes, es decir, los requerimientos para la herramienta propuesta son mucho más amplios. Por lo que se concluye en que es necesaria e imprescindible la realización de un nuevo sistema que cumpla con las características requeridas para satisfacer lo concerniente al análisis de secuencias mRNA para el diseño de siRNA.

Llevado a cabo todo el estudio crítico y comparativo sobre las aplicaciones que son utilizadas para el diseño de ARN pequeños de interferencia (siRNA) se llegó a la conclusión que ninguno satisface de forma general las necesidades que se han planteado. Cada especialista diseña su propio siRNA de acuerdo a sus estudios y necesidades, por tal razón las reglas de diseño no coinciden en ninguna de las aplicaciones mencionadas. Además en Internet sólo se pueden encontrar referencias o la base para desarrollar un estudio preliminar de lo que realmente desea un especialista en la materia.

Con el desarrollo de la aplicación, se les facilitará el trabajo a los especialistas. Su interfaz amigable, la posibilidad de diseñar ARN pequeños de interferencia (siRNA), harán de ella una herramienta que tanto para los biólogos como para los químicos que trabajen con ella, lo hagan de manera interactiva, práctica y agradable. La misma permite principalmente hacer un poco más fácil y ameno el proceso de obtención de siRNA, las cuales servirán en un futuro bastante cercano para el desarrollo de los silenciamientos génicos y una profundización en la investigación sobre el tema, basado en los resultados que obtengan al aplicar el sistema.

El tener una herramienta propia, proporciona la ventaja de conocer las técnicas de diseño implementadas y los códigos, lo cual brindará además la posibilidad de incorporar nuevos métodos y algoritmos para mejorar el diseño de siRNA a medida que se incursiona en el tema tratado. Estos aspectos o características hacen que la aplicación sea única, además por la principal razón que el CIBG no cuenta con una herramienta capaz de llevar a cabo este trabajo. También el hecho que sea multiplataforma posibilita su uso bajo cualquier sistema operativo.

1.6 Tendencias, técnicas, metodologías, tecnologías actuales y software usados.

Para el desarrollo del sistema, se realizó un estudio de las posibles herramientas a utilizar en su construcción, teniéndose en cuenta principalmente las peticiones y sugerencias de los clientes sobre el tema, así como la tendencia actual y las novedades de cada una de ellas. Para el desarrollo de esta aplicación se hará uso de estándares que utiliza el centro con el que se desarrolla el proyecto (CIGB), puesto que la aplicación puede ser usada por otros sistemas, por tal razón hay que ajustarse a esos requerimientos.

➤ ***Primera consideración para seleccionar tecnologías.***

Estándares establecidos por la organización a tener en cuenta:

1.6.1 Tecnología Java Server Page (JSP)

JSP es una tecnología Java que permite a los programadores generar contenido dinámico para web, en forma de documentos HTML, XML, o de otro tipo. JSP's permite al código Java y a algunas acciones predefinidas ser incrustadas en el contenido estático del documento web. **[10]**

JSP es un marco de trabajo de programación generado en Java, es un lenguaje estructurado muy potente, que puede utilizarse en un servidor para generar eficaces aplicaciones Web, permitiendo además un rápido desarrollo de las mismas.

Debido a que la tecnología JSP es abierta y multiplataforma, los servidores web, plataformas y otros componentes pueden ser fácilmente actualizados o cambiados sin que afecte a las aplicaciones basadas en la tecnología JSP.

1.6.2 Lenguaje JAVA

Características del lenguaje Java.

Java es un lenguaje de programación muy dinámico, basado en clases y orientado a objetos. Los objetos se agrupan en estructuras encapsuladas tanto sus datos como los métodos (o funciones) que manipulan esos datos. Además Java es compilado en un bytecode que es ejecutado, por una máquina virtual Java. El lenguaje toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos mucho más simple y elimina herramientas de bajo nivel como punteros. **[7]**

Java ofrece toda la funcionalidad de un lenguaje potente y robusto ya que 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 errores. Por la naturaleza distribuida de Java la seguridad se impuso como una necesidad de vital importancia. Así que se implementaron barreras de seguridad en el lenguaje y en el sistema de ejecución en tiempo real. [8]

Una de las características más importante de Java es que posee una arquitectura neutral, es decir el compilador Java compila su código a un fichero objeto de formato independiente de la arquitectura de la máquina en que se ejecutará. La indiferencia a la arquitectura representa sólo una parte de su portabilidad. Además, Java especifica los tamaños de sus tipos de datos básicos y el comportamiento de sus operadores aritméticos, de manera que los programas son iguales en todas las plataformas.

1.6.3 Gestor de bases de datos. PostgreSQL.

¿Qué es PostGreSQL?

PostGreSQL es un sistema de gestión de bases de datos objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de esto, PostGreSQL no es un sistema de gestión de bases de datos puramente orientado a objetos. [5].

Lo mejor de PostGreSQL...

Las características positivas que posee este gestor según las opiniones más comunes en Internet, son:

Posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta (en algunos benchmarks se dice que ha llegado a soportar el triple de carga de lo que soporta MySQL).

Implementa el uso de rollback's, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que MySQL no podría.

Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle. [5]

1.6.4 Web Services. Tomcat 5.0.12

Tomcat. (También llamado Jakarta Tomcat o Apache Tomcat) Es un servidor web con soporte de servlets y JSPs, es considerado un servidor de aplicaciones potente. Incluye el compilador Jasper, que compila JSPs convirtiéndolas en servlets.

En sus inicios existió la percepción de que el uso de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones. Hoy en día ya no existe esa percepción y Tomcat es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad. Dado que Tomcat fue escrito en Java, funciona en cualquier sistema operativo que disponga de la máquina virtual Java. [3]

➤ ***Segunda consideración para seleccionar tecnologías.***

Se tuvo en cuenta otras tecnologías para la construcción del sistema, buscando que la herramienta que se desarrolle cumpla con las condiciones establecidas por la sociedad de software libre, por tal razón, se realizó un estudio profundo donde se analizaron y compararon las más adecuadas a los propósitos trazados.

1.6.5 Herramienta de desarrollo.

NetBeans vs. Eclipse

NetBeans

NetBeans es un Entorno Integrado de Desarrollo (IDE), además es un producto libre y gratuito sin restricciones de uso. Es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. Entre sus características se encuentra sistema de proyectos basado en Ant, control de versiones y refactorización.

Modularidad. Todas las funciones del IDE son provistas por módulos. Cada módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente. [16]

Eclipse

Eclipse es una potente herramienta universal de entorno de desarrollo de software desarrollado en Java, usa java como lenguaje de programación aunque permite plugins para varios lenguajes más. La plataforma esta construida en base a plugins, mecanismo que permite desarrollar, integrar y correr nuevos plugins. Otros beneficios que aporta el uso de Eclipse son:

- Es una herramienta de código abierto.
- Eclipse es neutral y adaptable a cualquier tipo de lenguaje, por ejemplo C/C++, Cobol, C#, XML, etc. La característica clave de Eclipse es la extensibilidad. [3]
- Soporta la programación orientada a objetos (POO).
- La depuración e implementación de aplicaciones resultan mucho más sencillas.
- Corre en una gran cantidad de sistemas operativos incluyendo Windows y Linux.
- Provee a los desarrolladores, herramientas (ej.- PDE) que facilitan la creación de plugins.

¿Por qué utilizar Eclipse?

Después de analizar las características de NetBeans y las de **Eclipse** se decidió que este último fuera el IDE seleccionado para desarrollar la herramienta propuesta por sus grandes potencialidades y sus posibilidades de extensión mediante el uso de plugins, además por presentar funcionalidades más convincentes y realmente necesarias para llevar a cabo un software eficiente y que corra en diferentes sistemas operativos, y la principal razón se encuentra en el conocimiento y experiencia del equipo de desarrollo de software, que ha usado este IDE en múltiples aplicaciones lo que permite mayor rapidez de trabajo porque como se sabe este constituye uno de los elementos que influye en el tiempo de terminación y desarrollo de cualquier sistema .

1.6.6 Metodología de Desarrollo de Software.

RUP (Rational Unified Process) vs. XP (eXtreme Programming)

RUP (Rational Unified Process)

El proceso unificado de desarrollo (RUP), es en la actualidad uno de los más usados por las empresas de software y es validado continuamente para el perfeccionamiento de su uso. Está concebido para que desde el inicio del proceso, se establezca una definición acertada del proyecto, haciendo innecesarias las reconstrucciones parciales posteriores. Además está dirigido a la programación orientada a objetos que permite obtener sistemas escalables en el tiempo que no necesitarán grandes inversiones de recursos en sus modificaciones posteriores.

Las características principales de este proceso son:

- ✓ Guiado por los Casos de Uso
- ✓ Centrado en la Arquitectura
- ✓ Iterativo e incremental

A través de un proyecto guiado por RUP, los requerimientos funcionales son expresados en la forma de Casos de Uso, que guían la realización de una arquitectura ejecutable de la aplicación. Además el proceso focaliza el esfuerzo del equipo en construir los elementos críticos estructuralmente y del comportamiento (llamados Elementos Arquitecturales) antes de construir elementos menos importantes.

Este proceso se basa en el modelo en espiral que organiza las iteraciones por etapas y fases para obtener una estructura más sólida, clara y ajustable a las necesidades particulares de cada organización, facilitando además la administración del proyecto. Cada iteración se considera un subproyecto que no solo genera documentación, sino también productos de software, permitiendo con esto que el usuario tenga puntos de verificación y control más rápidos, y que se realice un proceso continuo de pruebas y de integración desde las primeras iteraciones.

Además controla, a través de un esquema predefinido de escalabilidad y gestión de riesgos, el manejo de los mismos, con lo cual posibilita que se reconozcan previamente los posibles problemas, y fallos, para su temprana prevención y corrección.

Presenta la particularidad de que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software y como su enfoque está basado en modelos utiliza un lenguaje bien definido para tal fin, el UML. [1]

XP (eXtreme Programming).

Extreme Programming (XP): Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizada para proyectos de corto plazo. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

Características de XP, la metodología se basa en:

- **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que se pueda hacer pruebas de las fallas que pudieran ocurrir.
- **Refabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.

Lo fundamental en este tipo de metodología es:

- La comunicación, entre los usuarios y los desarrolladores.
- La simplicidad, al desarrollar y codificar los módulos del sistema.
- La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

[17]

Comparación entre RUP y XP.

- XP es un proceso ligero, significa, que los creadores del proceso ponen demasiadas tareas organizativas sobre los desarrolladores, RUP es un proceso pesado, basado mucho en la documentación, en la que no son deseables todos esos cambios volátiles.
- Mientras que el RUP intenta reducir la complejidad del software por medio de estructura y la preparación de las tareas pendientes en función de los objetivos de la fase y actividad actual, XP, como toda metodología ágil, lo intenta por medio de un trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción.

- XP es orientado al cliente y de iteraciones cortas y rápidas. También hay que decir que debido al carácter general de RUP, se consideran todos los demás procesos de desarrollo como casos particulares de este.
- RUP esta pensado para proyectos y equipos grandes, en cuanto a tamaño y duración. XP se implementa mejor para proyectos cortos y equipos más pequeños.
- RUP más escalable que XP, ya que a mayor tamaño de código y/o equipo mayor es la necesidad de cierta organización.
- RUP y XP, ambos describen los requerimientos de la aplicación desde el punto de vista del usuario. Ambos definen los requisitos técnicos sin meterse con detalles de implementación. **[18]**

Después de analizar las características de RUP y compararlas con otras metodologías de desarrollo de software y muy de cerca con XP, se decidió que RUP fuera la metodología de desarrollo seleccionada por todo lo antes explicado.

1.6.7 UML (Unified Modeling Language)

UML es el lenguaje de modelado que utiliza RUP.

“UML son las siglas de Unified Modeling Language (Lenguaje Unificado de Modelado), con que se construyen sistemas por medio de conceptos orientados a objetos”. **[1]**

“El UML (Lenguaje Unificado para la Construcción de Modelos) se define como un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software...” **[1]**. “Es un sistema notacional (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelado que utilizan conceptos orientados a objetos”.

EL UML (Lenguaje Unificado de Modelado) es una de las herramientas más utilizadas en el mundo del desarrollo de software, esto se debe a que permite a los desarrolladores de sistemas generar diseños que capturen sus ideas en una forma convencional y comunicarlas a terceras personas. Básicamente el UML es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

“El UML es un lenguaje para construir modelos; no guía al desarrollador en la forma de realizar el análisis y diseño orientados a objetos ni le indica cual proceso de desarrollo a adoptar.” [1]

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. Diagramas de Casos de Uso para modelar los procesos del negocio.

1.6.8 Herramienta CASE.

Rational Rose.

Es la herramienta CASE (Computer Aided Software Engineering), desarrollada por los creadores de UML que cubren todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes y certificación de las distintas fases. Permite una trazabilidad real entre modelo (análisis y diseño) y el código ejecutable.

Rational Rose domina el mercado de herramientas para el análisis, modelamiento, diseño y construcción orientada a objetos, tiene todas las características que los desarrolladores, analistas, y arquitectos exigen – soporte UML incomparable, desarrollo basado en componentes con soporte para arquitecturas líderes en la industria y modelos de componentes, facilidad de uso e integración optimizada.

La corporación Rational Rose ofrece el Proceso Unificado de desarrollo (RUP), que unifica las mejores prácticas de muchas disciplinas en un consistente y completo proceso del ciclo de vida, que permite al equipo de desarrollo disminuir los tiempos de liberación, además de hacer más predecible el software que ellos producen. Este proceso está basado en el Lenguaje Unificado de Modelación (UML – estándar de la industria) y únicamente integrado a herramientas líderes en el desarrollo de software de Rational, el Proceso Unificado de Rational apoya el equipo completo de desarrollo de software con guías detalladas e información crítica aplicable a la mayoría de las aplicaciones de la industria.

Rational Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros del equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas es que utiliza la notación estándar en la arquitectura de software (UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

Visual Paradigm

Visual Paradigm es una herramienta CASE que utiliza UML como lenguaje de modelaje.

Visual Paradigm ofrece:

- ✓ Entorno de creación de diagramas para UML 2.0
- ✓ Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- ✓ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✓ Capacidades de ingeniería directa (versión profesional) e inversa.
- ✓ Disponibilidad en múltiples plataformas **[19]**

¿Por qué Rational Rose?

Al realizar un estudio comparativo con otras herramientas CASE en particular con el Visual Paradigm se decidió utilizar el Rational Rose porque es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros del equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Siendo el Visual Paradigm una herramienta con características similares pero las computadoras de las que se disponen tienen muy poca memoria RAM por lo que su uso proporciona lentitud en el trabajo, lo que trae consigo demora durante todo el proceso de aplicación del mismo.

1.7 Fundamentación de la tecnología a utilizar.

Finalmente se ha llegado a la conclusión de que el sistema se desarrollará sobre la plataforma **Java Server Page (JSP)**, usando como lenguaje de programación **Java**, y como gestor de bases de datos, se usará el **PostgreSQL**. Además se determinó hacer uso del **Rational Unified Process (RUP)** puesto que es un proceso de desarrollo de software que enriquece la productividad en equipo y proporciona prácticas óptimas de software a todos los miembros del equipo. RUP es un mentor fácil de usar que hace el proceso práctico brindando amplias guías, plantillas y ejemplos. Se integra estrechamente con herramientas Rational, permitiendo a los equipos de desarrollo aprovechar todas las ventajas de las características de los productos Rational y el Lenguaje unificado de modelado (**UML**) como lenguaje utilizado para la metodología.

1.8 Conclusiones

En el presente capítulo se han analizado las tecnologías actuales y se profundizó en algunos conceptos necesarios para la comprensión de la solución de este trabajo y la necesidad de la creación de una nueva aplicación después de haber analizado las existentes que realizan funcionalidades similares a las que el cliente necesita. Además se ha fundamentado la elección de cuáles herramientas se utilizarán para el desarrollo de la aplicación, teniendo en cuenta que se necesita una aplicación multiplataforma y documentación para posibles versiones futuras.

Capítulo 2

Características del sistema.

Introducción.

En el presente capítulo se describe el objeto de estudio, problemas existentes, se especifican los detalles de la construcción de la herramienta y la solución propuesta utilizando los componentes del modelo de dominio de la metodología RUP. De este modelo se tendrán en cuenta la definición de las entidades y los conceptos principales, así como su representación gráfica, también se conocerán las reglas del negocio. Además se describen los requisitos funcionales y no funcionales del sistema, los actores que intervienen en el sistema, así como los diagramas de casos de uso.

2.1 Objeto de estudio.

De un análisis relativamente sencillo se puede definir dentro de la Biología Molecular varias esferas de trabajo, una de estas esferas es la Genética. El sistema a desarrollar tiene como **objeto de estudio** lo referente al desarrollo de aplicaciones informáticas Cliente-Servidor con interfaz Web.

El sistema que se propone no tiene relación con ningún otro sistema; además en la actualidad no existe ningún software que se ajuste exactamente a las exigencias que requieren las investigaciones que se están realizando en el CIGB por parte de los especialistas del área de Bioinformática.

2.1.1 Objetivos estratégicos de la organización.

Las diversas compañías u organizaciones que centran sus trabajos investigativos al estudio generalizado de genes, como por ejemplo, en Cuba el Centro de Ingeniería Genética y Biotecnología (CIGB) son instituciones de desarrollo dinámico que les ha permitido alcanzar un alto nivel en las investigaciones científico-tecnológicas dirigidas al desarrollo, producción y comercialización de productos biológicos obtenidos a través de los métodos de la Biotecnología moderna. En estos centros se llevan a cabo investigaciones importantes en el campo de la Genética, Biotecnología y la Bioinformática.

Actualmente estos centros son muy importantes y reconocidos, debido a que asumen la responsabilidad de contribuir directamente en el desarrollo científico-social y económico de la comunidad científica de hoy en día.

2.1.2 Flujo actual de los procesos.

Actualmente los procesos fluyen de la siguiente manera:

En el CIGB el diseño de siRNA, se realiza a través de los sitios de Internet que están disponibles en la WWW, para este proceso los especialistas de la institución que trabajan en estas investigaciones tienen que formar una secuencia de mRNA, que es el parámetro de entrada que aceptan los sitios; para formarla hay que realizar un proceso engorroso de trabajo con base de datos, donde tienen que hacer varias consultas y anotaciones para poder llegar a obtener la secuencia que quieren analizar, luego de tener esta, tienen que acceder a los sitios disponibles, entrar dicha secuencia para alcanzar los resultados que luego serán analizados, en estas herramientas online hay que especificar reglas de análisis, tales como: por ciento de C/G(bases nitrogenadas), longitud de las secuencias que se obtienen como resultado y la posición de inicio y fin del segmento de la cadena que se quiere analizar.

2.1.3 Análisis crítico de la ejecución de los procesos

Los procesos de estudio e investigaciones sobre análisis de secuencias para el silenciamiento de genes a través de siRNA que son llevados a cabo actualmente, poseen una estructura de realización y seguimiento que no resuelven todos los problemas y necesidades que se originan sobre este marco de trabajo, ni se hacen con la rapidez y la eficiencia que exigen y necesitan.

Primeramente se comenzará analizando la vía por la que se canalizan estos procesos, se realizan por vía Internet, y se cuenta con una red de una velocidad bastante aceptable como para poder acceder a estos sitios, pero que puede llegar a saturarse por la cantidad de recursos e información que se intercambia constantemente, además de que la información que encontramos en la WWW es bastante variable, lo que implica que hoy se pueda encontrar una información o una herramienta disponible y al día siguiente puede no estar o haber sido modificada, lo que trae consigo inseguridad cosa que no puede existir en investigaciones de esta índole, donde entran en juego el futuro de las grandes aspiraciones de la Biología Molecular en cuanto a descubrimientos de nuevas soluciones a enfermedades y problemas que presenta la población mundial.

Haciendo un análisis de la cantidad de información y recursos que necesitan los especialistas para lograr sus propósitos en estos trabajos, se nota que existe gran ineficiencia en los posibles resultados que hayan alcanzado, pues teniendo en cuenta que si se hace uso de un software o herramienta que esté en sitios online, estos casi nunca cumplen con todas las necesidades de los usuarios pues no se saben los algoritmos que estos implementan, ni si están bien implementados, y por supuesto no dan la posibilidad de escoger entre estos algoritmos, pues existen varios y unos mejores que otros; no se pueden especificar las reglas para el análisis y poder obtener así varias soluciones y hacer comparaciones entre estos resultados con diferentes especificaciones, en lo referente a la secuencia de mRNA que reciben como entrada haría falta prácticamente otro sistema que la conforme, por lo difícil que resulta el trabajo que están llevando a cabo con la base de datos de una forma que además de ser muy lenta no está totalmente automatizada porque no poseen interfaces que permitan el trabajo con la misma .

Como máximo los sitios disponibles solo resuelven un poco menos del 50% de los problemas, de cada solicitud debe obtenerse el resultado esperado y no una mera aproximación de los mismos.

Es necesario señalar que los especialistas que centran parte de su trabajo en este tema, necesitan saber los resultados precisos y concretos del análisis de las secuencias de genes en todo momento, pues les permitirán realizar estudios, comparaciones y llevar a cabo una posible aplicación de estos diseños de siRNA para el silenciamiento de algunos tipos de genes, estos trabajos investigativos se les ha dificultado bastante por no tener una herramienta que les permita estudiar los genes de forma minuciosa, con la propuesta que se pretende desarrollar, este problema quedaría resuelto.

2.2 Procesos objeto de automatización.

En el proceso de estudio e investigaciones sobre el análisis de secuencias mRNA para el diseño siRNA se desea automatizar todo lo referente a la información de los genes, cromosomas, así como las reglas que se necesitan para el análisis teniéndose en cuenta para el diseño de siRNA, que ayudarán a resolver las disyuntivas que se plantean en este trabajo.

Primeramente para realizar el análisis, se necesita entrar una secuencia de mRNA, la cual se desea analizar; o bien analizar una cadena formada a partir de informaciones recopiladas en una base de datos.

Para formar esta cadena es imprescindible especificar el Id del gen, con esta información se busca en la base de datos el gen especificado, luego se buscan los transcritos asociados a ese gen y se muestran,

para que a partir de este paso se pueda dar la opción al especialista de escoger el o los transcritos para los que se desea hacer el análisis, donde buscará los exones asociados a los transcritos seleccionados y a su vez poder mostrarlos. A partir de todos estos procedimientos llevados a cabo es cuando se procede a cargar un fichero de cromosomas con el cual se va a formar una cadena con los exones asociados al transcrito o transcritos seleccionados por el especialista, cuyos exones están formados por segmentos de ese cromosoma.

Después de conocer la secuencia de mRNA que se necesitan, se comienza con el análisis, donde se especifican las reglas antes de obtener el diseño de siRNA, importante condición para lograr resultados con la amplitud y diversidad que el especialista desee.

Las reglas implementadas para el análisis de mRNA que influyen en el diseño de siRNA son:

- Comienzan con AA.
- Contenido de CG de 30-50%
- 3 o más A/T entre las bases 15 y 19.
- Una A en la posición 19.
- Una A en la posición 3.
- Una U en la posición 10.
- Ausencia de GC en la posición 19.
- Ausencia de G e n la posición 13.

Al realizar el análisis se concluye con la obtención de los sitios blancos de la cadena de genes y una representación gráfica de los mismos.

Todos los resultados que se obtengan de los análisis que se realicen, incorporan aportes en las investigaciones que se hagan en cuanto al tema tratado.

Cuando se efectúa el diseño de siRNA para genes diferentes y con distintas reglas seleccionadas para el análisis, esto permite que puedan utilizarse los resultados para llegar a conclusiones importantes en las investigaciones sobre las características, funcionalidad y estructura que pueden tener un gen en particular y pensar en la posibilidad de silenciar este gen después de hacer las comparaciones necesarias, además

incorporar si resulta satisfactorio un determinado beneficio en los estudios que se hacen del genoma humano.

Como primer paso se pretende dar culminación a las actividades que hacen funcional y extensible al sistema, de manera que sea flexible a cambios estructurales y a múltiples interfaces, haciendo posible el trabajo y las investigaciones de forma independiente y de manera concurrente, garantizando la seguridad e integridad de la información. El objetivo final es obtener un sistema que englobe todas las actividades de los procesos, que brinde además una interfaz de configuración con facilidades para la utilización de los especialistas, para así lograr automatizar de la forma más idónea y flexible el sistema de análisis de secuencias mRNA para el diseño de siRNA.

2.3 Propuesta del sistema.

2.3.1 Descripción de la solución propuesta del sistema.

Con el objetivo de mejorar la calidad en la investigación aplicada y el desarrollo del trabajo sobre la materia, así como darle solución a los problemas mencionados se ha decidido desarrollar un sistema que se centra en cómo llevar a cabo un proceso tan importante para los especialistas que inclinan sus investigaciones al estudio de los genes, y en lógica consecuencia la posibilidad de lograr un diseño de siRNA más estable y eficiente, que les traiga consigo significativos aportes en los descubrimientos. Para lograr esto se hace necesario realizar muchos cambios dentro del proceso actual, dando como resultado una propuesta del sistema que brindará amplias facilidades para almacenar, procesar, manipular y proteger la información que se maneja sobre el análisis de secuencias mRNA para el diseño de siRNA, importante para el desarrollo de grandes investigaciones y aportes a la Biología Molecular. A continuación se mencionan algunos de los servicios que la herramienta brindará:

Gestionar autenticación de usuarios: Permitirá solo el acceso al sistema del personal autorizado y los usuarios deberán autenticarse antes de acceder al mismo.

Registrar usuarios: Permitirá a los especialistas crearse nuevas cuentas para el acceso a la aplicación informática para el diseño de siRNA.

Formar secuencias mRNA: Permitirá al especialista conocer la secuencia mRNA a partir de la especificación de un criterio (id gen, símbolo del gen o nombre del transcrito) para obtener toda la

información necesaria para generar las secuencias de mRNA para un análisis que posteriormente se llevará a cabo.

Analizar secuencias mRNA: Permitirá analizar las secuencias mRNA, ya sea la secuencia entrada o formada. Brindará la posibilidad a los especialistas de seleccionar las reglas para el análisis y obtener el diseño de siRNA. Además posibilitará ver los sitios blancos de la secuencia, así como las subcadenas, la posición inicial y final de los mismos. También posibilitará ver la secuencia con los sitios blancos resaltados (color rojo) y una gráfica donde se muestre una representación de los sitios blancos en la secuencia.

Mostrar genes: Permitirá a los especialistas conocer información suficiente de los genes, como por ejemplo: los transcritos asociados a un determinado gen, sin necesidad de ir a consultar los resultados de un análisis previo de una secuencia de mRNA.

Mostrar transcritos: Permitirá a los especialistas conocer información suficiente de los transcritos, donde se incluyen además los datos de exones asociados a un transcrito y una grafica que resume la representación de los exones en el transcrito, toda esta información es brindada sin necesidad de consultar los resultados de un análisis previo de una secuencia de mRNA.

2.3.2 Análisis comparativo de otras soluciones existentes con la propuesta.

Teniendo en cuenta otras soluciones existentes, las cuales son accesibles a través de Internet en los sitios online, se puede decir que la propuesta que brinda esta investigación es la recomendada y la más aceptada por las siguientes razones:

- Independiza a los clientes del uso de Internet.
- Se tienen en cuenta todos los aspectos y características que requieren los clientes para que la herramienta cumpla con sus necesidades.
- Brinda un alto nivel de confiabilidad, cosa muy importante para mantener seguros y confiables los proyectos que se realicen.
- Cuenta con una conexión a base de datos segura y con gran capacidad de almacenamiento, para lograr tener guardados los resultados de los estudios que realicen y realizar consultas, posibilidad que no han obtenido.

- Brindará aportes no solo en nuestro país sino a nivel mundial, incorporando de esta manera una herramienta global que le permita a todos los investigadores de dicha rama hacer análisis de secuencia para el diseño de siRNA, de esta forma se sigue la política de socialización del conocimiento.

2.4 Modelo de Dominio.

Un Modelo del Dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan los eventos que suceden en el entorno en el que trabaja el sistema.

La modelación del dominio tiene como objetivo fundamental la comprensión y descripción de las clases más importantes en el sistema.

Debido a que se hace difícil encontrar una estructura de los procesos de negocios que tienen que ver con el objeto de estudio, y haciéndose necesario poder entender el funcionamiento de la aplicación, se necesita describir mediante una serie de conceptos, entidades y sus relaciones, agrupados en un modelo del dominio para el logro de un fácil entendimiento de la aplicación.

2.4.1 Reglas del negocio.

Las reglas del negocio permiten definir los límites lógicos de las aplicaciones. En ellas las aplicaciones reflejan las restricciones que existen, evitando la realización de acciones no válidas. A continuación se enumeran las reglas definidas para el sistema:

- Tener la información necesaria para el análisis de la secuencia de mRNA.
- Seleccionar las reglas de cumplimiento para obtener el análisis de la secuencia de mRNA.

2.4.2 Definición de las entidades y los conceptos principales.

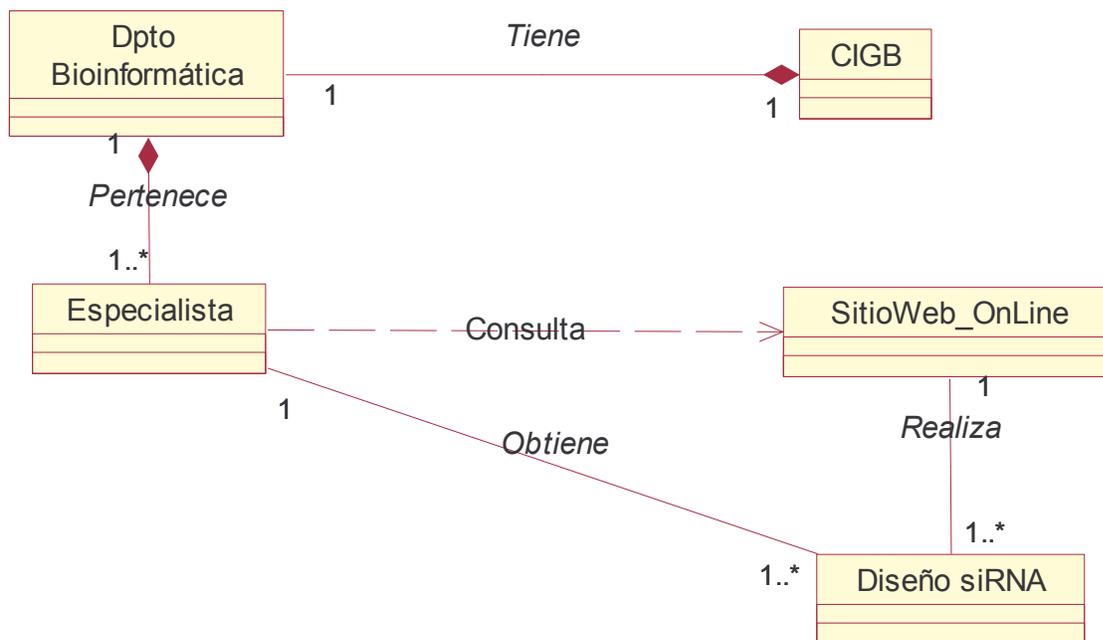
Con la aplicación de Modelo del dominio se detectaron las siguientes entidades y conceptos (objetos):

- **Especialista:** Persona capacitada para interactuar con la aplicación.
- **CIGB:** Centro de Ingeniería Genética y Biotecnología. Centro que realiza importantes investigaciones en las ramas que su nombre indica.
- **Dpto. Bioinformática:** Departamento que automatiza los procesos biológicos que forman parte de los

análisis que realizan los especialistas, en el mismo se desarrollan herramientas que permiten una rápida respuesta ante las exigencias del centro y se le facilita el trabajo a los mismos.

- **Sitios Web _ online:** Sitios disponibles en Internet para el diseño de siRNA.
- **Diseño siRNA:** Es el resultado del análisis de secuencias mRNA.

2.4.3 Diagrama Modelo del Dominio.



2.5 Requerimientos Funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. Además permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el sistema.

Los requerimientos funcionales del software propuesto son los siguientes:

R1: Validar entrada de usuarios al sistema.

R2: Registrar usuarios.

2.1: Crear nuevas cuentas de usuarios.

R3: Formar secuencia mRNA.

3.1: Especificar criterio de búsqueda.

3.2: Buscar gen especificado.

3.3: Escoger nombre transcrito para el que se desea hacer el análisis.

3.4: Cargar fichero de cromosomas asociados al transcrito.

3.5: Formar cadena con los exones asociados al transcrito.

3.6: Mostrar id del transcrito.

3.7: Mostrar posición inicial y final del transcrito.

3.8: Mostrar id del gen y símbolo.

3.9: Mostrar cromosoma al que pertenecen el transcrito y el gen.

3.10: Mostrar secuencia de nucleótidos (mRNA).

R4: Realizar análisis de las secuencias mRNA.

4.1: Seleccionar reglas.

4.2: Buscar sitios blancos en la secuencia mRNA.

4.3: Mostrar subcadena de los sitios blancos.

4.4: Mostrar posición inicial y final de sitios blancos.

4.6: Muestra la secuencia mRNA resaltando sitios blancos.

4.5: Mostrar grafica de la secuencia de mRNA resaltando sitios blancos.

R5: Mostrar genes.

5.1: Mostrar id y símbolo del gen.

5.2: Buscar transcritos asociados al gen.

5.3: Mostrar datos de los transcritos asociados al gen.

R6: Mostrar transcrito.

6.1: Buscar exones asociados al transcrito.

6.2: Mostrar exones asociados al transcrito.

6.3: Mostrar posición inicial y final de los exones asociados al transcrito.

6.4: Mostrar id y nombre del transcrito.

6.5: Muestra gráfica de exones asociados al transcrito.

2.6 Requerimientos No Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Representan las características del producto.

Apariencia o interfaz externa:

La aplicación está diseñada con una interfaz amigable, de forma tal que el usuario haga uso de la misma sin dificultad alguna, ajustándose a los estándares establecidos para el desarrollo de un buen diseño.

Usabilidad:

El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora, solo se necesita contar con conocimientos especializados en biología para entender los resultados dados por la aplicación.

Rendimiento:

Al estar concebida para un ambiente cliente/servidor, se trata de garantizar la rapidez de respuesta del sistema ante las solicitudes de los usuarios, al igual que la velocidad de procesamiento de la información. Para lo cual se realiza la validación de los datos y la manipulación de eventos en el cliente y en el servidor aquellas que por cuestiones de seguridad, o de acceso a los datos lo requieran. Lográndose así un tiempo de respuesta más rápido, una mayor velocidad de procesamiento, y un mayor aprovechamiento de los recursos.

Soporte:

El sistema debe propiciar su mejoramiento y la anexión de otras opciones que se le incorporen en un futuro. Además debe ser de fácil instalación, adaptable a numerosas plataformas y de fácil mantenimiento.

Portabilidad:

El sistema puede ser ejecutado sobre los sistemas operativos Linux y Windows, por su característica de ser multiplataforma.

Seguridad:

El especialista tiene que autenticarse para poder acceder al sistema.

Evaluar mecanismos de tolerancias a faltas. Predicción de fallos.

Protección contra los fallos. El sistema debe ser capaz de en pocos segundos recuperarse de un fallo de una operación.

Confiabilidad:

El sistema debe ser confiable y preciso en la información que le suministra al usuario para evitar cualquier tipo de error.

Software:

-Se debe disponer de Linux, Windows 95 o versiones superiores para la instalación de la aplicación. La aplicación se realizará en un ambiente Web, la Base de Datos es independiente de la aplicación.

Hardware:

Para el desarrollo y puesta en práctica del proyecto se requieren máquinas con los siguientes requisitos:

- Procesador Pentium 3 o superior.
- 256 MB de RAM.
- Un mínimo de 2 GB de espacio disponible en disco duro.

2.7 Definición de los actores y casos de uso del sistema a automatizar.

En este epígrafe se abordarán los casos de uso del sistema y su descripción detallada, definiendo además los actores que intervienen en los mismos.

2.7.1 Actores

Un actor es una entidad externa del sistema que de alguna manera participa en la historia del caso de uso. Siempre se beneficia de la realización de un caso de uso. Estimula el sistema con eventos de entradas o recibe algo de él. O sea, es un rol de un usuario, que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema.

Tabla # 1. Definición de actores del sistema a automatizar.

Nombre del Actor.	Descripción.
Especialista.	Representa el usuario que hará uso del sistema, teniendo la posibilidad de interactuar con todas la funcionalidades de este.

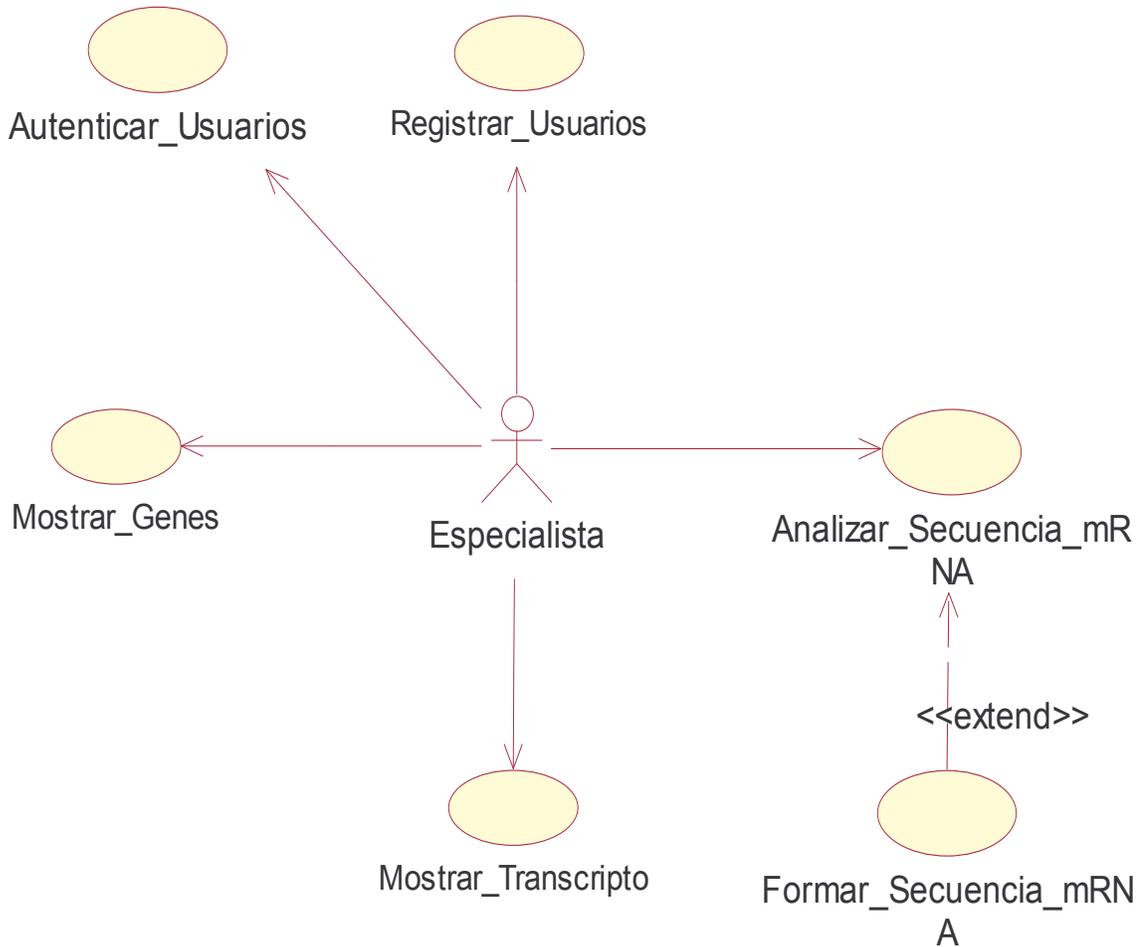
2.7.2 Casos de Uso Definidos.

Los casos de uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Son una secuencia de acciones que el sistema debe llevar a cabo, interactuando con sus actores donde se obtiene un resultado de valor para los mismos. Se utilizan para obtener información de cómo debe trabajar el sistema, describen bajo la forma de acciones y reacciones el comportamiento de un sistema desde el punto de vista del usuario.

Los casos de usos definidos son los siguientes:

- Autenticar usuarios.
- Registrar Usuarios.
- Formar secuencia de mRNA.
- Analizar secuencia de mRNA.
- Mostrar genes.
- Mostrar transcrito.

2.7.3 Diagrama de casos de uso del sistema a automatizar.



2.7.4 Descripción de casos de usos expandidos.

Caso de Uso:	Autenticar usuarios.
Actores:	Especialista.
Propósito:	Realizar la autenticación del especialista, validarla y permitir el acceso del mismo al sistema.

Resumen:	El caso de uso se inicia cuando el especialista introduce su usuario y contraseña para poder acceder a las opciones que le sean permitidas en la aplicación una vez que se verifiquen estos datos.	
Referencia:	R1	
Precondiciones:	El especialista debe estar previamente registrado.	
Poscondiciones:	El sistema brinda la posibilidad al especialista de tener acceso general a la aplicación.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1) El especialista le solicita al sistema autenticarse.	1.1) El sistema brinda la opción para entrar datos requeridos para esta acción.	
2) El especialista llena los datos: usuario y contraseña.	2.1) El sistema receptiona la información. 2.2) El sistema chequea si el usuario existe. 2.3) Los datos del usuario son correctos. 2.4) El sistema verifica el privilegio de este usuario. 2.5) El sistema permite el acceso a la aplicación. 2.6) El sistema finaliza la ejecución del caso de uso.	
Flujos Alternos()		
Acción del Actor ()	Respuesta del Sistema	
2) El especialista introduce datos incorrectos.	2.1) El usuario no existe o sus datos son incorrectos. 2.2) El sistema le envía un mensaje de error al especialista indicándole que su usuario o contraseña son incorrectos. 2.3) El sistema brinda la opción de introducir nuevamente el usuario y contraseña.	
Prioridad:	Critico.	
Prototipo Interfaz		

Caso de Uso:	Registrar usuarios.
---------------------	---------------------

Actores:	Especialista.
Propósito:	Crear nuevas cuentas de usuarios.
Resumen:	El caso de uso se inicia cuando un especialista desea crearse una cuenta para hacer uso de las funcionalidades del sistema.
Referencia:	R2
Precondiciones:	
Poscondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1) El especialista accede a la opción de crear cuenta.	1.2) El sistema brinda la opción de introducir datos. <ul style="list-style-type: none"> • Usuario. • Contraseña. • e-mail. • País de residencia.
2) El especialista introduce los datos solicitados.	2.1) El sistema muestra un mensaje confirmando el registro. Finalizando la ejecución del caso de uso.
Flujos Alternos()	
Acción del Actor ()	Respuesta del Sistema
2) El especialista introduce datos existentes.	2.1) El sistema muestra un mensaje de error y brinda la posibilidad de realizar un nuevo registro. Finalizando la ejecución del caso de uso.
Prioridad:	Critico.
Prototipo Interfaz	

Caso de Uso:	Formar secuencia de mRNA.
Actores:	Especialista.
Propósito:	Formar la secuencia de mRNA para que la misma sea utilizada en el

	proceso de análisis que llevará a cabo la aplicación.
Resumen:	El caso de uso se inicia cuando el especialista escoge la opción “Diseño del siRNA”, especificando un criterio de búsqueda de mRNA, y a partir del mismo se busca en la base de datos la información necesaria para conformar la secuencia de mRNA correspondiente.
Referencia:	R3.
Precondiciones:	El usuario debe tener permisos para acceder a esta parte del sistema.
Poscondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1) El especialista selecciona la opción “Diseño del siRNA”.	1.1) El sistema brinda la opción de seleccionar un criterio de búsqueda (id _ gen, símbolo del gen y nombre del transcrito).
2) El especialista selecciona un criterio de búsqueda (id _ gen o símbolo del gen).	2.2) El sistema brinda la opción de introducir la información a partir del criterio de búsqueda seleccionado.
3) El especialista introduce la información que se solicita según el criterio de búsqueda seleccionado (id _ gen o símbolo del gen).	<p>3.1) El sistema realiza las acciones siguientes:</p> <ul style="list-style-type: none"> ➤ Busca el gen especificado. ➤ Busca transcrito asociado al gen. ➤ Muestra transcrito asociado al gen. ➤ Muestra id del transcrito. ➤ Muestra posición inicial y final del transcrito. ➤ Muestra id del gen y símbolo. ➤ Muestra el cromosoma donde se encuentra el transcrito y el gen. <p>3.2) El sistema brinda la posibilidad de introducir un criterio (nombre de transcrito) para generar la secuencia de nucleótidos (mRNA).</p>

4) El especialista introduce el nombre del transcrito.	4.1) El sistema muestra datos del gen (orientación, cromosoma al que pertenece). 4.2) El sistema muestra la secuencia mRNA generada y finaliza la ejecución del caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2) El especialista selecciona el criterio (nombre del transcrito).	2.2) Se realizan los puntos 4.1 y 4.2.
3) El especialista introduce datos incorrectos.	3.1) El sistema muestra un mensaje de error y brinda la posibilidad de introducir nuevamente el dato.
Prioridad: Prototipo Interfaz	Critico.

Caso de Uso:	Analizar Secuencia mRNA.
Actores:	Especialista.
Propósito:	Obtener un diseño de siRNA a partir de un adecuado análisis de la secuencia mRNA que se haya escogido.
Resumen:	El caso de uso se inicia cuando el especialista conoce la secuencia de mRNA y procede a analizarla, seleccionando las reglas para el análisis que se llevará a cabo, y de esa manera obtener el diseño de siRNA esperado.
Referencia:	1. R4. 2. CU Formar secuencia de mRNA (extensión).
Precondiciones:	El usuario debe tener permisos para acceder a esta parte del sistema.
Poscondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

1) El especialista selecciona la opción "Diseño siRNA".	1.1) El sistema muestra las opciones de "Generar secuencia mRNA" e "Introduzca la secuencia de nucleótidos (mRNA)".
2) El especialista accede a la opción "Generar secuencia mRNA".	2.1) El sistema genera la secuencia mRNA a partir de la información que se introdujo. 2.2) El sistema muestra la secuencia mRNA generada. 2.3) El sistema muestra las reglas para el análisis. 2.4) El sistema muestra la opción de "Analizar".
3) El especialista selecciona las reglas para realizar el análisis de la secuencia mRNA.	
4) El especialista escoge la opción de "Analizar".	4.1) El sistema realiza el análisis. 4.2) El sistema busca sitios blancos en la secuencia mRNA. 4.3) Muestra las subcadenas de sitios blancos. 4.4) Muestra posición inicial y final de los sitios blancos. 4.5) Muestra la secuencia mRNA resaltando sitios blancos. 4.6) Muestra una grafica de la secuencia mRNA resaltando sitios blancos y finaliza la ejecución del caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2) El especialista accede a la opción "Introduzca la secuencia de nucleótidos (mRNA)".	2.1) El sistema reconoce la secuencia mRNA. 2.2) Se realizan todos los puntos a partir 2.3 y 2.4.
Prioridad:	Critico.

Prototipo Interfaz	
---------------------------	--

Caso de Uso:	Mostrar Genes.
Actores:	Especialista.
Propósito:	Mostrarle información relevante a los especialistas.
Resumen:	El caso de uso se inicia cuando el especialista escoge la opción "Diseño del siRNA", y desea conocer más información sobre los genes, sin necesidad de realizar un análisis previo. Entonces accede al vínculo "Mostrar Genes" especificando un criterio de búsqueda, y a partir del mismo se busca en la base de datos la información necesaria para que sea consultada por los especialistas.
Referencia:	R5; R3
Precondiciones:	El usuario debe tener permisos para acceder a esta parte del sistema.
Poscondiciones:	

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1) El especialista selecciona la opción "Diseño siRNA".	1.1) El sistema brinda varias opciones.
2) El especialista accede a la opción de "Mostrar Genes".	2.1) El sistema brinda la opción de especificar un criterio (id gen, símbolo gen) para buscar los transcrito asociados a un gen.
3) El especialista selecciona el criterio (id gen, símbolo gen).	3.1) El sistema brinda la opción de introducir la información a partir del criterio de búsqueda seleccionado.
4) El especialista introduce el criterio (id gen, símbolo gen) y escoge la opción "Buscar".	4.1) El sistema muestra id y símbolo del gen. 4.2) El sistema busca transcritos asociados al gen. 4.3) El sistema muestra datos de los transcritos asociados al gen.

	4.4) El sistema finaliza así la ejecución del caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
4) El especialista introduce datos incorrectos.	4.1) El sistema muestra un mensaje de error y brinda la posibilidad de introducir nuevamente el dato. 4.2) Se realizan todos los puntos a partir del 4.1.
Prioridad: Prototipo Interfaz	Critico.

Caso de Uso:	Mostrar transcrito.
Actores:	Especialista.
Propósito:	Mostrarle información relevante a los especialistas.
Resumen:	El caso de uso se inicia cuando el especialista escoge la opción "Diseño del siRNA", y desea conocer más información sobre los transcritos, sin necesidad de realizar un análisis previo. Entonces accede al vínculo "Mostrar transcritos" especificando el nombre del transcrito, y a partir del mismo buscar en la base de datos la información necesaria para que sea consultada por los especialistas.
Referencia:	R6
Precondiciones:	El usuario debe tener permisos para acceder a esta parte del sistema.
Poscondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1) El especialista selecciona la opción "Diseño siRNA".	1.1) El sistema brinda varias opciones.

2) El especialista accede a la opción de “Mostrar transcrito”.	2.1) El sistema brinda la opción de introducir el nombre del transcrito.
3) El especialista introduce el nombre del transcrito.	3.1) El sistema busca exones asociados al transcrito. 3.2) El sistema muestra exones asociados al transcrito. 3.3) El sistema muestra posición inicial y final de los exones asociados al transcrito. 3.4) El sistema muestra id y nombre del transcrito. 3.5) El sistema muestra gráfica de los exones asociados al transcrito y finaliza la ejecución del caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3) El especialista introduce datos incorrectos.	3.1) El sistema muestra un mensaje de error y brinda la posibilidad de introducir nuevamente el dato. 3.2) Se realizan todos los puntos a partir del 3.1.
Prioridad: Prototipo Interfaz	Critico.

2.8 Conclusiones

Este capítulo ha brindado una clara definición de los requisitos que deberá cumplir el sistema, además se hizo una representación de las funcionalidades a construir mediante el diagrama de casos de uso, y finalmente se describieron paso a paso todas las acciones del actor del sistema con los casos de uso que interactúa.

El empleo de los modelos de casos de uso para describir el sistema propuesto permitió una adecuada captación y modelación de los requerimientos, demostrando su importancia en esta etapa. Se ganó claridad en cuanto a la concepción del sistema a construir y se sentaron las bases para las restantes fases del proceso de diseño e implementación del sistema.

Introducción.

En el presente capítulo se modelan los artefactos del análisis y diseño del sistema, utilizando para su modelado el Lenguaje Unificado de Modelado (UML). Se define los diagramas de clases e interacción para el análisis y diseño. Se tratan los principios del diseño del sistema, se explica el tratamiento de errores. En los últimos epígrafes, se diseña la Base de Datos y se realiza el diagrama de despliegue.

3.1 Modelo de clases de análisis del sistema.

El Diagrama de clase es el diagrama principal de diseño y análisis para un sistema. En él, la estructura de clases del sistema se especifica, con relaciones entre clases y estructuras de herencia. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal.

En el expediente del proyecto se muestran los diagramas de clases de análisis del sistema.

3.2 Propósitos del diseño.

Según el Proceso Unificado de Desarrollo los propósitos del diseño son [1]:

- Adquirir una comprensión en profundidad de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes utilizables, sistemas operativos, tecnologías de distribución y concurrencia, tecnologías de interfaz de usuario, tecnologías de gestión de transacciones.
- Crear una entrada apropiada y un punto de partida para actividades de implementación subsiguientes capturando los requisitos o subsistemas individuales, interfaces y clases.

El diseño es el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción. Esto contribuye a una arquitectura estable y sólida y crea un plano del modelo de implementación.

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar.

3.3 Principios de diseño.

UML posee una extensión para el modelado de aplicaciones Web, propuesta por Conallen [21], dicha extensión es usada para el diseño de las clases. Los estereotipos que usa esta extensión son:



nombre_clase

<<Server Page>> Representa la página Web que tiene código que se ejecuta en el servidor. Este código interactúa con recursos en el servidor. Las operaciones representan las funciones del código y los atributos las variables visibles dentro del alcance de la página.



nombre_clase

<<Client Page>> Una instancia de Página Cliente es una página Web, con formato HTML; mezcla de datos, presentación y lógica. Son interpretadas por el browser. Cada página cliente solo puede ser construida por una página servidor.



nombre_clase

<<Form>> Grupo de elementos de entrada que son parte de una página cliente. Se relaciona directamente con la etiqueta de igual nombre del HTML. Sus atributos son los elementos de entrada del formulario (Text Field, Text Area, Button, Label, Radio Button, Radio Group, Select, Check Box y Hidden Fields).

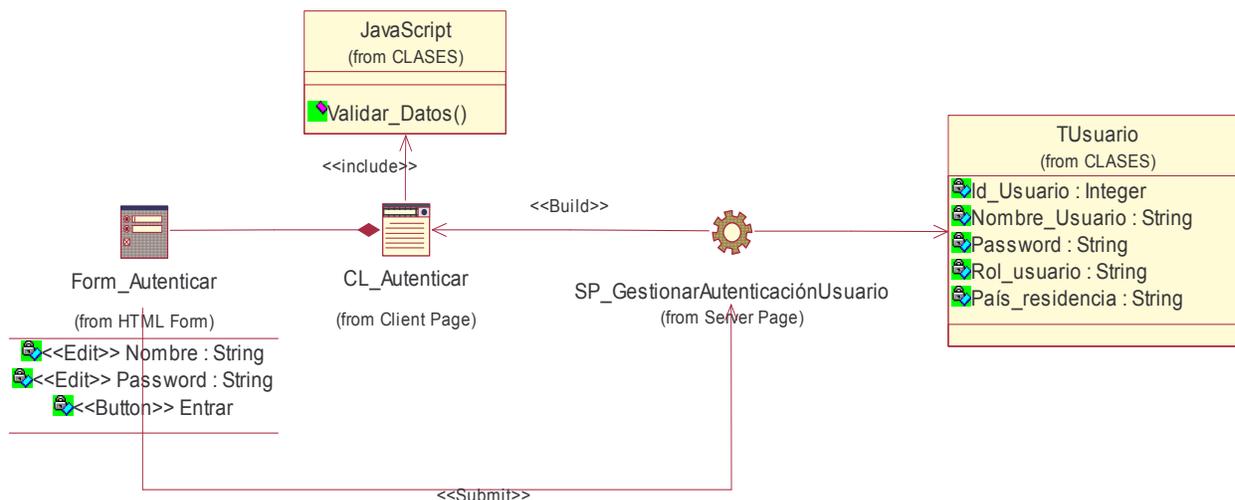
3.4 Diagramas de Clases Web.

A continuación se muestran los diagramas de clases del diseño por cada caso de uso, donde se describe gráficamente las especificaciones de las clases de software. Para ganar claridad en los diagramas se ocultaron los métodos de las clases entidades y controladoras.

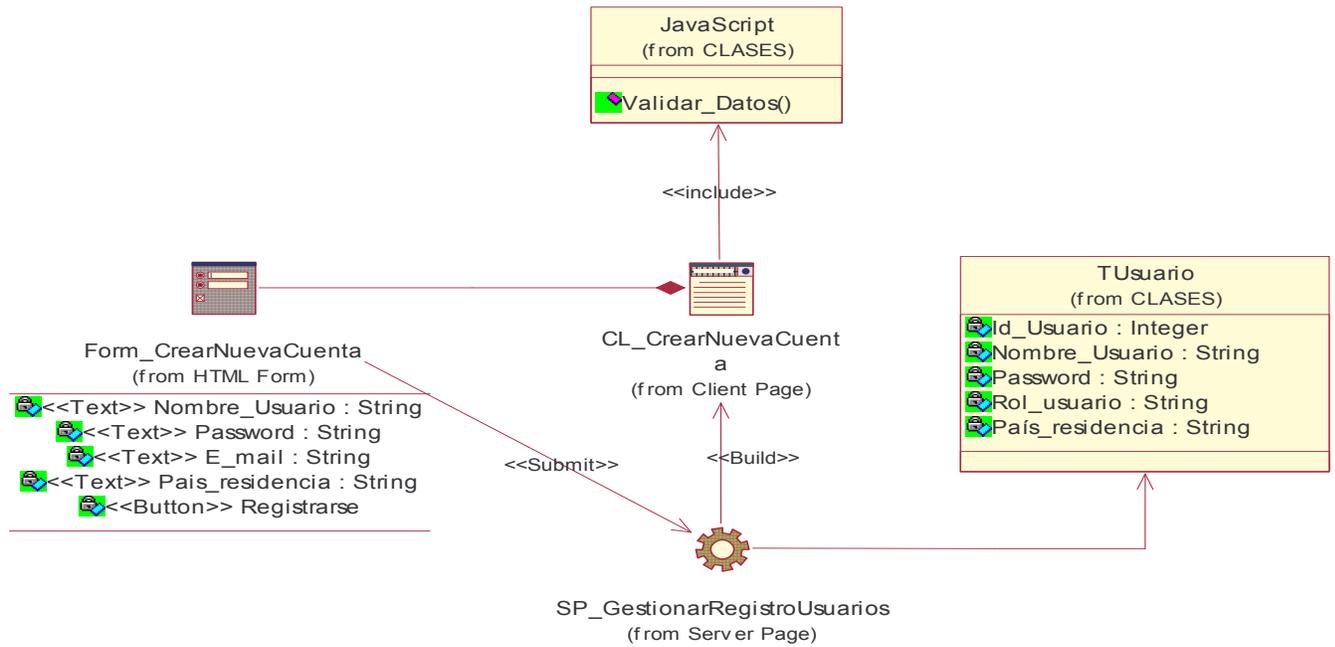
Los diagramas de clases contienen las siguientes informaciones:

- Clases, asociaciones y atributos.
- Métodos.
- Información sobre los tipos de atributos.
- Navegabilidad.
- Dependencias.

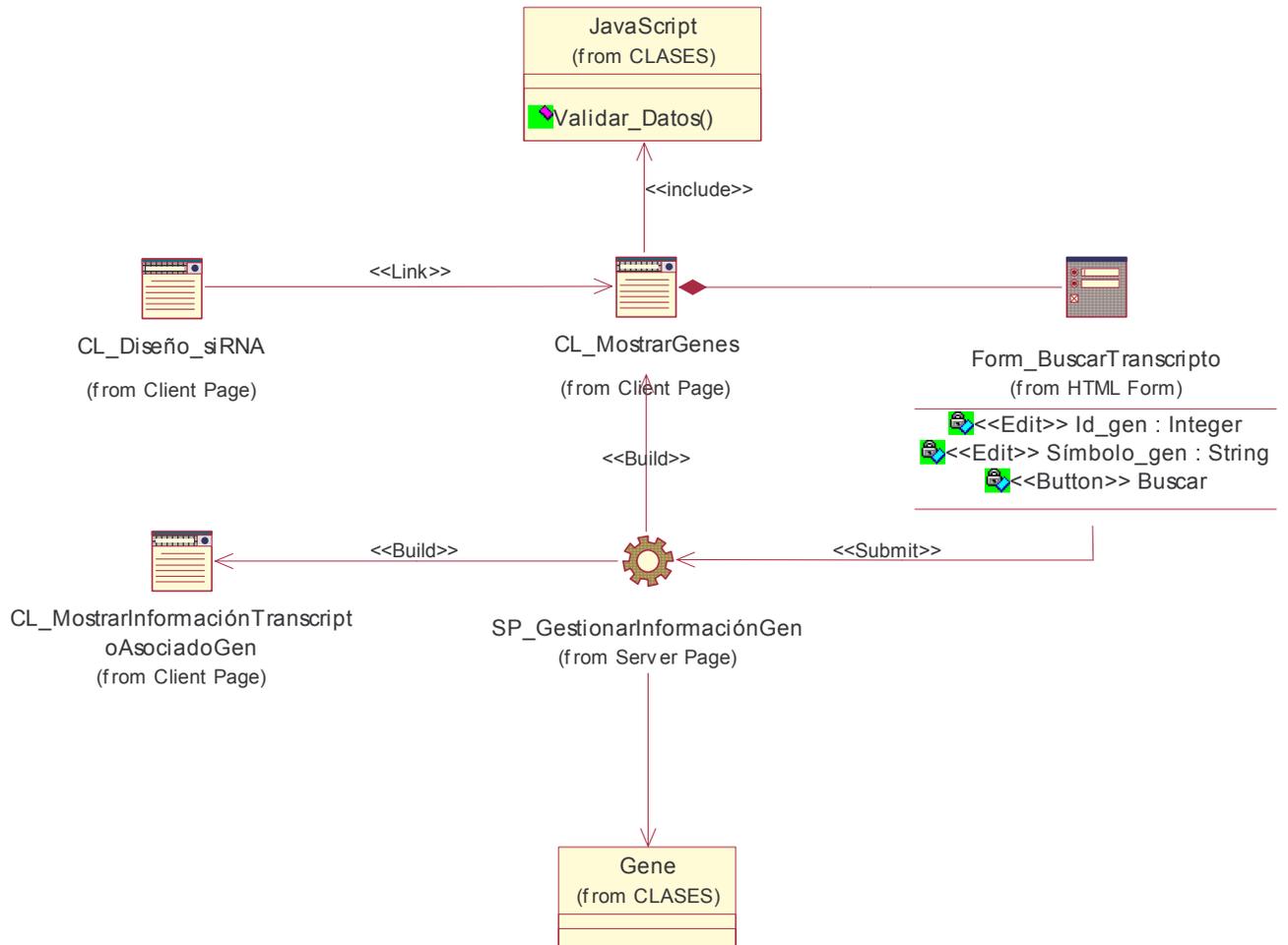
CU Autenticar_Usuario



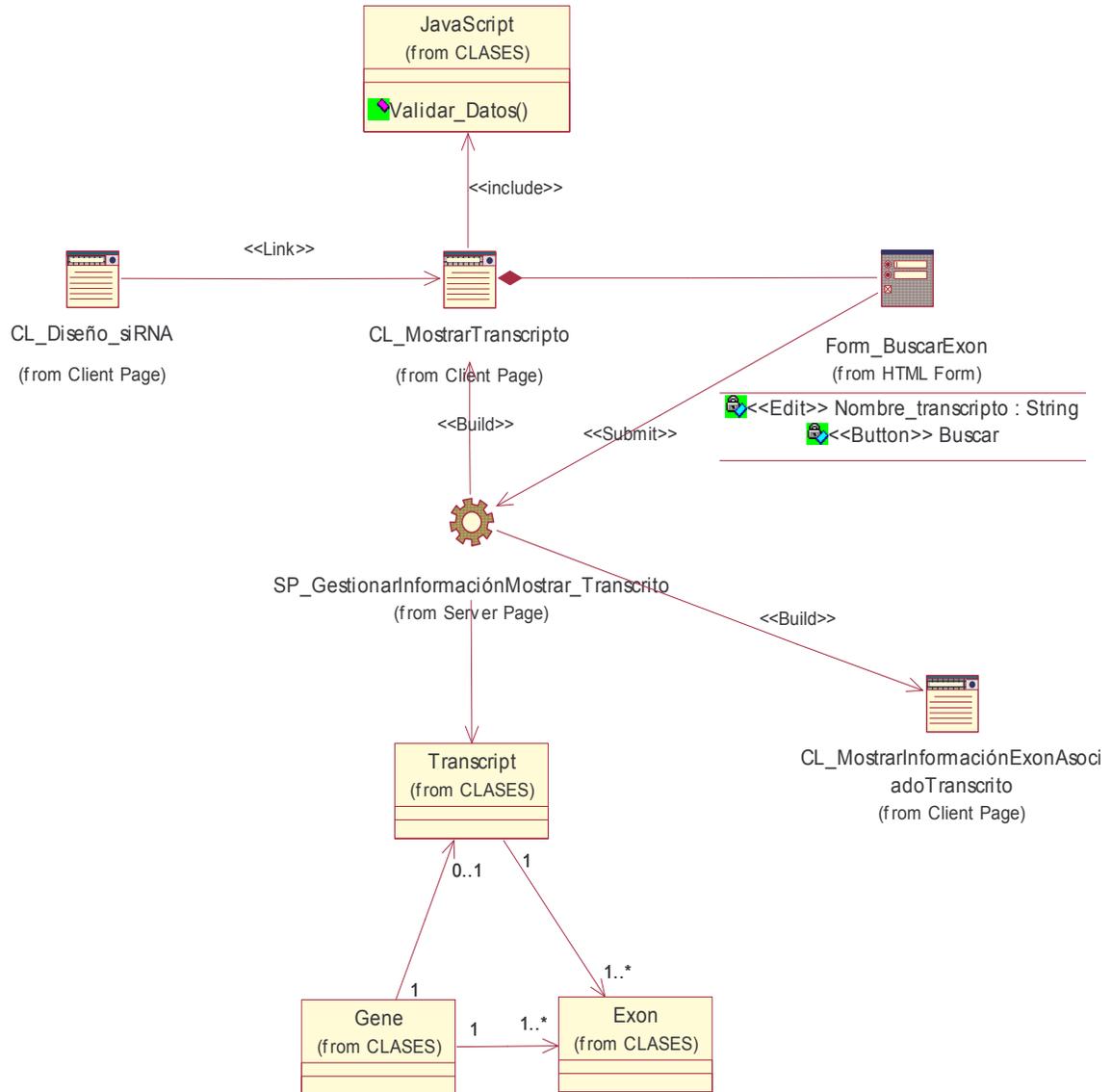
CU Registrar_Usuarios.



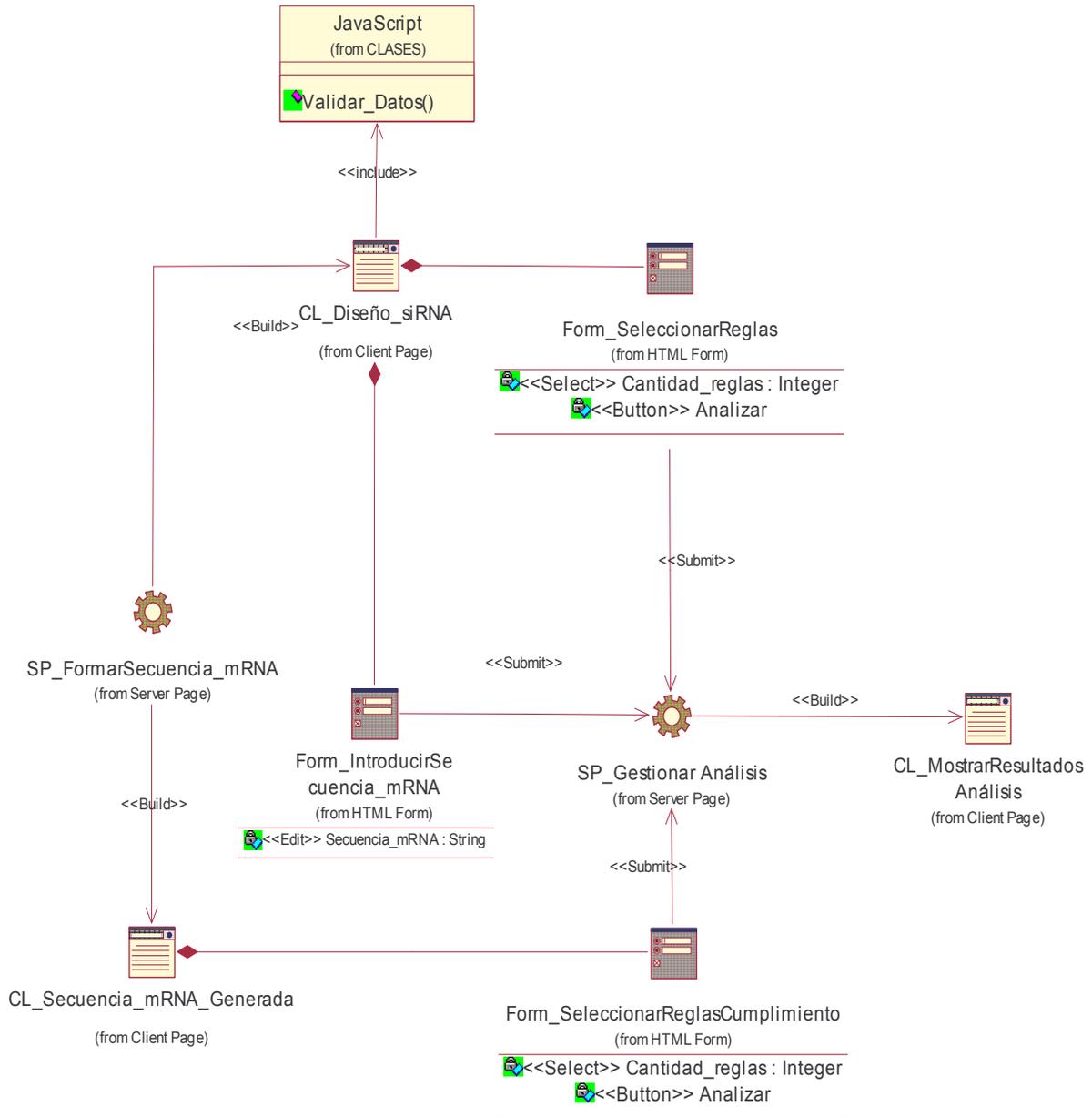
CU Mostrar_Genes.



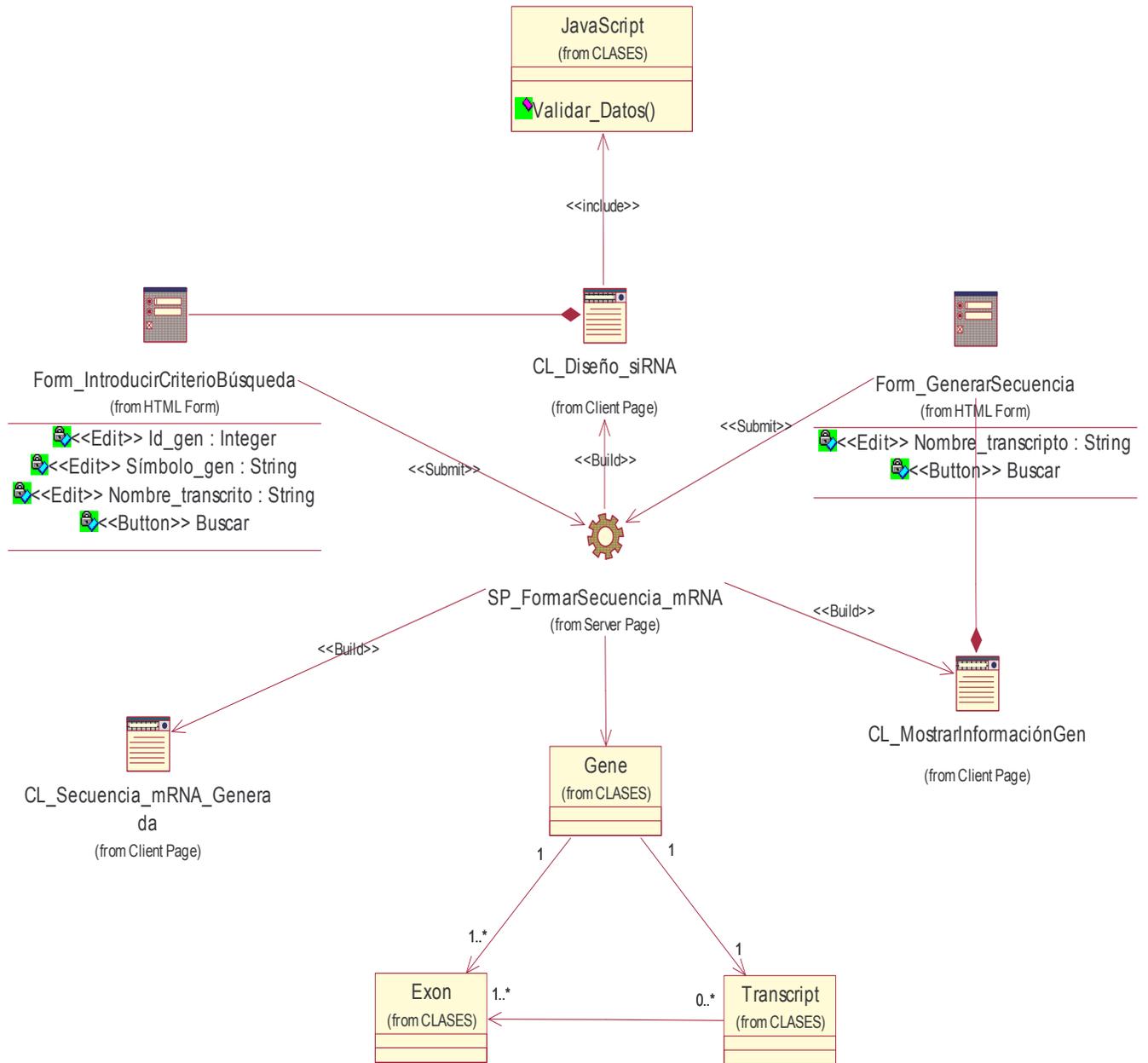
CU Mostrar_Transcrito.



CU Realizar_Análisis_Sec_mRNA.



CU Formar_Secuencia_mRNA



3.5 Descripción de las clases.

Clases Controladoras:

Nombre: GestionarAutenticaciónUsuario.	
Tipo de clase: Controladora.	
Atributo	Tipo
driver	String
url	String
user	String
contraseña	String
connection	Connection
Para cada responsabilidad:	
Nombre:	Init()
Descripción:	Cargar el driver en memoria y crear la conexión.
Nombre:	destroy()
Descripción:	En el método destroy() una vez que el servlet se esta apagando, se desconecta de la base de datos.
Nombre:	doGet (HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i> .
Nombre:	doPost (HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i> .
Nombre:	usuarioExistente()
Descripción:	Permite chequear si el usuario existe o no.
Nombre:	contador()
Descripción:	Permite contar las veces que existe un usuario.

Nombre: GestionarRegistroUsuarios.	
Tipo de clase: Controladora.	
Atributo	Tipo
driver	String
url	String
user	String
contraseña	String
connection	Connection
Existe	int
Para cada responsabilidad:	
Nombre:	Init()
Descripción:	Cargar el driver en memoria y crear la conexión.
Nombre:	destroy()
Descripción:	En el método destroy () una vez que el servlet se esta apagando, se desconecta de la base de datos.

Nombre:	doGet (HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i> .
Nombre:	doPost (HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i> .
Nombre:	comprobaruser()
Descripción:	Comprueba si existe o no el usuario.
Nombre:	contador()
Descripción:	Cuenta las veces que existe.

Nombre: Gestionar Información Mostrar_Transcrito.	
Tipo de clase: Controladora.	
Atributo	Tipo
driver	String
url	String
user	String
contraseña	String
connection	Connection
Para cada responsabilidad:	
Nombre:	init()
Descripción:	Cargar el driver en memoria y crear la conexión.
Nombre:	destroy()
Descripción:	En el método destroy() una vez que el servlet se esta apagando, desconectaremos de la base de datos
Nombre:	doGet(HttpServletRequest request, HttpServletResponse response)
Descripción:	Aquí se implementa el método <i>doGet</i> que se encargará de conectarse a la base de datos extraer los datos y mostrar el informe.
Nombre:	doPost(HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i> .
Nombre:	CrearPaginaHtml (HttpServletResponse response, ResultSet resultado)
Descripción:	Crea la página html con los datos del transcrito.
Nombre:	SacarPtos IF(nombre_transcrito)
Descripción:	Permite obtener los puntos inicial y final del transcrito.

Nombre: Gestionar Análisis.	
Tipo de clase: Controladora.	
Atributo	Tipo
driver	String
url	String
user	String
contraseña	String
connection	Connection

Para cada responsabilidad:	
Nombre:	init()
Descripción:	Cargar el driver en memoria y Crear la conexión
Nombre:	destroy()
Descripción:	En el método destroy () una vez que el servlet se está apagando, se desconecta de la base de datos.
Nombre:	doGet(HttpServletRequest request, HttpServletResponse response)
Descripción:	Aquí se implementa el método <i>doGet</i> que se encargará de conectarse a la base de datos extraer los datos y mostrar el informe
Nombre:	doPost(HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una llamada a <i>doGet</i> .
Nombre:	ADN()
Descripción:	Reserva memoria a ADN que es el objeto que va a hacer el análisis.
Nombre:	getCadena()
Descripción:	Devuelve la cadena.
Nombre:	setCadena(String cadena1)
Descripción:	Modifica la cadena.
Nombre:	getReglas()
Descripción:	Devuelve la 1ra regla.
Nombre:	setReglas(String regla)
Descripción:	Modifica la regla.
Nombre:	saberBaseNitrogenada(char l)
Descripción:	Identifica si es una base nitrogenada.
Nombre:	saber(char l)
Descripción:	Comprueba si es base nitrogenada.
Nombre:	add(char l)
Descripción:	Añade un nucleótido.
Nombre:	contar_caracter()
Descripción:	Cuenta los nucleótidos.
Nombre:	una_letra_pos()
Descripción:	Identifica si hay nucleótido ahí.
Nombre:	site_targets()
Descripción:	Busca los sitios blancos.
Nombre:	longitudCadena()
Descripción:	Devuelve la longitud de la secuencia.
Nombre:	devol_caracter(int pos)
Descripción:	Devuelve el nucleótido.
Nombre:	getPuntos_blanco()
Descripción:	Devuelve un punto.
Nombre:	setPuntos_blanco(Lista_puntos punto_blanco)
Descripción:	Modifica un punto.
Nombre:	devolver_punto_pos()
Descripción:	Devuelve la posición de un punto.

Nombre:	subcadena(int p1,int p2)
Descripción:	Devuelve una subcadena.
Nombre:	isR1()
Descripción:	Comprueba si es regla1
Nombre:	setR1(String re1)
Descripción:	Modifica la regla1
Nombre:	isR2()
Descripción:	Comprueba si es regla2
Nombre:	setR2(String re2)
Descripción:	Modifica la regla2
Nombre:	isR3()
Descripción:	Comprueba si es regla3
Nombre:	setR3(String re3)
Descripción:	Modifica la regla3
Nombre:	isR4()
Descripción:	Comprueba si es regla4
Nombre:	setR4(String re4)
Descripción:	Modifica la regla 4
Nombre:	isR5()
Descripción:	Comprueba si es regla5
Nombre:	setR5(String re5)
Descripción:	Modifica la regla5
Nombre:	isR6()
Descripción:	Comprueba si es regla6
Nombre:	setR6(String re6)
Descripción:	Modifica la regla6
Nombre:	isR7()
Descripción:	Comprueba si es regla7
Nombre:	setR7(String re6)
Descripción:	Modifica la regla7
Nombre:	isR8()
Descripción:	Comprueba si es regla8
Nombre:	setR8(String re8)
Descripción:	Modifica la regla 8
Nombre:	cantidadregla()
Descripción:	Devuelve la cantidad de reglas.

Nombre: Gestionar Secuencia_mRNA.	
Tipo de clase: Controladora.	
Atributo	Tipo
driver	String
url	String

user	String
contraseña	String
connection	Connection
Para cada responsabilidad:	
Nombre:	init()
Descripción:	Permite cargar el driver en memoria y crear la conexión.
Nombre:	destroy()
Descripción:	En el método destroy() una vez que el servlet se esta apagando, se desconecta de la base de datos
Nombre:	doGet(HttpServletRequest request, HttpServletResponse response)
Descripción:	Aquí se implementa el método <i>doGet</i> que se encargará de conectarse a la base de datos extraer los datos y mostrar el informe.
Nombre:	doPost(HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i> .
Nombre:	SacarPuntos(ResultSet resultado, HttpServletResponse response)
Descripción:	Buscar los puntos para formar la secuencia.
Nombre:	SacarSubCadena (int pos_i, int pos_f, String chromosome)
Descripción:	Permite sacar una subcadena.
Nombre:	FormarLaCadena(HttpServletResponse response, HttpServletRequest request)
Descripción:	Permite formar la cadena.
Nombre:	CrearPaginaHtml1(HttpServletResponse response)
Descripción:	Crea una página con la secuencia para analizarla.
Nombre:	CrearPaginaHtmlgenes(HttpServletResponse response, ResultSet resultado)
Descripción:	Crea una página con alguna información de un gen.
Nombre:	getTranscript()
Descripción:	Devuelve un transcrito.
Nombre:	setTranscript(Transcript transcript)
Descripción:	Modifica un transcrito.
Nombre:	VirarCadena(String cadenaformada)
Descripción:	Permite vira la secuencia para que cambien la orientación.
Nombre:	CambiarNucleotido (char Nucleotido)
Descripción:	Permite cambiar un nucleótido.

Nombre: GestionarInformaciónGen.	
Tipo de clase: Controladora.	
Atributo	Tipo
driver	String
url	String
user	String
contraseña	String
connection	Connection
Para cada responsabilidad:	

Nombre:	init()
Descripción:	Cargar el driver en memoria y crear la conexión.
Nombre:	destroy()
Descripción:	En el método destroy() una vez que el servlet se esta apagando, se desconecta de la base de datos
Nombre:	doGet(HttpServletRequest request, HttpServletResponse response)
Descripción:	Aquí se implementa el método <i>doGet</i> que se encargará de conectarse a la base de datos extraer los datos y mostrar el informe
Nombre:	doPost(HttpServletRequest request, HttpServletResponse response)
Descripción:	El método <i>doPost</i> , simplemente hace una rellamada a <i>doGet</i>
Nombre:	CrearPaginaHtma1(HttpServletResponse response, ResultSet resultado)
Descripción:	Crea una página con toda la información de un gen

Clases Interfaces:

Nombre: Autenticar.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Nombre	string
Contraseña	string
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CrearNuevaCuenta.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Nombre_usuario	string
Contraseña	string
E_mail	string
País_residencia	string
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: IntroducirCriterioBúsqueda.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Id_gen	Integer
Símbolo_gen	string

Nombre _ transcrito	string
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: GenerarSecuencia.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Nombre _ transcrito	string
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: SeleccionarReglasCumplimiento.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Cantidad _ reglas	Integer
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: SeleccionarReglas.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Cantidad _ reglas	Integer
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: BuscarExon.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Nombre _ transcrito	String
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: BuscarTranscrito.	
----------------------------------	--

Tipo de clase: Interfaz.	
Atributo	Tipo
Id_gen	Integer
Símbolo_gen	String
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: IntroducirSecuencia_mRNA.	
Tipo de clase: Interfaz.	
Atributo	Tipo
Secuencia_mRNA	String
Para cada responsabilidad:	
Nombre:	
Descripción:	

Clases Entidades:

Nombre: TUsuario.	
Tipo de clase: Entidad.	
Atributo	Tipo
Id_Usuario	Integer
Nombre_Usuario	String
Contraseña	String
Rol	String
Para cada responsabilidad:	
Nombre:	Get id_Usuario()
Descripción:	Devuelve el id del usuario.
Nombre:	Set id_Usuario(id_usuario)
Descripción:	Modifica el id del usuario.
Nombre:	Get Nombre_Usuario()
Descripción:	Devuelve el nombre del usuario.
Nombre:	Set Nombre_Usuario(nombre_usuario)
Descripción:	Modifica el nombre del usuario.
Nombre:	Get Contraseña()
Descripción:	Devuelve la contraseña del usuario
Nombre:	Set Contraseña(contraseña)
Descripción:	Modifica la contraseña del usuario.
Nombre:	Get Rol()
Descripción:	Devuelve el rol del usuario.
Nombre:	Set Rol()
Descripción:	Modifica el rol del usuario.

Nombre: Gene.	
Tipo de clase: Entidad.	
Atributo	Tipo
gene_id	int
name	varchar
chromosome	varchar
start_pos	int
end_pos	int
strand	char
band	varchar
type_of_gene	varchar
description	text
taxonomy_id	int
source_inf_id	int
ncbi_geneid	varchar
version	varchar
Para cada responsabilidad:	
Nombre:	Get gene_id()
Descripción:	Devuelve el identificador del gen especificado.
Nombre:	Get name()
Descripción:	Devuelve el nombre del gen de acuerdo al NCBI.
Nombre:	Get chromosome()
Descripción:	Devuelve el número del cromosoma al que pertenece un gen.
Nombre:	Get start_pos()
Descripción:	Devuelve la posición inicial de un gen en el cromosoma.
Nombre:	Get end_pos()
Descripción:	Devuelve la posición final de un gen en el cromosoma.
Nombre:	Get strand()
Descripción:	Devuelve la orientación del gen en el cromosoma, cadena (+) o cadena(-)
Nombre:	Get type_of_gene()
Descripción:	Devuelve el tipo de gen.
Nombre:	Get description()
Descripción:	Devuelve el símbolo o descripción de la función del gen.
Nombre:	Get taxonomy_id()
Descripción:	Devuelve Id del organismo a que pertenece el gen.
Nombre:	Get source_inf_id()
Descripción:	Devuelve el identificador de la fuente de información.
Nombre:	Get ncbi_geneid()
Descripción:	Devuelve el identificador del gen de acuerdo al NCBI.
Nombre:	Get version()

Descripción:	Devuelve la versión del gen.
---------------------	------------------------------

Nombre: Transcript.	
Tipo de clase: Entidad.	
Atributo	Tipo
id_transcript	varchar
gene_id	int
name	varchar
start_pos	int
end_pos	int
description	text
ncbi_geneid	varchar
version	int
transcript	varchar
chromosome	varchar
translation	varchar
translation_version	varchar
trans	varchar
Para cada responsabilidad:	
Nombre:	Get id_transcript()
Descripción:	Devuelve el iidentificador autogenerado del transcrito.
Nombre:	Get gene_id()
Descripción:	Devuelve el id del gen al que pertenece el transcript.
Nombre:	Get name()
Descripción:	Devuelve el nombre del transcrito de acuerdo al NCBI.
Nombre:	Get start_pos()
Descripción:	Devuelve la posición inicial de un transcrito en el cromosoma.
Nombre:	Get end_pos()
Descripción:	Devuelve la posición final de un transcrito en el cromosoma.
Nombre:	Get description()
Descripción:	Devuelve la descripción de un transcrito.
Nombre:	Get ncbi_geneid()
Descripción:	Devuelve el identificador del gen que produce el transcrito.
Nombre:	Get version()
Descripción:	Devuelve la versión del transcrito.
Nombre:	Get transcript()
Descripción:	Devuelve identificador del transcrito de acuerdo al NCBI.
Nombre:	Get chromosome()
Descripción:	Devuelve el cromosoma del gen que produce el transcrito.
Nombre:	Get translation()
Descripción:	Devuelve el identificador de la secuencia aminoacídica de acuerdo al NCBI.
Nombre:	Get translation_version()

Descripción:	Devuelve la versión de la secuencia aminoacídica.
---------------------	---

Nombre: Exon.	
Tipo de clase: Entidad.	
Atributo	Tipo
id_exon	varchar
gene_id	int
transcript_id	int
start_pos	varchar
end_pos	varchar
band	varchar
feature_type	varchar
ncbi_geneid	varchar
transcript	varchar
Para cada responsabilidad:	
Nombre:	Get id_exon()
Descripción:	Devuelve el identificador autogenerado del exon.
Nombre:	Get gene_id()
Descripción:	Devuelve el id del gen donde se encuentra el exon.
Nombre:	Get transcript_id()
Descripción:	Devuelve el id del transcript donde se encuentra el exon.
Nombre:	Get start_pos()
Descripción:	Devuelve la posición inicial de un exon en el cromosoma.
Nombre:	Get end_pos()
Descripción:	Devuelve la posición inicial de un exon en el cromosoma.
Nombre:	Get feature_type()
Descripción:	Devuelve tipo de exon (UTR o CDS).
Nombre:	Get ncbi_geneid()
Descripción:	Devuelve el identificador del gen a que pertenece.
Nombre:	Get transcript()
Descripción:	Devuelve el identificador del transcrito al que pertenece el exon.

3.6 Diagramas de Interacción.

Los diagramas de interacción describen secuencias de intercambios de mensajes entre los roles que implementan el comportamiento de un sistema. Estos diagramas proporcionan una vista integral del comportamiento del sistema, es decir, muestra el flujo de control a través de muchos objetos. Los diagramas de interacción se exhibe en dos diagramas centrados en distintos aspectos pero complementarios: centrados en los objetos individuales y centrados en objetos cooperantes.

Los diagramas de secuencia para el diseño del sistema propuesto muestran la secuencia de mensajes entre objetos durante un escenario concreto (anexo 1). Los diagramas de colaboración son útiles en la fase exploratoria para identificar objetos. La distribución de los objetos en el diagrama permite observar adecuadamente la interacción de un objeto con respecto de los demás.

3.7 Diseño de la base de datos.

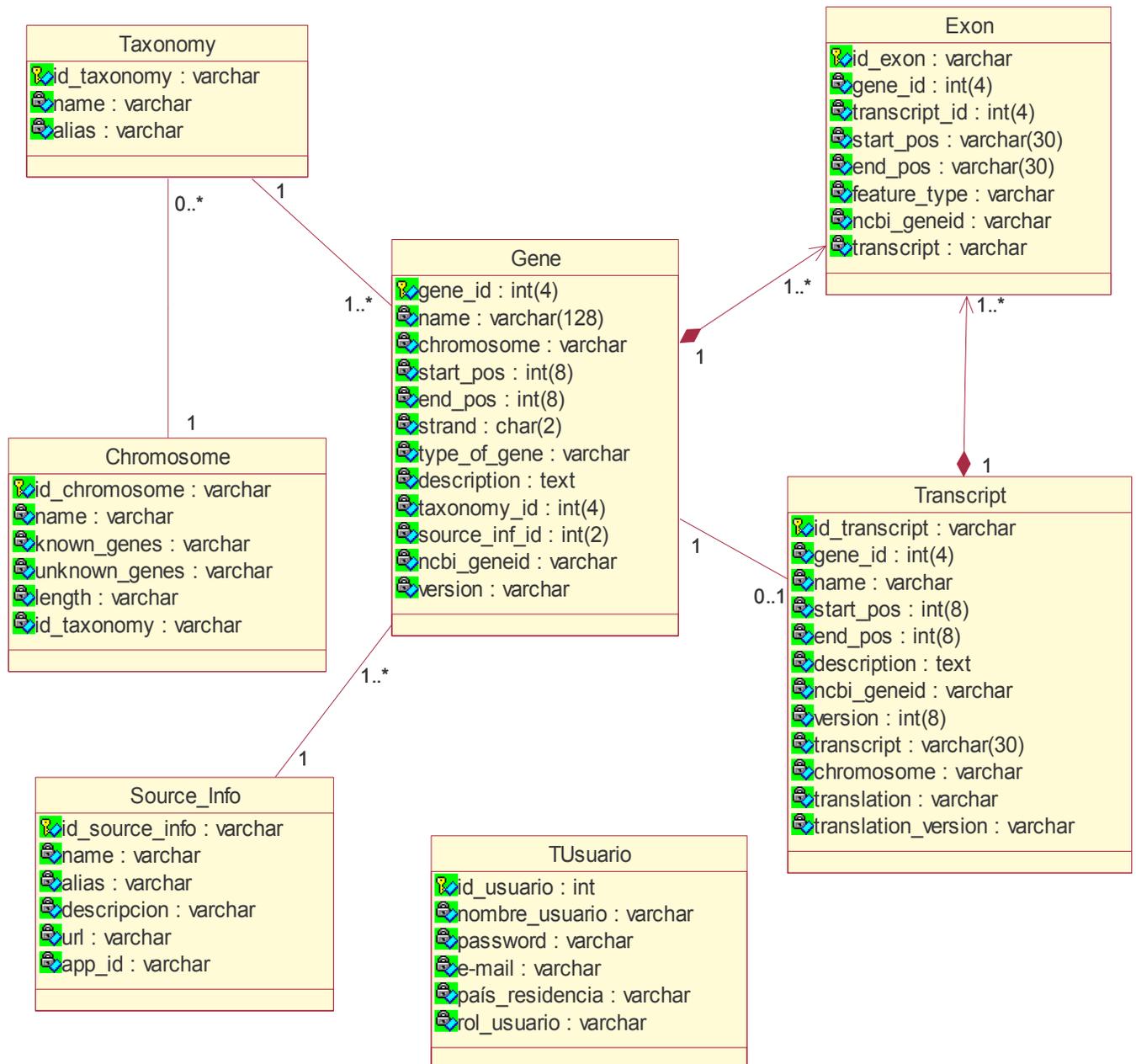
Para el diseño de la base de datos se realizó el diagrama de clases persistentes y el modelo de datos.

Una clase persistente (persistent) es una clase entidad que tiene la capacidad de mantener su valor en el espacio y en el tiempo. Lo contrario son las clases temporales (transient) que son manejadas y almacenadas por el sistema en tiempo de ejecución por lo que dejan de existir cuando termina el programa.

En el diagrama de clases persistentes se muestran dichas clases y las relaciones entre ellas (asociación, agregación/composición y generalización/especialización).

En el diagrama del Modelo físico de datos, se muestran las tablas y sus relaciones.

3.7.1 Modelo lógico de datos.



3.7.2 Modelo físico de datos.



3.8 Descripción de las Tablas de la Base de Datos.

Nombre: TUsuario		
Descripción: En esta tabla se almacenan todos los datos de los usuarios.		
Atributo	Tipo	Descripción
id_usuario	int	Identificador del usuario.
nombre_usuario	varchar	Nombre del usuario.
contraseña	varchar	Es la contraseña del usuario.
e-mail	varchar	El correo electrónico del usuario.
país_residencia	varchar	País de residencia del usuario.
rol_usuario	Varchar	Rol de los usuarios.

Nombre: Gene		
Descripción: En esta tabla se almacenan las informaciones referentes a los genes.		
Atributo	Tipo	Descripción
gene_id	int	Identificador autogenerado del gen.
name	varchar	Nombre del gen de de acuerdo al NCBI
chromosome	varchar	Número del cromosoma donde se localiza el gen
start_pos	int	Posición inicial en el cromosoma.
end_pos	int	Posición final en el cromosoma.
strand	char	Orientación del gen en el cromosoma, cadena + o cadena -
type_of_gene	varchar	Tipo del gen (codificador, rna, etc.)
description	text	Descripción de la función del gen.
taxonomy_id	int	Id del organismo a que pertenece el gen de acuerdo a la base Taxonomy del NCBI
source_inf_id	int	Identificador de la fuente de información
ncbi_geneid	varchar	Identificador del gen de acuerdo al NCBI
version	varchar	Versión del gen.

Nombre: Transcript		
Descripción: En esta tabla se almacenan las informaciones de los transcritos.		
Atributo	Tipo	Descripción
id_transcript	varchar	Identificador autogenerado del transcrito
gene_id	int	Gene _id del gene en la tabla Gene
name	varchar	Nombre del transcrito de de acuerdo al NCBI
start_pos	int	Posición inicial en el cromosoma
end_pos	int	Posición final en el cromosoma

description	text	Descripción del transcrito
ncbi_geneid	varchar	Identificador del gen que produce el transcrito
version	int	Versión del transcrito
transcript	varchar	Identificador del transcrito de acuerdo al NCBI
chromosome	varchar	Cromosoma del gen que produce el transcrito
translation	varchar	Identificador de la secuencia aminoacídica de acuerdo al NCBI.
translation_version	varchar	Versión de la secuencia aminoacídica.

Nombre: Exon		
Descripción: En esta tabla se almacenan las informaciones de los exones.		
Atributo	Tipo	Descripción
id_exon	varchar	Identificador autogenerado del exon
gene_id	int	Gene_id del gene en la tabla Gene
transcript_id	int	transcript_id del transcrito gene en la tabla Transcript
start_pos	varchar	Posición inicial en el cromosoma
end_pos	varchar	Posición final en el cromosoma
feature_type	varchar	Tipo de exon (UTR o CDS)
ncbi_geneid	varchar	Identificador del gen a que pertenece
transcript	varchar	Identificador del transcrito a que pertenece

Nombre: Chromosome		
Descripción: En esta tabla se almacenan las informaciones referentes a los cromosomas.		
Atributo	Tipo	Descripción
id_chromosome	varchar	Identificador autogenerado del cromosoma
name	varchar	Nombre del cromosoma (1,2,3 ... X, Y)
known_genes	varchar	Genes conocidos.
unknown_genes	varchar	Genes desconocidos.
length	varchar	Longitud del cromosoma
id_taxonomy	varchar	Identificador del organismo.

Nombre: Source_Info		
Descripción: En esta tabla se almacenan toda la información de la fuente.		
Atributo	Tipo	Descripción
id_source_info	varchar	Identificador autogenerado de la fuente de información.
name	varchar	Nombre de la fuente.
alias	varchar	Alias de la fuente.
descripcion	varchar	Descripción de la fuente.
url	varchar	URL de la fuente.

app_id	varchar	Identificador de la aplicación que generó la información
--------	---------	--

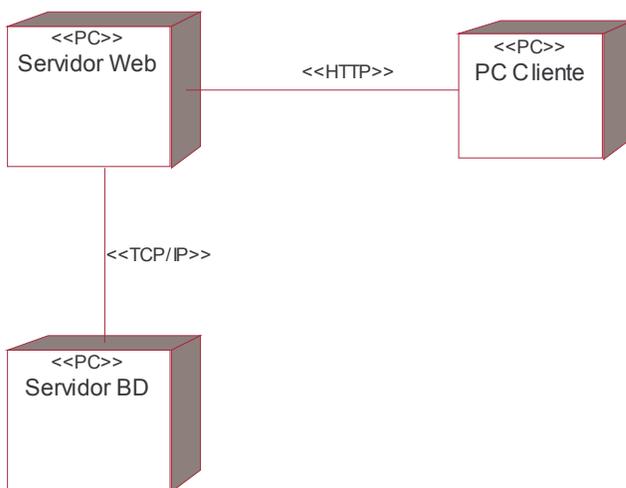
Nombre: Taxonomy		
Descripción: En esta tabla se almacenan todo lo referente a la información de los organismos.		
Atributo	Tipo	Descripción
id_taxonomy	varchar	Identificador del organismo.
name	varchar	Nombre del organismo.
alias	varchar	Alias del organismo.

3.9 Diagrama de Despliegue.

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo.

Se puede observar lo siguiente sobre el modelo de despliegue:

- Cada nodo representa un recurso de cómputo, normalmente un procesador o dispositivo.
- Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como HTTP, TCP/IP, etc.
- Puede describir diferentes configuraciones de red, incluidas las configuraciones para pruebas y para simulación.



3.10 Tratamiento de errores

Con el objetivo de prevenir los errores por parte del usuario, se verifica siempre la integridad de los datos antes de ser utilizados para realizar búsquedas o cualquier otra operación como es el caso de los análisis de secuencias mRNA, todas las prevenciones se hacen con el objetivo de evitar acciones innecesarias y además evitar análisis con problemas en los resultados.

El sistema de “Análisis de secuencias mRNA para el diseño de siRNA” utiliza mensajes para rectificar datos introducidos antes de llevar a cabo las acciones solicitadas por los clientes, donde es de vital importancia la integridad de los mismos para obtener un resultado satisfactorio. Un ejemplo de las validaciones que realiza el sistema es cuando se introduce incorrectamente una secuencia de mRNA que se desea analizar. (Fig. 1).



Figura 1.Utilización de mensajes para rectificar datos introducidos.

3.11 Definiciones de diseño que se aplica.

Se trabajó con formularios web organizados mediante pestañas (Tabpane), para lograr una mayor visualización de la información, permitiendo una mejor navegabilidad. Además se utilizó un mismo color en todas las páginas, así como el uso de un tipo y tamaño de letra adecuado con textos claros, el color predominante es el verde.

La realización de búsquedas de información y el envío de los datos debe ser lo más rápido posible por lo que no se utilizaron muchas imágenes ni funciones muy complejas que atenten contra esto.

3.12 Seguridad.

La seguridad de la aplicación informática realizada está implementada a través del servidor de base de datos PostGreSQL, es decir, se basa en la confiabilidad e integridad de la información que se almacena

en la base de datos y en los resultados que se obtienen de los análisis de secuencias de mRNA para el diseño de siRNA.

3.13 Interfaz.

Se desarrolló una aplicación informática con una interfaz amigable, sencilla y entendible para los usuarios que hagan uso de ella, permitiendo además un trabajo más ameno en la misma.

3.14 Concepción de la ayuda.

La aplicación cuenta con un sistema de ayuda, brindada en cada página a la que accede el usuario en forma de comentario, permitiéndoles tener una mejor y mayor orientación de las diferentes funcionalidades con que cuenta la aplicación en el momento en que acceden a las mismas.

3.15 Conclusiones.

En éste capítulo se determina el diseño apropiado para el software, brindando una clara visión de los principios de diseño utilizados para la construcción del sistema propuesto, así como la definición de diseño aplicada. El resultado principal fue el modelo de diseño que sirve como esquema para la implementación. Además se realizaron los diagramas de interacción correspondiente para cada escenario de cada caso de uso identificado. Se realizó también el modelo de datos generado a partir del diagrama de clases persistentes y se describieron cada una de las tablas que conforman la base de datos y se obtuvo el modelo de despliegue, el cual describe como resultado todas las configuraciones sobre las cuales debería implementarse el sistema.

Se demostró la importancia de ésta etapa y se ganó claridad en cuanto a la concesión del sistema a construir y se sentaron las bases para la fase de implementación y prueba del sistema.

Capítulo 4

Implementación y prueba.

Introducción.

En el presente capítulo se realizará el modelo de implementación a partir de los resultados del flujo de trabajo anteriormente descrito (Diseño),

El modelo de implementación incluye los siguientes elementos:

- Subsistemas de implementación y sus dependencias, interfaces y contenidos.
- Componentes, incluyendo componentes fichero y ejecutables, y las dependencias entre ellos.
- La vista de arquitectura del modelo de implementación, incluyendo sus elementos arquitectónicamente significativos.

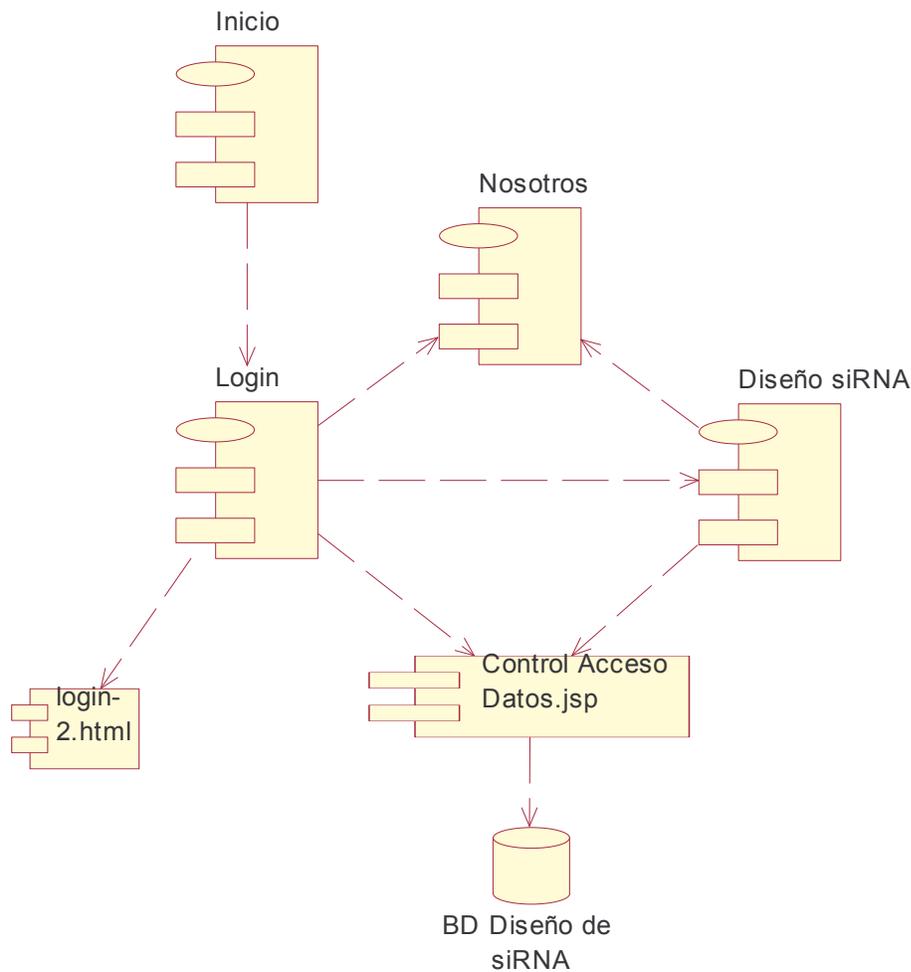
Además se van a detallar las pruebas realizadas al sistema, teniéndose en cuenta principalmente los casos de prueba de caja negra, que las mismas se realizan sobre la interfaz del software.

4.1 Diagrama de Componentes.

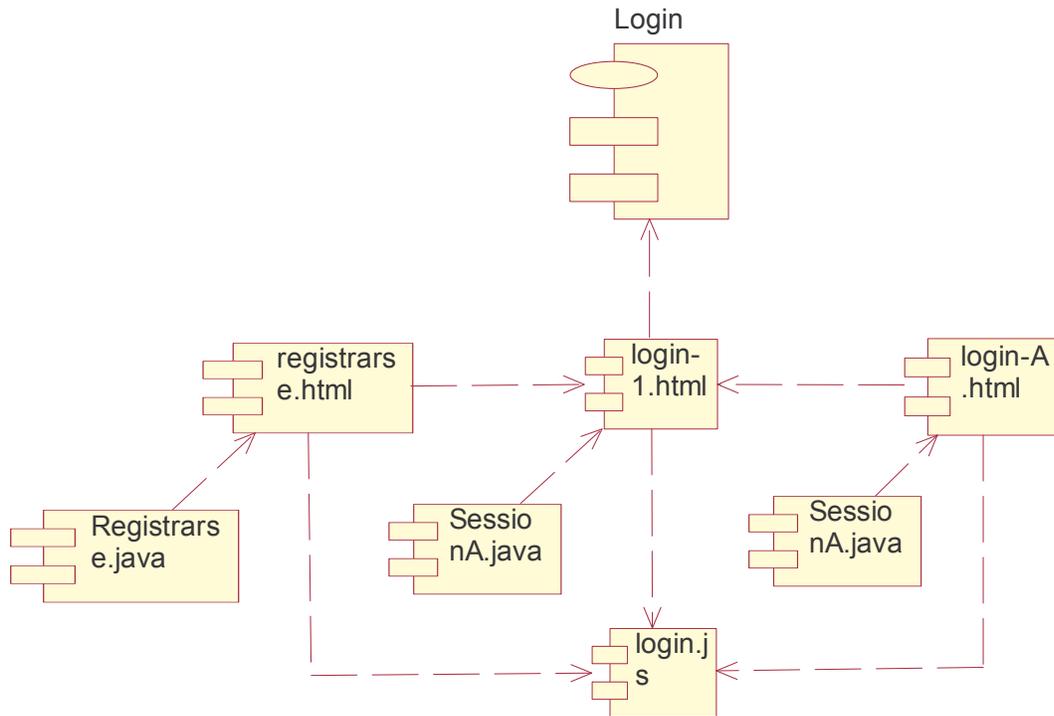
El diagrama de componentes describe los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Se presenta un diagrama de componentes general y un diagrama de componentes para cada módulo especificado, con el objetivo de lograr una mejor y mayor comprensión. En el componente módulo de “Inicio”, se detalla información referente al tema del diseño de siRNA y se dan a conocer conceptos importantes para el buen uso de términos relacionados con el tema que se desarrolló en la aplicación, como es el caso del significado de silenciamiento de genes; el módulo de “Login” se especifica lo referentes a los usuarios, es decir, registro y autenticación de los mismos; el módulo “Diseño siRNA” que se refiere a todas las operaciones que se le realizan para obtener el diseño de siRNA a partir de un análisis de las secuencia mRNA, así como el manejo de informaciones relevantes para llegar a conocer y generar las secuencias mRNA; y el módulo de

“Nosotros” detalla información de los desarrolladores y de la misión objetiva del “Sistema de análisis de secuencias mRNA para el diseño de siRNA”. Todos los módulos anteriormente mencionados y descritos se encuentran localizados según el diagrama de despliegue en el nodo servidor Web y la base de datos diseño de siRNA se encuentra en el servidor de BD.

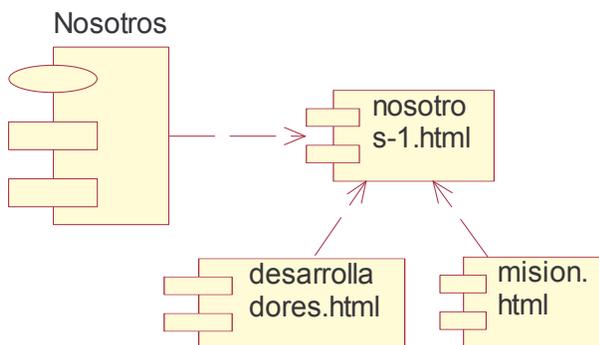
4.1.1 Diagrama de Componentes General.



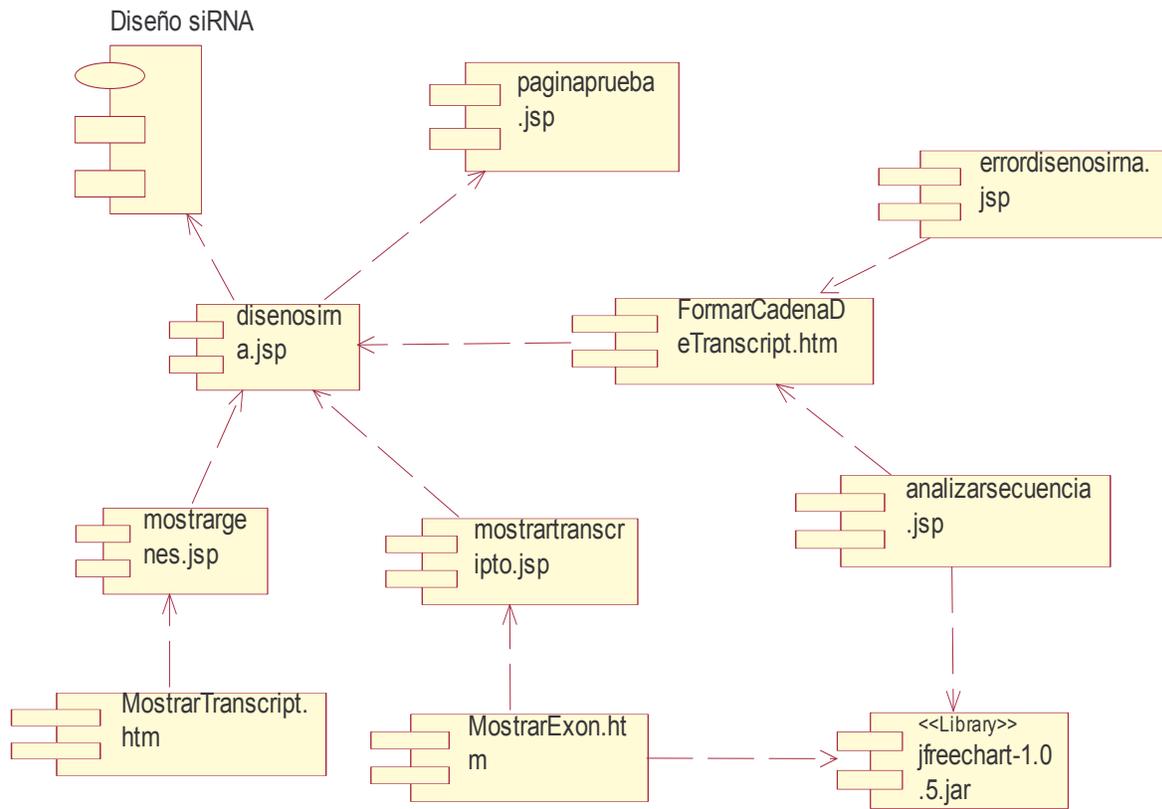
4.1.2 Diagrama de Componentes para el módulo "LOGIN"



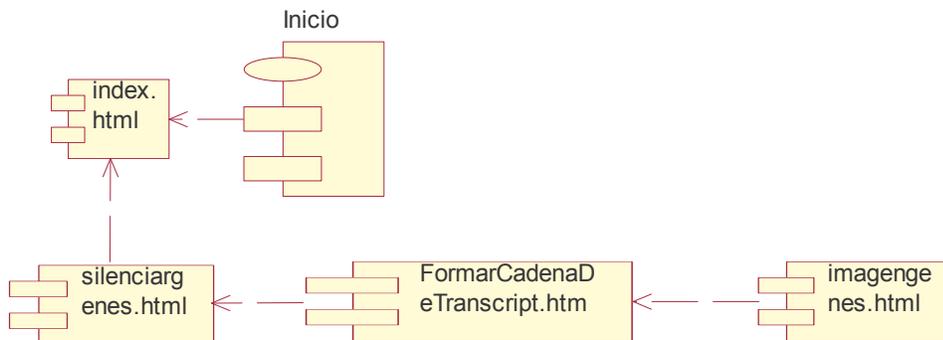
4.1.3 Diagrama de Componentes para el módulo "NOSOTROS"



4.1.4 Diagrama de Componentes para el módulo “DISEÑO siRNA”



4.1.5 Diagrama de Componentes para el módulo “INICIO”



4.2 Modelo de Prueba.

La etapa de prueba es tan o más importante que todas las realizadas hasta el momento puesto que en ella se refleja la calidad con que ha sido llevada a cabo la proyección del sistema. El objetivo de la prueba de software es descubrir errores. En todas las fases del desarrollo del proyecto hay que probar el software que se va construyendo, aunque como el grueso de la programación se realiza en la construcción, es en esa fase en la que se centran los mayores esfuerzos de éste flujo.

4.2.1 Casos de prueba de caja negra.

Las pruebas de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema sin tener mucho en cuenta la estructura interna del software.

Las pruebas de caja negra se realizan principalmente en el nivel de sistema, desarrollándose generalmente el método de particiones de equivalencia que es una técnica de estas pruebas de caja negra que sirve para comprobar si las entradas y salidas desde la interfaz son correctas.

Caso de Prueba “CU Autenticar Usuario”.

El CU “Autenticar Usuario” acepta como parámetro un usuario y una contraseña. En éste caso para determinar las clases de equivalencia se tiene:

- Usuario que existe y contraseña incorrecta
- Usuario y contraseña que no existen en la BD
- Usuario que no existe y contraseña que si existe
- Usuario y contraseña correctos.

Es necesario realizar un caso de prueba para cada una de las 4 clases de equivalencia identificadas.

Iniciar sesión

Usuario:

Contraseña:

Caso de uso: Autenticar Usuario.

Caso de prueba: Comprobar que no deja entrar a un usuario existente con una contraseña equivocada

Descripción:

Se intentará acceder al sistema con un usuario existente y una contraseña incorrecta, una vez entrado los datos el sistema dirige al usuario a una página de error para intentar nuevamente la autenticación.

Entrada: usuario “admin” y contraseña “apto66106”

Condiciones de ejecución: No existe en la tabla TUsuario(nombre, contraseña, email, id_usuario) la tupla <“admin”, “apto66106”,x> pero sí una tupla <“admin”,“admin”,x>

Resultado esperado: No deja entrar y cambia la tupla a <“admin”,“admin”,x+1>

Tabla 1: CP1: Usuario que existe y contraseña incorrecta. “CU Autenticar Usuario”

Caso de uso: Autenticar Usuario.

Caso de prueba: Comprobar que no deja entrar a un usuario y contraseña que no existen en la base

de datos.
Descripción: Se intentará acceder al sistema con un usuario y una contraseña que no existen en la base de datos, una vez entrado los datos el sistema dirige al usuario a una página de error para intentar nuevamente la autenticación.
Entrada: ” usuario “ arianna ” y contraseña “ haro ”
Condiciones de ejecución: No existe en la tabla TUsuario(nombre, contraseña, email, id_usuario) la tupla <“ arianna ”, “ haro ”,x>
Resultado esperado: No deja entrar y cambia la tupla a <“ nuevo_usuario ”,“ nueva_contraseña ”,x+1>

Tabla 2: CP2: Usuario y contraseña que no existen en la BD. “CU Autenticar Usuario”

Caso de uso: Autenticar Usuario.
Caso de prueba: Comprobar que no deja entrar a un usuario que no existe y una contraseña que si existe.
Descripción: Se intentará acceder al sistema con un usuario que no existe y una contraseña correcta, una vez entrado los datos el sistema dirige al usuario a una página de error para intentar nuevamente la autenticación.
Entrada: ” usuario “ sinay96 ” y contraseña “ yani ”
Condiciones de ejecución: No existe en la tabla TUsuario(nombre, contraseña, email, id_usuario) la tupla <“ sinay96 ”, “ yani ”,x> pero sí una tupla <“ yani ”, “ yani ”,x>
Resultado esperado: No deja entrar y cambia la tupla a <“ yani ”, “ yani ”,x+1>

Tabla 3: CP3: Usuario que no existe y contraseña que si existe. “CU Autenticar Usuario”

Caso de uso: Autenticar Usuario.
Caso de prueba: Comprobar que deja entrar a un usuario y contraseña que existen en la base de datos.
Descripción: Se intentará acceder al sistema con un usuario y contraseña existente, una vez entrado los datos el sistema permite el acceso a la aplicación.
Entrada: ” usuario “yani” y contraseña “yani”
Condiciones de ejecución: Existe en la tabla TUsuario(nombre, contraseña, email, id _ usuario) la tupla <“yani”, “yani”,x>
Resultado esperado: Deja entrar al usuario, quedando éste satisfactoriamente autenticado y con los permisos para acceder a la aplicación.

Tabla 4: CP4: Usuario y contraseña correctos. “CU Autenticar Usuario”

Caso de Prueba “CU Registrar Usuario”.

El CU “Registrar Usuario” acepta como parámetro un usuario, contraseña, email y país de residencia. En éste caso para determinar las clases de equivalencia se tiene:

- Usuario y el email ya existen.
- Usuario nuevo, email existente.
- Usuario existente, email nuevo.
- Usuario y el email son nuevos.

Es necesario realizar un caso de prueba para cada una de las 4 clases de equivalencia identificadas.

Iniciar sesión

Usuario:	<input type="text"/>
Contraseña:	<input type="password"/>
Email:	<input type="text"/>
País:	<input type="text" value="Cuba"/>
<input type="button" value="Registrarse"/>	

Caso de uso: Registrar Usuario.

Caso de prueba: Comprobar que deja registrar a un usuario y contraseña que ya existen en la base de datos.

Descripción: Se intentará crear una nueva cuenta y quedar registrado en el sistema, con un usuario y un email existente, una vez entrado los datos el sistema muestra un mensaje para rectificar los mismos y dirige al usuario a una página de error para intentar nuevamente el registro.

Entrada: usuario “yani” , email “yanisleyky@gmail.com”

Condiciones de ejecución: Existe en la tabla TUsuario(nombre, contraseña, email, id_ usuario) la tupla <“yani”, “yanisleyky@gmail.com”,x>

Resultado esperado: No deja registrar al nuevo usuario y cambia la tupla a <“nuevo _ usuario”,“nueva_ contraseña”,x+1>

Tabla 1: CP1: Usuario y el email ya existen. “CU Registrar Usuario”

Caso de uso: Registrar Usuario.

Caso de prueba: Comprobar que deja registrar a un usuario nuevo, pero con un email que ya existe en la base de datos.

Descripción: Se intentará crear una nueva cuenta y quedar registrado en el sistema, con un usuario nuevo, pero con un email existente, una vez entrado los datos el sistema muestra un mensaje para rectificar los mismos y dirige al usuario a una página de error para intentar nuevamente el registro.

Entrada: usuario “Cuca” , email “yanisleyky@gmail.com”

Condiciones de ejecución: Existe en la tabla TUsuario(nombre, contraseña, email, id_ usuario) la tupla <“yani”, “yanisleyky@gmail.com”,x>

Resultado esperado: No deja registrar al nuevo usuario y cambia la tupla a <“Cuca”, “nuevo_email”,x+1>

Tabla 2: CP2: Usuario nuevo, email existente. “CU Registrar Usuario”

Caso de uso: Registrar Usuario.

Caso de prueba: Comprobar que deja registrar a un usuario ya existente en la base de datos pero con email nuevo.

Descripción: Se intentará crear una nueva cuenta y quedar registrado en el sistema, con un usuario que ya existe y un email nuevo, una vez entrado los datos el sistema muestra un mensaje de registro satisfactorio.

Entrada: usuario “yani” , email “cubana100@cuba.cu”

Condiciones de ejecución: Existe en la tabla TUsuario(nombre, contraseña, email, id_ usuario) la tupla <“yani”, “yanisleyky@gmail.com”,x>

Resultado esperado: Deja registrar al nuevo usuario <“yani”, “cubana100@cuba.cu”, x+1>

Existiendo en la base de datos dos usuarios con el mismo nombre de registro.

Tabla 3: CP3: Usuario existente, email nuevo. “CU Registrar Usuario”

Caso de uso: Registrar Usuario.

Caso de prueba: Comprobar que deja registrar a un usuario nuevo con email nuevo.

Descripción: Se intentará crear una nueva cuenta y quedar registrado en el sistema, con un usuario y un email nuevo, una vez entrado los datos el sistema muestra un mensaje de bienvenida y se realiza un registro satisfactorio.

Entrada:” usuario “yoendris” , email “ydiazga@estudiantes.uci.cu”

Condiciones de ejecución: No existe en la tabla TUsuario(nombre, contraseña, email, id_ usuario) la tupla <“yoendris”, “ydiazga@estudiantes.uci.cu”>

Resultado esperado: Queda registrado el nuevo usuario.

Tabla 4: CP4: Usuario y el email son nuevos. “CU Registrar Usuario”

Caso de Prueba “CU Formar secuencia de mRNA”.

El CU “Formar secuencia de mRNA” acepta como parámetro un criterio de búsqueda id _ gen, símbolo del gen y nombre del transcrito para buscar información y luego formar la secuencia. En éste caso para determinar las clases de equivalencia se tiene:

- Id _ gene no existe.
- Id _ gene existe.
- Símbolo del gen, no existe.
- Símbolo del gen, existe.

- Nombre del transcrito, no existe.
- Nombre del transcrito, existe.

Es necesario realizar un caso de prueba para cada una de las 6 clases de equivalencia identificadas.

Generar secuencia de mRNA:

Nombre del transcrito
 ID del gen
 Símbolo del gen

Caso de uso: Formar secuencia de mRNA.

Caso de prueba: Comprobar que se realiza la búsqueda de información para generar secuencias de mRNA a partir de un id _ gene que no existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los genes para formar una secuencia de mRNA, con un id_ gene no existente en la base de datos, una vez entrado el dato el sistema muestra un mensaje para que el usuario conozca que el id_ gene especificado no existe en la BD y dirige al usuario a una página de error para que introduzca un nuevo id _ gene.

Entrada: id_ gene “060FB”.

Condiciones de ejecución: No existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“060FB”, x>.

Resultado esperado: No se realiza la búsqueda y por tanto no se genera la secuencia de mRNA, y cambia la tupla <“nuevo_id_gene”,x+1>.

Tabla 1: CP1: Id _ gene no existe. “CU Formar secuencia de mRNA “

Caso de uso: Formar secuencia de mRNA.
Caso de prueba: Comprobar que se realiza la búsqueda de información para generar secuencias de mRNA a partir de un id _ gene que existe en la base de datos.
Descripción: Se intentará realizar una búsqueda de información de los genes para formar una secuencia de mRNA, con un id _ gene que existe en la base de datos, una vez entrado el dato el sistema busca la información referente al criterio introducido y dirige al usuario a una página donde se muestran los resultados y brinda la oportunidad de formar la secuencia.
Entrada: ” id _ gene “1000”.
Condiciones de ejecución: Existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“1000”>.
Resultado esperado: Se realiza la búsqueda y se genera la secuencia de mRNA.

Tabla 2: CP2: Id _ gene existe. “CU Formar secuencia de mRNA “

Caso de uso: Formar secuencia de mRNA.
Caso de prueba: Comprobar que se realiza la búsqueda de información para generar secuencias de mRNA a partir de un símbolo del gen que no existe en la base de datos.
Descripción: Se intentará realizar una búsqueda de información de los genes para formar una secuencia de mRNA, con un símbolo de gen no existente en la base de datos, una vez entrada la información el sistema muestra un mensaje para que el usuario conozca que el símbolo del gen especificado no existe en la BD y dirige al usuario a una página de error para que introduzca un nuevo

símbolo del gen.

Entrada: Símbolo del gen “XXX TT 55500 ribosomas”.

Condiciones de ejecución: No existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“XXX TT 55500 ribosomas”, x>.

Resultado esperado: No se realiza la búsqueda y por tanto no se genera la secuencia de mRNA, y cambia la tupla <“nuevo _ símbolo _ gen”,x+1>

Tabla 3: CP3: Símbolo del gen no existe. “CU Formar secuencia de mRNA “

Caso de uso: Formar secuencia de mRNA.

Caso de prueba: Comprobar que se realiza la búsqueda de información para generar secuencias de mRNA a partir de un símbolo de gen que existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los genes para formar una secuencia de mRNA, con un símbolo del gen que existe en la base de datos, una vez entrada la información el sistema busca la información referente al criterio introducido y dirige al usuario a una página donde se muestran los resultados y brinda la oportunidad de formar la secuencia.

Entrada: símbolo del gen “N-acetyltransferase 2 (arylamine N-acetyltransferase)”.

Condiciones de ejecución: Existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“N-acetyltransferase 2 (arylamine N-acetyltransferase)”>.

Resultado esperado: Se realiza la búsqueda y se genera la secuencia de mRNA.

Tabla 4: CP4: Símbolo del gen existe. “CU Formar secuencia de mRNA “

Caso de uso: Formar secuencia de mRNA.

Caso de prueba: Comprobar que se realiza la búsqueda de información para generar secuencias de mRNA a partir de un nombre de transcrito que no existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los genes para formar una secuencia de mRNA, con un nombre de transcrito no existente en la base de datos, una vez entrada la información el sistema muestra un mensaje para que el usuario conozca que el nombre de transcrito especificado no existe en la BD y dirige al usuario a una página de error para que introduzca un nuevo símbolo del gen.

Entrada: "Nombre de transcrito "NM_A0010C".

Condiciones de ejecución: No existe en la tabla Transcript (name, transcript _ id, translation, strand, start_pos, end_pos) la tupla <"NM_A0010C", x>.

Resultado esperado: No se realiza la búsqueda y por tanto no se genera la secuencia de mRNA, y cambia la tupla <"nuevo _ nombre _ transcrito",x+1>

Tabla 5: CP5: Nombre del Transcrito no existe. "CU Formar secuencia de mRNA "

Caso de uso: Formar secuencia de mRNA.

Caso de prueba: Comprobar que se realiza la búsqueda de información para generar secuencias de mRNA a partir de un nombre de transcrito que existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los genes para formar una secuencia de mRNA, con un nombre de transcrito que existe en la base de datos, una vez entrada la información el sistema busca la información referente al criterio introducido y dirige al usuario a una página donde se muestra la secuencia de mRNA generada.

Entrada: símbolo del gen “NM_000035”.

Condiciones de ejecución: Existe en la tabla Transcrito (name, transcript _ id, translation, strand, start_pos, end_pos) la tupla <“NM_000035”>.

Resultado esperado: Se genera la secuencia de mRNA y se muestra.

Tabla 6: CP6: Nombre del Transcrito existe. “CU Formar secuencia de mRNA “

Caso de Prueba “CU Analizar secuencia de mRNA”.

El CU “Analizar secuencia de mRNA” acepta como parámetro una secuencia de mRNA y una especificación de la cantidad de regla que se desean para el análisis para obtener el análisis de la misma. En éste caso para determinar las clases de equivalencia se tiene:

- Secuencia mRNA entrada.
- Secuencia de mRNA generada.

Es necesario realizar un caso de prueba para cada una de las clases de equivalencia identificadas.

Secuencia de nucleótidos (mRNA) generada:

```
AGATCCCTGGGGTATCCATACATGATCTGGTGTGGGGTGGTACG  
GAATCCTGTGTTTTTACGTATTTAAAAATACGTTATACCTATAATTA  
GTCACACGAGGAAATCAAATGCTAAAAGTATGATATGTTTTTATGTTT  
TGTTTTTCTTGCTTAGGGGATCATGGACATTGAAGCATATTTTGAAA  
GAATTGGCTATAAGAACTCTAGGAACAAATTGGACTTGAAGACATTA  
ACTGACATTCTTGAGCACCCAGATCCGGGGCTGTTCCCTTTGAGAACCT  
TAACATGCATTGTGGGCAAGCCATGGAGTTGGGCTTAGAGGCTATTT  
TTGATCACATTGTAAGAAGAAACCGGGGTGGGTGGTGTCTCCAGGTC  
AATCAACTTCTGTACTGGGCTCTGACCACAAATCGGTTTTTCAGACCAC  
AATGTTAGGAGGGTATTTTTTACATCCCTCCAGTTAACAAATACAGCA  
CTGGCATGGTTTACCTTCTCTGCAAGGTGACCATTGACGGCAGGAAT
```

Reglas a tener en cuenta:

- Comienza con AA
- Contiene CG de 30-50%
- 3 o mas A/T entre las bases 15 y 19
- A en la posición 19
- A en la posición 3
- T en la posición 10
- Ausencia de GC en la posición 19
- Ausencia de G en la posición 13

Analizar

Introduzca la secuencia de nucleótidos (mRNA):

Reglas a tener en cuenta:

- Comienza con AA
- Contiene CG de 30-50%
- 3 o mas A/T entre las bases 15 y 19
- A en la posición 19
- A en la posición 3
- T en la posición 10
- Ausencia de GC en la posición 19
- Ausencia de G en la posición 13

Borrar Analizar

Caso de uso: Analizar secuencia de mRNA.

Caso de prueba: Comprobar que se realiza el análisis adecuado de la secuencia mRNA entrada a partir de la cantidad de reglas especificadas por el usuario.

Descripción: Se intentará realizar un análisis de las secuencias de mRNA entrada por un usuario, una vez entrada la misma el sistema valida si la secuencia está formada sólo por nucleótidos, entonces procede a realizar el análisis y dirige al usuario a una página donde se muestran los resultados obtenidos.

Entrada:” Secuencia mRNA “**agaatcgtacatgcataaacggtgcacatgcatagctacaactatagccat**”.

Condiciones de ejecución: Que existan sólo nucleótidos en la secuencia.

Resultado esperado: Se obtiene el análisis de la secuencia de mRNA y se muestran los resultados.

Tabla 1: CP1: Secuencia mRNA entrada. “CU Analizar secuencia de mRNA “

Caso de uso: Analizar secuencia de mRNA.

Caso de prueba: Comprobar que se realiza el análisis adecuado de la secuencia mRNA generada a partir de la cantidad de reglas especificadas por el usuario.

Descripción: Se intentará realizar un análisis de las secuencias de mRNA generada a partir de la especificación de criterios de búsquedas realizados por un usuario, una vez obtenida la misma y seleccionadas las reglas el sistema lleva a cabo el análisis y luego dirige al usuario a una página donde se muestran los resultados obtenidos.

Entrada:” Cantidad de reglas “3”.

Condiciones de ejecución: Que se especifique la cantidad adecuada de reglas con relativa concordancia.

Resultado esperado: Se obtiene el análisis de la secuencia de mRNA y se muestran los resultados.

Tabla 2: CP2: Secuencia mRNA generada. “CU Analizar secuencia de mRNA “

Caso de Prueba “CU Mostrar genes”.

El CU “Mostrar genes” acepta como parámetro un id_ gene y un símbolo _ gen para buscar información necesaria de los genes y luego mostrarla al usuario. En éste caso para determinar las clases de equivalencia se tiene:

- Id_ gene no existe.
- Id_ gene existe.
- Símbolo _ gen no existe

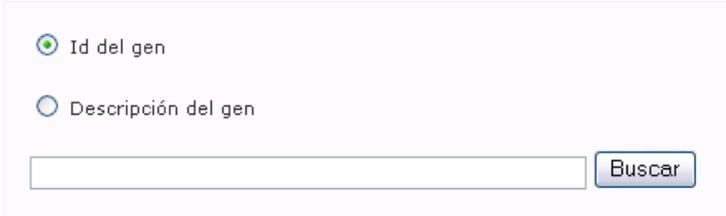
- Símbolo _ gen existe.

Es necesario realizar un caso de prueba para cada una de las 4 clases de equivalencia identificadas.

Mostrar información de un gen

En esta sección usted podrá realizar búsquedas de transcritos asociados a un gen interactuando con nuestra base de datos, obteniendo el resultado que desea.

Buscar transcritos:



Formulario de búsqueda de transcritos. Incluye dos radio buttons: "Id del gen" (seleccionado) y "Descripción del gen". Debajo hay un campo de texto vacío y un botón "Buscar".

Caso de uso: Mostrar genes.

Caso de prueba: Comprobar que se realiza la búsqueda de información de los genes y transcritos asociados al gen para ser mostrada al usuario, a partir de un id _ gene que se especificó y que no existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los genes y transcritos asociados al gen para mostrar al usuarios datos importantes de los mismos, con un id_ gene no existente en la base de datos, una vez entrado el dato el sistema muestra un mensaje para que el usuario conozca que el id_ gene especificado no existe en la BD y dirige al usuario a una página de error para que introduzca un nuevo id _ gene.

Entrada: id_ gene "060FB".

Condiciones de ejecución: No existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <"060FB", x>.

Resultado esperado: No se realiza la búsqueda y por tanto no se muestra la información de los genes ni de transcritos asociados al gen, y cambia la tupla <"nuevo_id_gene",x+1>

Tabla 1: CP1: Id _ gene no existe. “CU Mostrar genes “

Caso de uso: Mostrar genes.
Caso de prueba: Comprobar que se realiza la búsqueda de información de los genes y transcritos asociados al gen para ser mostrada al usuario, a partir de un id _ gene que se especificó y que existe en la base de datos.
Descripción: Se intentará realizar una búsqueda de información de los genes y transcritos asociados al gen para mostrar al usuario datos importantes de los mismos, con la especificación de un id_ gene existente en la base de datos, una vez entrado el dato el sistema busca la información y luego muestra los resultados.
Entrada: ” id_ gene “1000”.
Condiciones de ejecución: Existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“1000”>.
Resultado esperado: Se realiza la búsqueda y por tanto se muestra la información de los genes ni de transcritos asociados al gen.

Tabla 2: CP2: Id _ gene existe. “CU Mostrar genes “

Caso de uso: Mostrar genes.
Caso de prueba: Comprobar que se realiza la búsqueda de información para mostrarla al usuario, a partir de la especificación de un símbolo del gen que no existe en la base de datos.
Descripción: Se intentará realizar una búsqueda de información de los genes y de los transcritos asociados al gen para mostrarla al usuario, con la especificación de un símbolo de gen no existente en la base de datos, una vez entrado los datos el sistema muestra un mensaje para que el usuario

conozca que el símbolo del gen no existe en la BD y dirige al usuario a una página de error para que introduzca un nuevo símbolo del gen.

Entrada: Símbolo del gen “**XXX TT 55500 ribosomas**”.

Condiciones de ejecución: No existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“**XXX TT 55500 ribosomas**”, x>.

Resultado esperado: No se realiza la búsqueda y por tanto no se muestra la información de los genes ni de transcritos asociados al gen, y cambia la tupla <“**nuevo _ símbolo _ gen**”,x+1>

Tabla 3: CP3: Símbolo del gen no existe. “CU Mostrar genes “

Caso de uso: Mostrar genes.

Caso de prueba: Comprobar que se realiza la búsqueda de información para mostrarla al usuario, a partir de la especificación de un símbolo del gen que existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los genes y de los transcritos asociados al gen para mostrarla al usuario, con la especificación de un símbolo de gen existente en la base de datos, una vez entrado los datos el sistema busca la información referente al criterio introducido y dirige al usuario a una página donde se muestran los resultados obtenidos.

Entrada: símbolo del gen “**N-acetyltransferase 2 (arylamine N-acetyltransferase)**”.

Condiciones de ejecución: Existe en la tabla Gene (gene _ id, name, description, chromosome, strand) la tupla <“**N-acetyltransferase 2 (arylamine N-acetyltransferase)**”>.

Resultado esperado: Se realiza la búsqueda y por tanto se muestra la información de los genes ni de transcritos asociados al gen.

Tabla 4: CP4: Símbolo del gen existe. “CU Mostrar genes “

Caso de Prueba “CU Mostrar transcrito”.

El CU “Mostrar transcrito” acepta como parámetro el nombre de un transcrito para buscar información necesaria de los exones asociados al transcrito y del gen donde se encuentra el transcrito, para luego mostrarla al usuario. En éste caso para determinar las clases de equivalencia se tiene:

- Nombre de transcrito no existe.
- Nombre de transcrito existe.

Es necesario realizar un caso de prueba para cada una de las clases de equivalencia identificadas.

Mostrar información de un transcrito

En esta sección usted podrá realizar búsquedas de exones asociados a un transcrito interactuando con nuestra base de datos, obteniendo el resultado que desea.

Buscar exon:

Nombre de transcrito

Caso de uso: Mostrar Transcrito.

Caso de prueba: Comprobar que se realiza la búsqueda de información de los exones asociados al transcrito y del gen donde se encuentra el transcrito para mostrarla al usuario, ésta búsqueda se realiza a partir de un nombre de transcrito que no existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los exones asociados al transcrito y del gen donde se encuentra el transcrito para mostrarla al usuario, con la especificación de un nombre de transcrito no existente en la base de datos, una vez entrada la información el sistema muestra un mensaje para que el usuario conozca que el nombre de transcrito especificado no existe en la BD y dirige al usuario a una página de error para que introduzca un nuevo símbolo del gen.

Entrada: Nombre de transcrito “NM_A0010C”.

Condiciones de ejecución: No existe en la tabla Transcript (name, transcript _ id, translation, strand, start_pos, end_pos) la tupla <“NM_A0010C”, x>.

Resultado esperado: No se realiza la búsqueda y por tanto no se muestra ningún resultado, y cambia la tupla <“nuevo _ nombre _ transcrito”,x+1>

Tabla 5: CP5: Nombre del Transcrito no existe. “CU Mostrar Transcrito”

Caso de uso: Mostrar Transcrito.

Caso de prueba: Comprobar que se realiza la búsqueda de información de los exones asociados al transcrito y del gen donde se encuentra el transcrito para mostrarla al usuario, ésta búsqueda se realiza a partir de un nombre de transcrito que existe en la base de datos.

Descripción: Se intentará realizar una búsqueda de información de los exones asociados al transcrito y del gen donde se encuentra el transcrito para mostrarla al usuario, con la especificación de un símbolo de gen existente en la base de datos, una vez entrado los datos el sistema busca la información referente al criterio introducido y dirige al usuario a una página donde se muestran los resultados obtenidos.

Entrada: símbolo del gen “NM_000035”.

Condiciones de ejecución: Existe en la tabla Transcript (name, transcript _ id, translation, strand, start_pos, end_pos) la tupla <“NM_000035”>.

Resultado esperado: Se realiza la búsqueda y por tanto se muestran los resultados.

Tabla 6: CP6: Nombre del Transcrito existe. “CU Mostrar Transcrito”

4.3 Conclusiones.

Se describieron detalladamente los diagramas de componentes que corresponden al modelo de implementación del sistema y se realizaron las pruebas de caja negra donde se pudo comprobar las funcionalidades de la aplicación. Durante la etapa de prueba se comprobó la existencia de algunos errores que estaban en contra del buen funcionamiento del software, los cuales fueron rectificados a medida que se fueron realizando pruebas posteriores.

Conclusiones.

Con el presente trabajo se propone una solución íntegra, capaz de resolver los problemas que enfrentaban los especialistas en sus trabajos investigativos referente a los diseños de siRNA y el silenciamiento génico, esta solución será brindada por el Sistema de Análisis de Secuencias mRNA para el Diseño de siRNA, para uso del Centro de Ingeniería Genética y Biotecnología en Cuba y para todas las demás instituciones nacionales e internacionales que necesiten hacer uso de la herramienta desarrollada. Se presenta una aplicación capaz de realizar los diseños de siRNA, a partir de un análisis detallado de secuencias mRNA, brindando además resultados donde se muestra suficiente información de los sitios blancos que se utilizarán para los estudios que se llevarán a cabo. El buen uso de esta aplicación, puede convertirla en una poderosa herramienta para el avance científico de los especialistas que realicen investigaciones en el marco de trabajo de los diseños de los ARN pequeños de interferencia.

El sistema se desarrolló siguiendo la metodología RUP, y se utilizaron representaciones para la modelación de todas las fases del proyecto.

El sistema resultante tiene un ambiente fácil de entender y usar, el cual cumple con los estándares de diseño y utiliza técnicas modernas de la programación orientada a objetos.

Por todo lo anterior expuesto se concluye que los objetivos propuestos para el presente trabajo han sido cumplidos satisfactoriamente. Se incluyen además una serie de recomendaciones que deben tenerse en cuenta para el trabajo futuro.

Recomendaciones

A pesar de haberse cumplido los objetivos generales del trabajo, nuevas ideas han ido surgiendo en el desarrollo del mismo, permitiendo esto que en un futuro se cree un sistema más eficiente, por tanto recomendamos:

1. Continuar el desarrollo del sistema con el objetivo de implementar otras funcionalidades que se desean para próximas versiones.
2. Integrar el sistema con otras aplicaciones que requieran obtener la información almacenada por el mismo.
3. Aplicar el sistema a todas las instituciones nacionales e internacionales donde pueda ser usado, permitiendo un buen uso del mismo.

Referencias bibliográficas.

- [1] Booch, G., Rumbaugh, J., Jacobson, I. *“El Lenguaje Unificado de Modelado”*. Addison-Wesley. 1999.
- [2] *Eclipse: una herramienta profesional al alcance de todos*. [Consultado el: 16 de enero de 2007]. Disponible en:
http://www.gui.uva.es/~laertes/nuke/index.php?option=com_content&task=view&id=56&Itemid=41
- [3] *¿Qué es el Tomcat?* [Consultado el: 16 de enero de 2007]. Disponible en:
<http://es.wikipedia.org/wiki/Tomcat>
- [4] Joseph, Schmuller. *“Aprendiendo UML en 24 horas”*, Prentice-Hall, Inc. 2001.
- [5] *¿Qué es PostgreSQL?* [Consultado el: 16 de enero de 2007]. Disponible en:
http://www.netpecos.org/docs/mysql_postgres/x15.html
- [7] *Lenguaje de programación: Java*. [Consultado el: 17 de enero de 2007]. Disponible en:
http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java
- [8] *Características de JAVA*. [Consultado el: 18 de enero de 2007]. Disponible en:
<http://www.iec.csic.es/CRIPTONOMICON/java/quesjava.html>
- [9] *Análisis y Diseño Orientado a Objetos*. [Consultado el: 30 de marzo de 2007]. Disponible en:
<http://www.dcc.uchile.cl/~luquerre/cc40b/index.html>
- [10] *Java Server Pages* [Consultado el: 18 de enero de 2007]. Disponible en:
http://es.wikipedia.org/wiki/Java_Server_Pages
- [11] *siRNA Design Tools*. [Consultado el: 6 de febrero de 2007]. Disponible en:
http://www.Ambion.com/techlib/misc/siRNA_tools.html
- [12] *siRNA Design Software*. [Consultado el: 6 de febrero de 2007]. Disponible en:
<http://i.cs.hku.hk/~sirna/software/sirna.php>
- [13] *Custom siRNA Design Tool*. [Consultado el: 6 de febrero de 2007]. Disponible en:
<http://www.dharmacon.com/sidesign/default.aspx>

- [14] *siDirect*. [Consultado el: 6 de febrero de 2007]. Disponible en: http://nar.oxfordjournals.org/cgi/content/full/32/suppl_2/W124#B11
- [15] *IDT Integrated DNA Technologies* [Consultado el: 6 de febrero de 2007]. Disponible en: <http://www.idtdna.com/Scitools/Applications/RNAi/RNAi.aspx>
- [16] *NetBeans IDE*. [Consultado el: 8 de marzo de 2007]. Disponible en: http://es.wikipedia.org/wiki/NetBeans#La_Plataforma_NetBeans
- [17] Sanchez Mendoza María A. *Metodologías De Desarrollo De Software*. [Consultado el: 9 de marzo de 2007]. Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html
- [18] *Comparación entre RUP y XP*. [Consultado el: 9 de marzo de 2007]. Disponible en: <http://www.javahispano.org/articles.article.action?id=76>
- [19] *Visual Paradigm*. [Consultado el: 9 de marzo de 2007]. Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/LabTr1_VP.pdf
- [21] Conallen, Jim. *UML Extension for Web Applications v0.91*. Marzo, 1999.

Bibliografía.

Sitios visitados y fecha de la visita:

- *Eclipse: una herramienta profesional al alcance de todos.* [Consultado el: 16 de enero de 2007]. Disponible en: http://www.gui.uva.es/~laertes/nuke/index.php?option=com_content&task=view&id=56&Itemid=41
- *¿Qué es el Tomcat?* [Consultado el: 16 de enero de 2007]. Disponible en: <http://es.wikipedia.org/wiki/Tomcat>
- *¿Qué es PostgreSQL?* [Consultado el: 16 de enero de 2007]. Disponible en: http://www.netpecos.org/docs/mysql_postgres/x15.html
- *Lenguaje de programación: Java.* [Consultado el: 17 de enero de 2007]. Disponible en: http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java
- *Características de JAVA.* [Consultado el: 18 de enero de 2007]. Disponible en: <http://www.iec.csic.es/CRIPTONOMICON/java/quesjava.html>
- *Análisis y Diseño Orientado a Objetos.* [Consultado el: 30 de marzo de 2007]. Disponible en: <http://www.dcc.uchile.cl/~luguerre/cc40b/index.html>
- *Java Server Pages* [Consultado el: 18 de enero de 2007]. Disponible en: http://es.wikipedia.org/wiki/Java_Server_Pages
- *siRNA Design Tools.* [Consultado el: 6 de febrero de 2007]. Disponible en: http://www.Ambion.com/techlib/misc/siRNA_tools.html
- *siRNA Design Software.* [Consultado el: 6 de febrero de 2007]. Disponible en: <http://i.cs.hku.hk/~sirna/software/sirna.php>
- *Custom siRNA Design Tool.* [Consultado el: 6 de febrero de 2007]. Disponible en: <http://www.dharmacon.com/sidesign/default.aspx>
- *siDirect.* [Consultado el: 6 de febrero de 2007]. Disponible en: http://nar.oxfordjournals.org/cgi/content/full/32/suppl_2/W124#B11

- *IDT Integrated DNA Technologies* [Consultado el: 6 de febrero de 2007]. Disponible en: <http://www.idtdna.com/Scitools/Applications/RNAi/RNAi.aspx>
- Sanchez Mendoza María A. *Metodologías De Desarrollo De Software*. [Consultado el: 9 de marzo de 2007]. Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html (9/03/2007)
- *Visual Paradigm*. [Consultado el: 9 de marzo de 2007]. Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/LabTr1_VP.pdf

Libros y artículos:

- Joseph, Schmuller. “Aprendiendo UML en 24 horas”, Prentice-Hall, Inc. 2001.
- Booch, G., Rumbaugh, J., Jacobson, I. “El Lenguaje Unificado de Modelado”. Addison-Wesley. 1999.
- Conallen, Jim. UML Extension for Web Applications v0.91. Marzo, 1999.

Anexos.

Anexo 1 "Diagramas de secuencia"

Diagrama secuencia CU Autenticar Usuario.

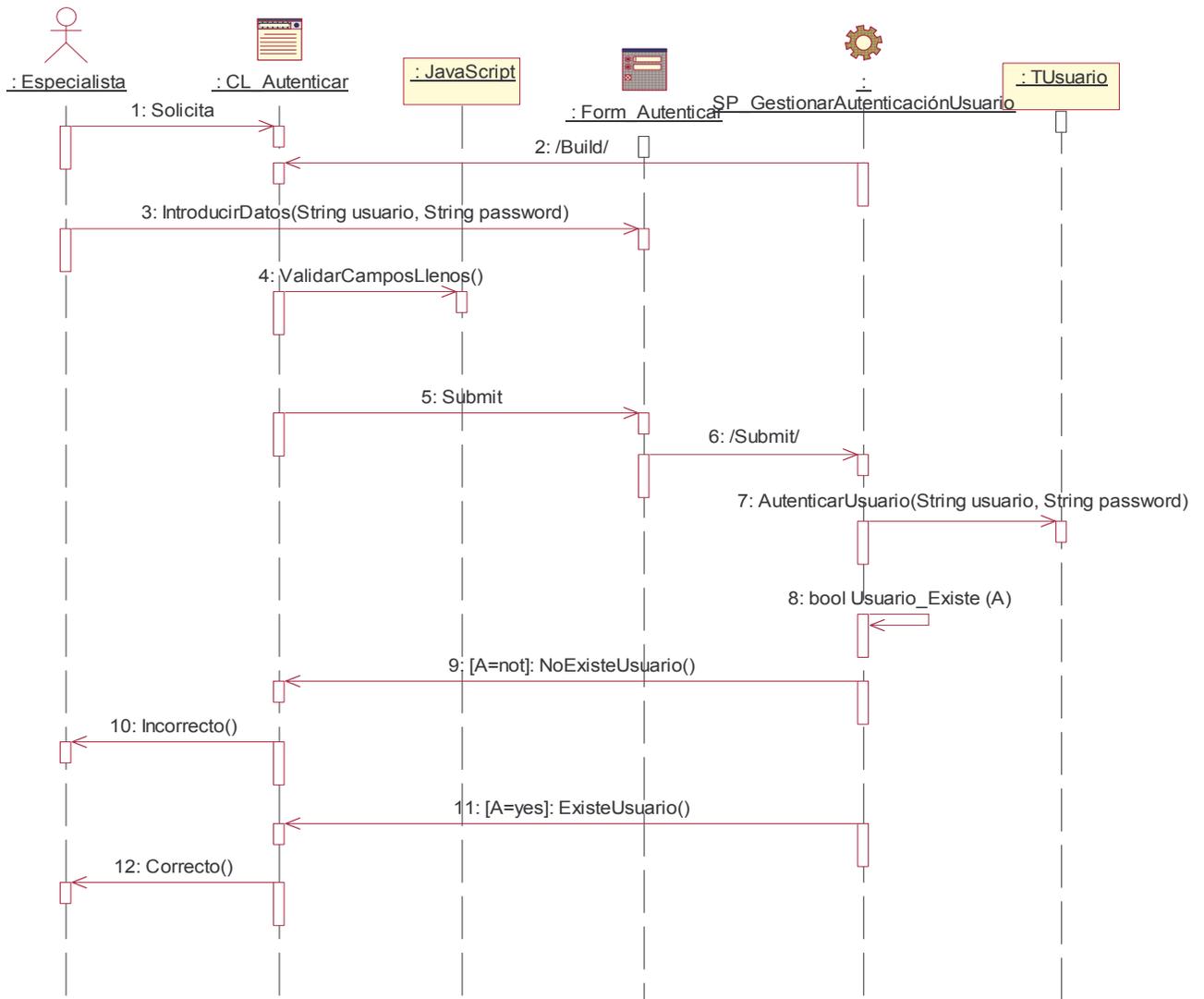


Diagrama secuencia CU Registrar Usuario.

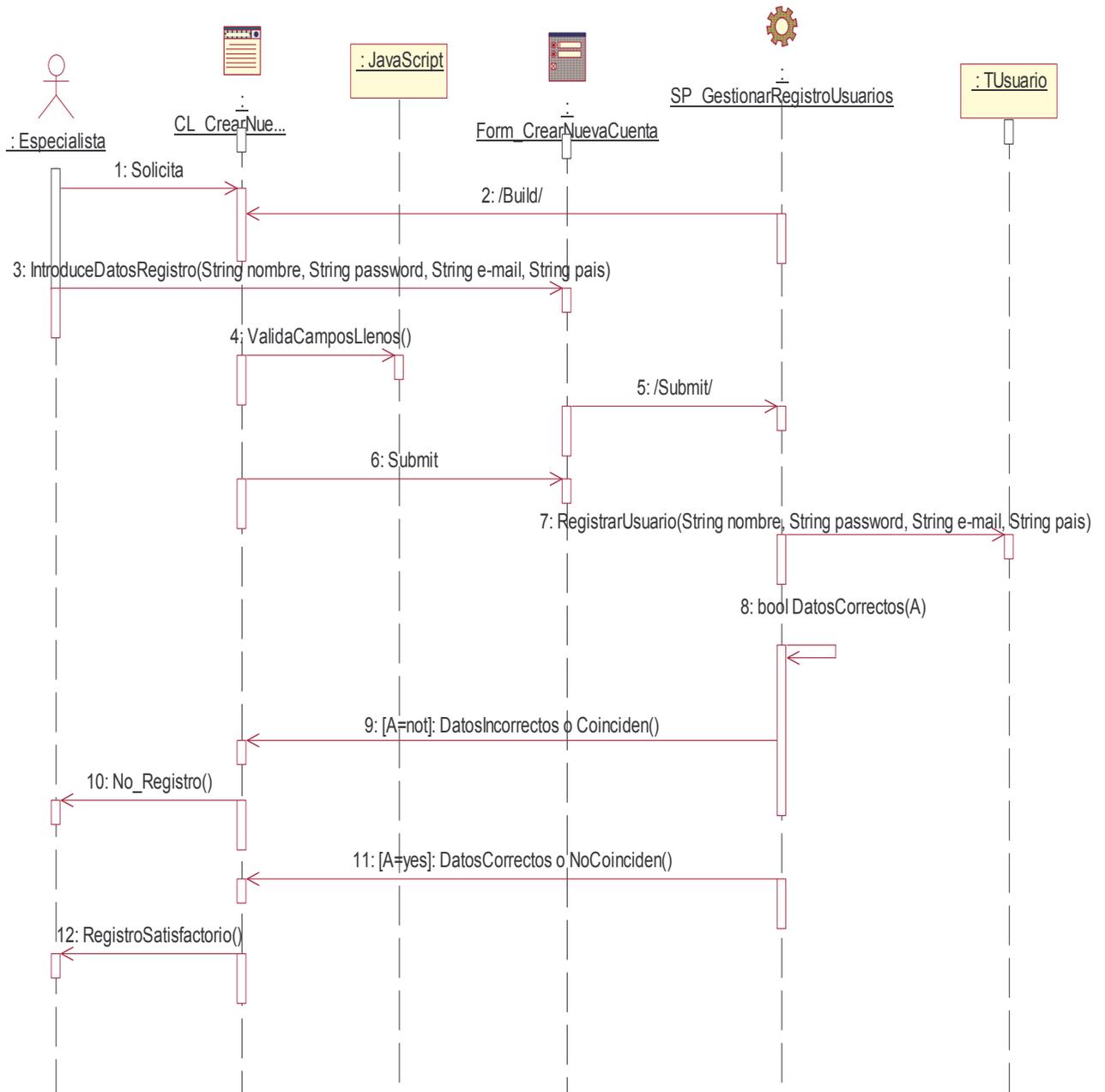


Diagrama secuencia CU Formar Secuencia mRNA. (Escenario 1) (Ver en el expediente del proyecto)

Diagrama secuencia CU Formar Secuencia mRNA. (Escenario 2)

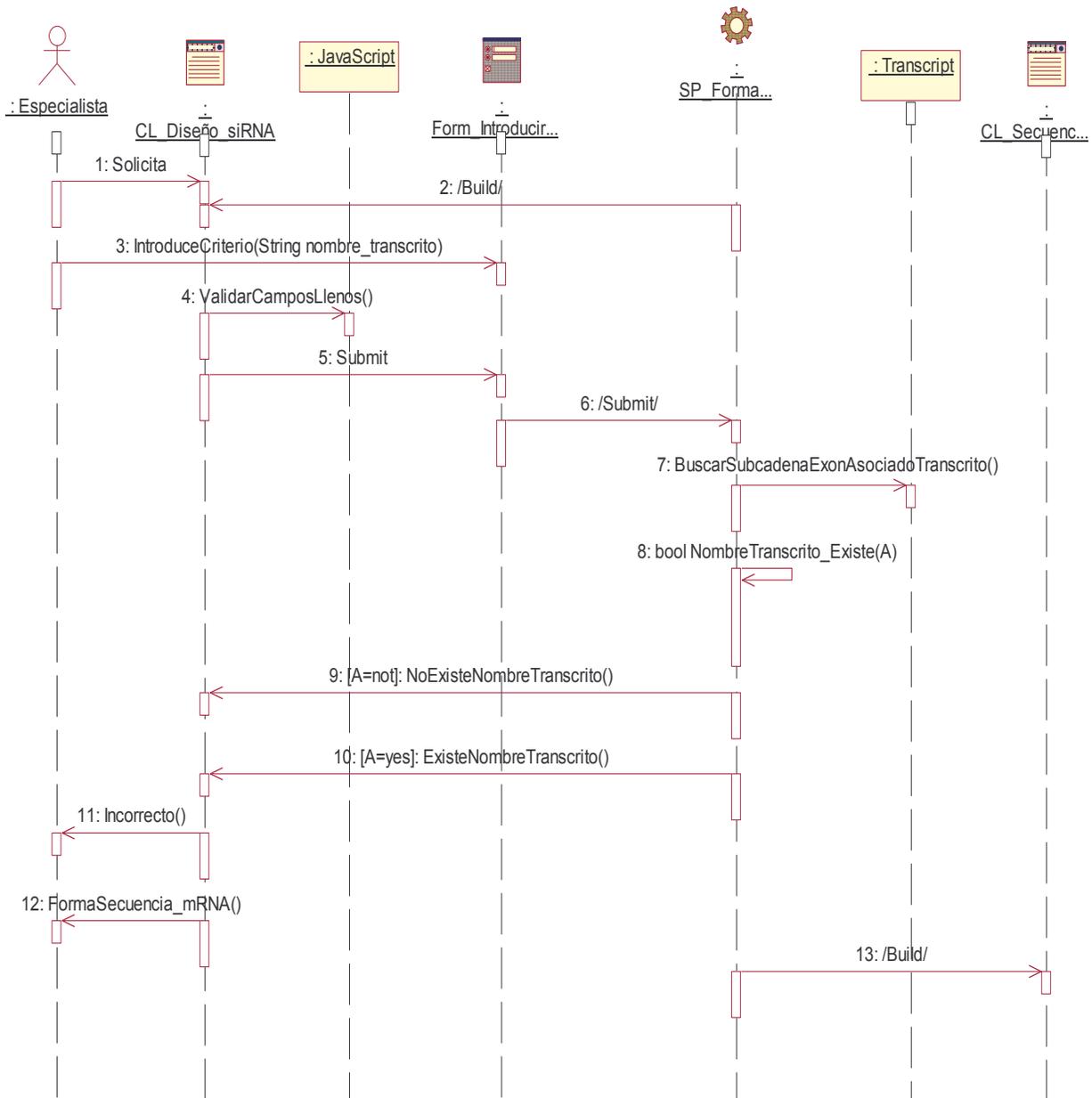


Diagrama secuencia CU Mostrar Genes.

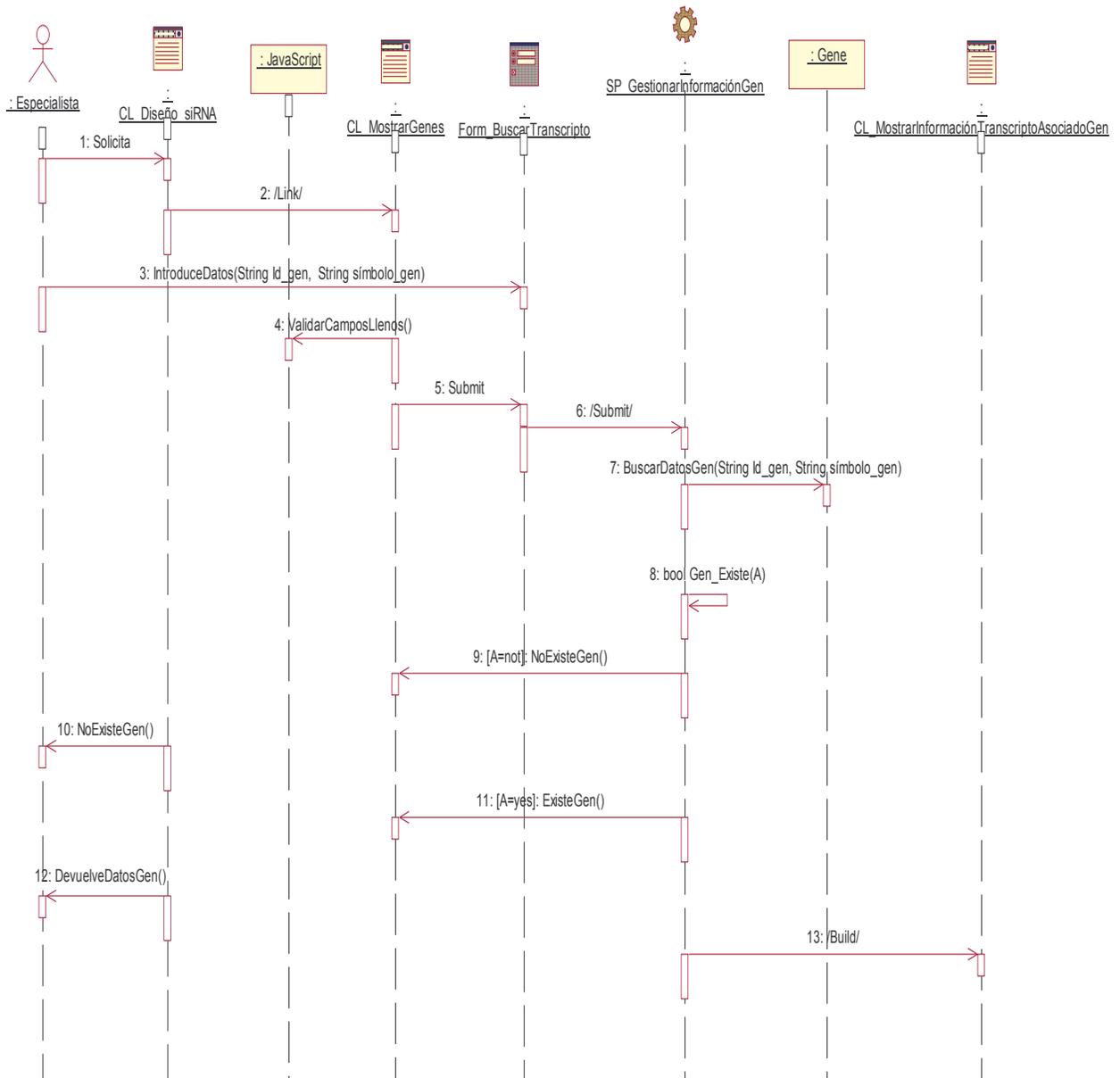


Diagrama secuencia CU Mostrar Transcrito.

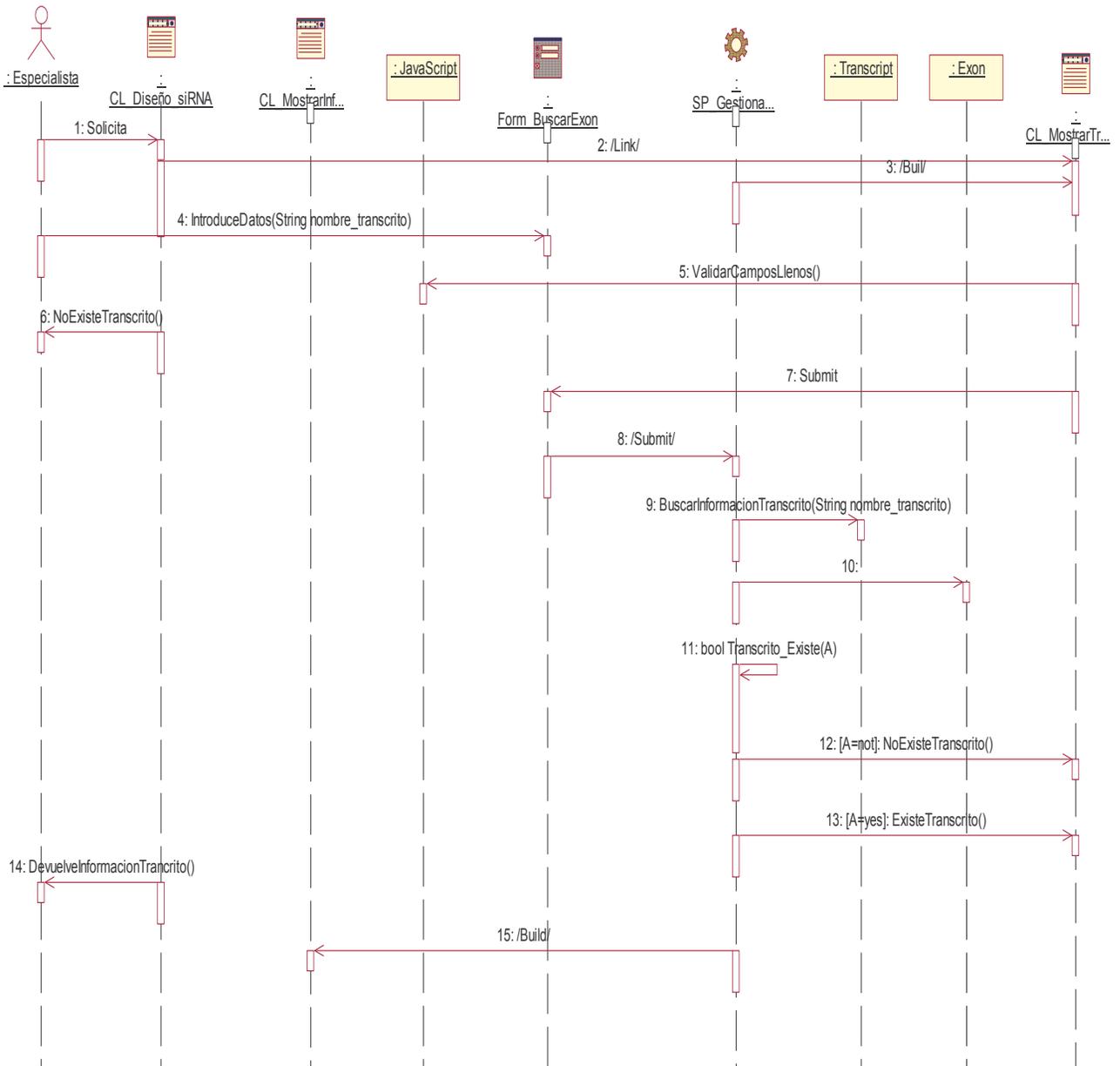


Diagrama secuencia CU Analizar Secuencia mRNA. (Escenario 1)

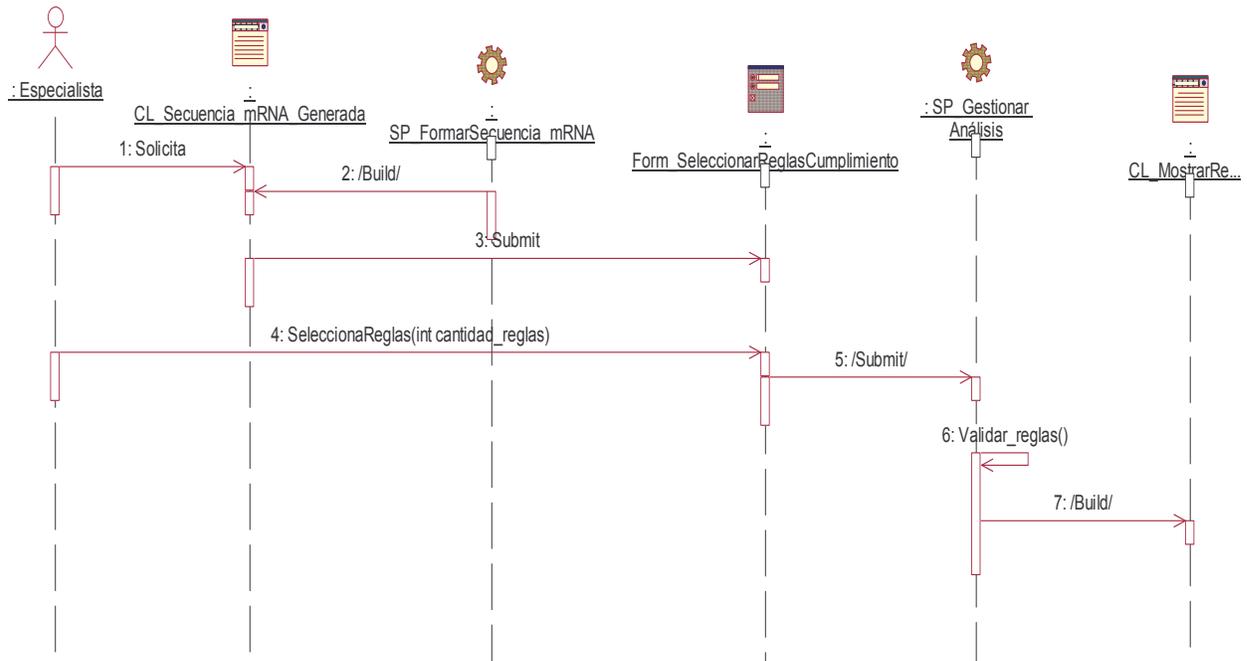
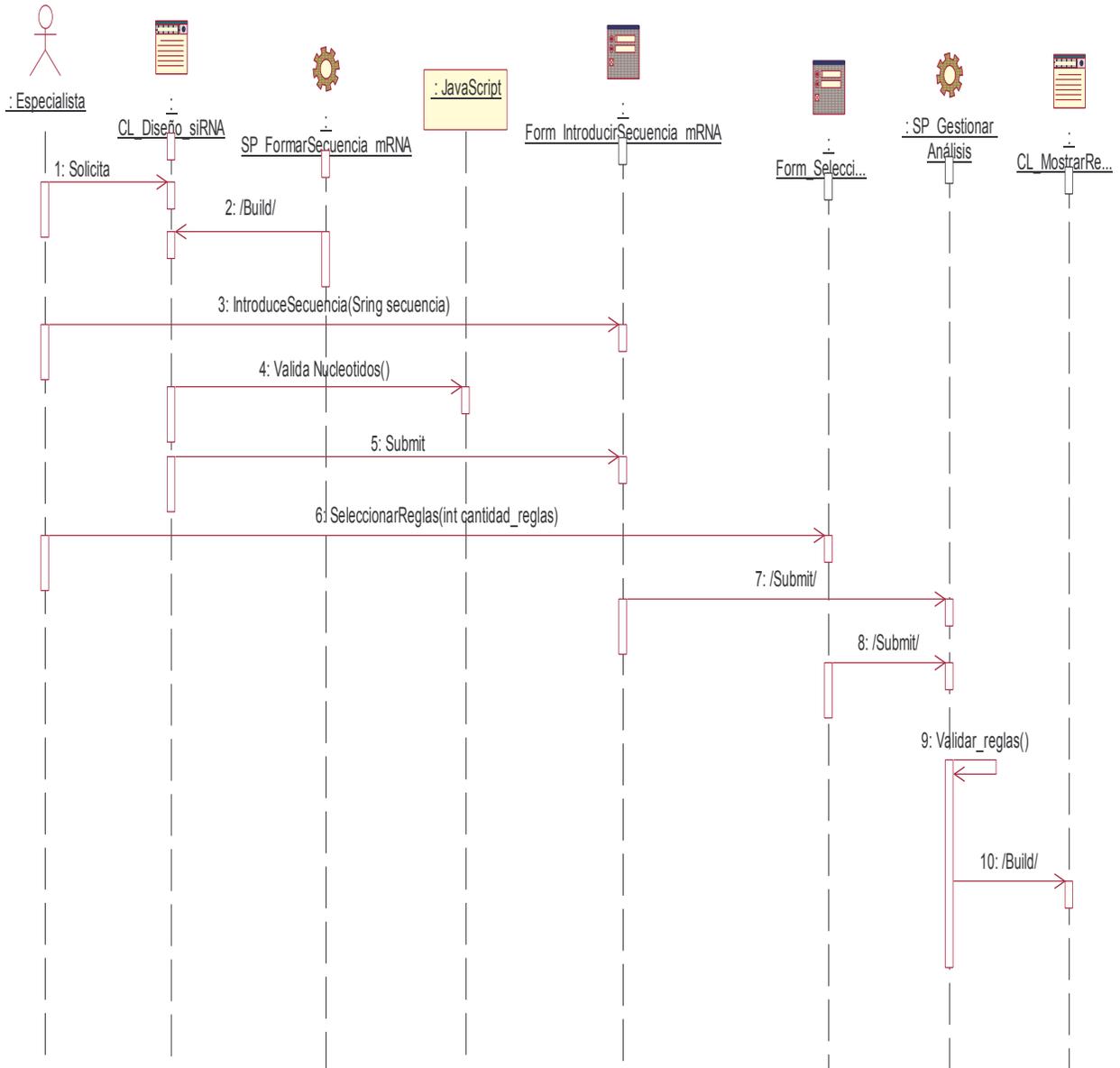


Diagrama secuencia CU Analizar Secuencia mRNA. (Escenario 2)



Glosario de Términos.

Inhibición: Es la condición de interrupción de alguna de las funciones de un determinado gen, para resolver alguna acción importante de la función de los genes en la expresión genética del organismo.

Transcrito: Molécula de ARN transcrita a partir de una hebra complementaria de ADN.

Gen: Es la unidad básica de herencia de los seres vivos y una secuencia lineal de nucleótidos en la molécula de ADN o ARN, que contiene la información necesaria para la síntesis de una macromolécula con función celular específica. El gen es considerado como la unidad de almacenamiento de información y unidad de herencia al transmitir esa información a la descendencia.

Cromosomas: Son las estructuras físicas de la célula eucariota que portan los genes. Estos cromosomas solo son visibles durante la división celular. Desde el punto de vista de su composición los cromosomas están formados de ADN y proteínas.

Exones: Son cada una de las regiones de un gen que contienen la información para producir la proteína codificada en el gen. Cada exón codifica una porción específica de la proteína completa.