



UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA EN INFORMÁTICA DE GESTIÓN

PROYECTO FIN DE CARRERA

LITTLE DNA ANALYZER

Pequeño analizador de secuencias de nucleótidos

Autor: Dolores García Pérez

Tutor: María Gregoria Casares Andrés



Agradecimientos

En primer lugar me gustaría agradecer a mi familia el apoyo dado para que finalmente presentara el Proyecto de Fin de Carrera. A mi madre por su insistencia y a mi padre por la idea y el tiempo que ha empleado en explicarme los conceptos básicos necesarios así como la definición de lo que realmente necesitaban.

No quisiera olvidarme tampoco de Miguel Calero, Director del Unidad Funcional de Investigación de Enfermedades Crónicas, Servicio de Encefalopatías Espongiformes del Instituto de Salud Carlos II, por haberme permitido hacer esta aplicación, que espero les facilite su trabajo.

A mis compañeros de Universidad, por su ayuda para sacar éste proyecto adelante. En especial a Alberto Rodríguez, que ha formado parte del proyecto en la distancia, con las largas tertulias y discusiones sobre cuáles eran los pasos a seguir.

No quiero olvidarme de mis compañeros de trabajo, gracias a los he aprendido gran parte de lo que sé ahora mismo y que han seguido la evolución del proyecto desde el principio.

A María Gregoria, tutora de éste proyecto, por haberme ayudado a presentarlo.

Y por supuesto, a mi novio Carlos, que es el que más ha sufrido éste Proyecto de Fin de Carrera.

Índice de contenido

1.	Introducción.....	5
2.	Objetivos, entorno y necesidad.....	7
2.1	Propósito y necesidades.....	7
2.2	Glosario	8
2.3	Alcance.....	11
2.4	Internacionalización	13
2.5	La aplicación	13
2.5.1	Los usuarios	14
2.5.2	Descripción funcional de la aplicación	15
2.5.3	Esquema de uso.....	28
2.5.4	Especificación de los casos de uso y diagramas de actividad.....	29
3.	Plan de pruebas	46
3.1	Pruebas Unitarias.....	46
3.2	Pruebas Funcionales	46
3.3	Pruebas de Rendimiento.....	47
4.	Gestión del proyecto	49
4.1	Metodologías ágiles. Introducción a SCRUM.	49
4.2	Adaptación de SCRUM al caso concreto	51
4.3	Sprint	52
4.4.1	Sprint 1	52
4.4.2	Sprint 2	54
4.4.3	Sprint 3	57
4.4.4	Sprint 4	58
5.	Entorno y Tecnología.....	61
5.1	Introducción.....	61
5.2	Tecnología Hardware	62
5.2.1	Tecnología Hardware en el Servidores	62
5.2.2	Tecnología Hardware en las estaciones de trabajo	63
5.3	Tecnología Software.....	63
5.3.1	Arquitectura	63
5.3.2	Tecnologías	64
5.4	Diagramas de Base de datos.....	68
5.5.1	Base de datos SQL.....	68
5.5.2	Base de datos NO SQL.....	69
6.	Presupuesto.....	71
7.	Manual de Referencia Técnica	72
7.1	Instalación del proyecto en el entorno de desarrollo.....	72
7.2	Ficheros de configuración de seguridad.....	74
7.3	Estructura del proyecto JAVA	75
7.4	Internacionalización	78
7.5	Estructura de la parte de la vista.....	79
8.	Pruebas Funcionales del proyecto	81



Administración	81
Análisis.....	82
9. Resultados.....	85
9.1 Pruebas de Carga	85
9.1.1 Caso Base	86
9.1.2 Forzar Usuario	88
9.1.3 Forzar Peticiones	89
9.1.4 Resumen de resultados	91
9.2 Pruebas sobre Análisis.....	92
10. Conclusiones.....	96
11. Futuras líneas de trabajos	97
12. Bibliografía y Referencias	98
13. Anexos.....	99



1. Introducción

Durante la última década se han producido grandes avances en secuenciación masiva de ADN que han permitido obtener las secuencias completas de ADN de cualquier organismo.

El siguiente paso consiste en analizar y llegar a un entendimiento profundo de la estructura y de la información que estas secuencias contienen.

La genómica es la ciencia encargada del estudio de la estructura y función del genoma, mientras que la Bioinformática es la disciplina que nació para facilitar el estudio de toda esta nueva información.

La Bioinformática se ha encargado de estructurar y desarrollar aplicaciones para el análisis de éstos datos biológicos, potenciando enormemente el estudio del genoma. A pesar del gran número de aplicaciones desarrolladas en el campo de la Bioinformática, a veces no es posible disponer de aplicaciones sencillas con finalidades muy concretas que facilite el trabajo de los investigadores. En ocasiones, a pesar de disponer de la herramienta necesaria para un estudio, el usuario necesita equipos o conocimientos informáticos de nivel alto para el uso de los mismos, por lo que al final, no resultan accesibles ni usables.

El proyecto de fin de carrera presentado, trata de solucionar un problema real que se presenta en la Unidad Funcional de Investigación de Enfermedades Crónicas, Servicio de Encefalopatía Espongiforme del Instituto de Salud Carlos III, en el que tras buscar algún tipo de desarrollo existente, no han encontrado ninguno capaz de cumplir sus expectativas y necesidades en estudio de epigenética. La epigenética es una parte de la genética, encargada del estudio de fenómenos que aunque no afectan directamente a la secuencia del ADN, si modulan su expresión y además son heredables. En algunos estudios epigenéticos es necesario buscar y cuantificar pequeñas secuencias de ADN dentro de dos contextos:

La localización de la aparición de una secuencia buscada en función de la distancia y frecuencia de su aparición en ventanas.

Fragmentos solapantes que cubran toda la secuencia de ADN del estudio

Los resultados de estos análisis suelen representarse en histogramas que resumen la información.

Podemos decir que cubrir esta necesidad, es el objetivo de éste proyecto.

Un ejemplo concreto en el que es necesario este tipo de análisis está relacionado con las enfermedades neurodegenerativas, donde estudios recientes sugieren que hay ciertos mecanismos epigenéticos que influyen en su regulación.

Estos mecanismos incluyen la metilación de regiones ricas en di-nucleótidos CG especialmente en zonas promotoras del ADN. Además, los di-nucleótidos CG metilados son inestables y provocan mutaciones, las mutaciones en CG constituyen uno de los puntos calientes para mutaciones en



líneas germinales y probablemente también en mutaciones somáticas. De hecho, las mutaciones en CG son entre 5 y 20 veces más frecuentes que otros tipos de mutaciones puntuales. Esta hiper-mutabilidad de los di-nucleótidos CG puede tener un papel en la etiología de distintas enfermedades degenerativas. Para este caso concreto, se requiere la búsqueda y representación gráfica de la frecuencia de aparición de las CG así como también de las posibles mutaciones que han podido darse en las regiones a estudio y que corresponderían a las secuencias CA y TG.

La solución que proponemos, da la posibilidad a analizar éste tipo de datos, facilitando la obtención de resultados y la presentación de los mismos. Se trata de una solución amigable para los investigadores en biomedicina o en general a los investigadores en ciencias de la vida. La solución será multiplataforma y de fácil uso, sea cual sea el conocimiento informático del usuario.

Se permitirá localizar, cuantificar y representar de forma gráfica la aparición de determinadas secuencias de pequeña longitud en el ADN, que pueden ser relevantes para una determinada función dentro de su estudio.

Con ésta solución se pretende reducir tiempo y simplificar el desarrollo de la investigación, ya que hasta el momento estos análisis solicitados se han realizado de forma manual o utilizando herramientas de ofimática.

Igualmente se agilizará el trabajo de análisis de regiones del genoma que pueden ser de especial interés en la biomedicina, permitiendo centrar los esfuerzos del investigar en el problema biológico y no en el análisis de los datos.

Además, la Genómica es una rama de la ciencia en continua evolución, debe tenerse en cuenta la flexibilidad de la aplicación para futuras necesidades de estudio o nuevas funcionalidades dentro del ámbito descrito, de forma que la aplicación pueda evolucionar y seguir siendo útil.

Éste documento contiene el análisis a nivel funcional y técnico de una aplicación web para el análisis de secuencias de ADN.

En primer lugar, vamos a definir cuáles son los objetivos del proyecto así como la necesidad concreta y la oferta del mercado actualmente respecto a dicha necesidad. Se va a crear un glosario de términos que nos ayudará a seguir más fácilmente la funcionalidad descrita en los distintos puntos del documento.

Se definirá un plan de pruebas de la aplicación, que nos permitirá asegurar que la aplicación es robusta a cambios y que permitirá detectar cualquier posible anomalía en el funcionamiento de la misma en caso de modificaciones o de nuevas funcionalidades. Además, se definirá un plan de pruebas de carga, en la que se someterá la aplicación a pruebas de estrés, gracias a las cuales se determinará si la aplicación tal y como está diseñada soportará un uso concurrente y con alto número de usuarios.

Se mostrará una planificación utilizando metodologías ágiles, que serán descritas y adaptadas en éste documento para el caso concreto de éste proyecto.



Se hará una introducción a las distintas tecnologías a nivel de *hardware* y *software*, así como distintos *frameworks* de desarrollo utilizados para la implementación de la aplicación. Se describirá igualmente cual es el diseño de la arquitectura del proyecto así como sus almacenamientos de datos.

Se incluirá un presupuesto en el que se contemplarán los costes de material y de los recursos que han intervenido en el proyecto.

Se incluirá un manual de referencia técnica que permitirá a cualquier persona encargada del mantenimiento o mejora de la aplicación instalar el entorno de desarrollo y conocer la configuración y estructura de la aplicación de una forma sencilla.

Por último, se incluirán resultados de rendimiento de la aplicación y conclusiones sobre el proyecto en general, analizando cuales son las futuras líneas de mejora o posibles nuevas funcionalidades que tendrían cabida en éste proyecto.

Además, se hará entrega de un manual de usuario detallado, para que cualquier persona interesada en utilizar la aplicación pueda hacerlo de manera sencilla y cómoda.

2. Objetivos, entorno y necesidad

El objetivo principal de este proyecto de fin de carrera es crear una aplicación amigable capaz de realizar un análisis o varios sobre una secuencia de nucleótidos, presentando los resultados de forma legible, entendible y útil, apoyada en gráficas para facilitar el entendimiento de los resultados complejos

2.1 Propósito y necesidades

La necesidad básica es dividir el número de veces que el usuario considere una secuencia de caracteres de longitud variable, buscando en cada división la secuencia de caracteres (cada carácter es realmente un nucleótido) que interesa en ese estudio. Dicha búsqueda y presentación posterior de resultados debe realizarse on-line y de la forma más rápida posible. Se necesita que la interfaz sea sencilla, amigable y cómoda, capaz de ser utilizada por cualquier usuario, aunque no tenga formación sobre dicha aplicación. Además, se necesita que el tratamiento de los datos y la presentación del informe final sean legibles y fáciles de entender.

Sobretudo necesitamos una aplicación que facilite la vida al usuario, no encontrando éste trabas para el acceso a la misma. Por lo tanto, debe poder utilizarse independientemente del sistema operativo que se utilice y el equipo desde el que se haga, no debe consumir un alto número de recursos del ordenador del usuario. Cualquier operación de actualización o mejora sobre la aplicación debe ser transparente, permitiendo trabajar sobre la aplicación sin que haya necesidad de actualizaciones por parte del usuario.

Por todo esto, se decide implementar una aplicación Web, facilitando el acceso, que sólo será necesario un ordenador con navegador web actualizado y con acceso al servidor en el que se encuentre desplegada la aplicación.

Se debe proporcionar una Base de Datos moldeable, capaz de adaptarse a modificaciones en las necesidades de búsqueda del usuario y que permita nuevas funcionalidades fácilmente. Por ello, debe tratarse de un sistema abierto a cambios, capaz de amoldarse a nuevas necesidades que aparezcan en el futuro, fruto de los cambios constantes en la investigación en el ámbito que nos ocupa.

El principal objetivo de éste proyecto, es proporcionar a los investigadores una herramienta sencilla, que les haga olvidarse del análisis de éstos datos, centrándose únicamente en el objeto de su estudio.

A continuación, vamos a ampliar los conceptos básicos que anteriormente se han expuesto y que resultan claves para el desarrollo de éste proyecto, mediante un pequeño glosario de términos.

2.2 Glosario

A continuación se van a describir los principales términos que facilitarán el entendimiento de las necesidades de la aplicación así como de la funcionalidad de la misma.

Bases de datos biológicas^[1] y de secuencias

Se trata de bibliotecas en las que se almacena la información sobre las ciencias que se ocupan del estudio de los seres vivos, tales como la recogida de experimentos científicos literatura publicada, tecnologías de experimentación... Contienen información de áreas como la genómica, proteómica, metabolómica, expresión génica mediante microarrays y filogenética entre otras. Incluyen información de funciones, estructura y localización de genes, efectos clínicos de mutaciones y similitudes entre secuencias y distintas estructuras biológicas. El diseño de estas bases de datos pertenece a la disciplina de la Bioinformática, que puede definirse como la aplicación de la tecnología de computadores a la gestión y análisis de datos biológicos.

Estas bases de datos biológicas son hoy día un instrumento básico de ayuda a los científicos para lograr explicar y comprender ciertos fenómenos biológicos o enfermedades.

Este conocimiento biológico se distribuye entre múltiples bases de datos generales y especializados, lo que complica la garantía de la coherencia de la información.

Uno de los tipos más comunes son las bases de datos de secuencias, en las que se guardan colecciones de secuencias de ADN, proteínas y nucleótidos. Estas bases de datos se consideran primarias si la información que contienen es directamente la secuencia y estructura. Se consideran secundarias si lo que guardan son los datos e hipótesis derivados del análisis de las bases de datos primarias. Uno de los principales problemas que se encuentran estas grandes bases de datos es que los registros provienen de muy diversas fuentes, por lo que las secuencias o las anotaciones adjuntas varían notablemente en calidad. Otro de los problemas es la redundancia, ya que muchas veces la misma secuencia se encuentra replicada o muy similar se encuentra en distintas bases de datos.

Las bases de datos más relevantes en biología incluyen datos de secuencias de nucleótidos, proteínas, estructura de proteínas, genomas, expresión genética, bibliografía, taxonomía, metabolismo, factores de transcripción, etc. En nuestro caso, estaremos interesados en las



secuencias de nucleótidos, de las que existen tres bases de datos principales, cuya colaboración hace posible acceder a casi toda la información desde cualquier de las tres sedes:

- EMBL-BANK ^[2] en el instituto europeo de Bioinformática
- DNA Data Bank of Japan ^[3] en el centro de Información Biológica
- GenBank ^[4] en el centro Nacional de Información Biotecnológica

Son mantenidas por organismos en distintos países, pero de forma coordinada, por lo que una secuencia enviada a cualquiera de ellas se verá reflejada en las otras dos con una semana de latencia aproximadamente, ya que ésta es la frecuencia de actualización entre estas bases de datos. Por este motivo, resulta indistinto qué base de datos se utilice, ya que la información a la que se accede, será la misma o prácticamente la misma. En éste proyecto en concreto accederemos a EMBL-BANK, que permite la consulta de datos mediante servicios REST publicados por ellos mismos.

Servicio Rest

Las siglas REST derivan de Representational State Transfer, es decir una transferencia de representación de estado. La clave de REST es que los servicios no tienen estado, entre dos llamadas que se realizan a los servicios, se pierden los datos, que no serán recuperables en una llamada posterior, siendo el consumidor de los servicios el que deba recordar estos datos en la siguiente petición. Esto claramente tiene la desventaja de tener que enviar el estado en cada llamada, pero la gran ventaja es la escalabilidad. Generalmente estos estados que necesitan ser guardados se almacenan en memoria, de tal forma que si es el servidor el que debe almacenar estos estados es relativamente sencillo quedarnos sin memoria, por tener demasiados datos en sesión o si hay muchos usuarios simultáneamente. Por lo tanto, evitando que sea el servidor el que controla este estado, sino que lo hace el cliente, evitamos este problema.

La aplicación consumirá los servicios REST que encontramos disponibles para acceder a la información de la base de datos EMBL-BANK.

Nucleótidos

Los nucleótidos Adenina (A), Citosina (C), Guanina (G), Timina (T) y Uracilo (U), son moléculas orgánicas que constituyen los monómeros de los ácidos nucleicos. Un monómero, es una molécula de pequeña masa molecular, que unida a otros monómeros por medio de enlaces químicos, forma macromoléculas o polímeros. EL ADN está compuesto de las bases o nucleótidos A, C, G y T, mientras que el ARN está compuesto de A, C, G y U. Podemos diferenciar estas bases o nucleótidos en dos tipos:

- Purínicas: A y G
- Pirimidínicas: C, T y U

Ambos tipos se complementan químicamente para formar las estructuras de ADN y ARN, emparejándose una purínica con una pirimidínica de acuerdo a AT y CG para el ADN o AU y CG para el ARN.



El ADN está compuesto por una secuencia de nucleótidos. Esta secuencia, es la responsable del almacenamiento de la información, no sólo para la codificación de las proteínas, sino también para la regulación de su expresión a nivel celular y de organismo. Las proteínas son moléculas formadas por cadenas lineales de aminoácidos y su secuencia viene codificada por la secuencia de nucleótidos del gen correspondiente.

Por lo tanto, el estudio de la estructura de las secuencias y sus variaciones, proporciona valiosa información sobre los mecanismos que regulan los organismos y sus posibles alteraciones.

Ventana

Para realizar el análisis debe dividirse la secuencia de nucleótidos. La ventana es el número de nucleótidos que va a incluirse en cada división. El tamaño de ventana debe ser siempre mayor que uno y menor o igual al tamaño de la secuencia completa.

Estas ventanas van a servir para establecer regiones del ADN que presentan propiedades diferenciales y que pueden estar implicadas en una determinada función. De ésta forma es posible analizar las variaciones que se presentan en la población y su relación con las alteraciones de dicha función.

Solapamiento

Además de definir el tamaño de la ventana, también es necesario definir el grado de solapamiento entre ventanas. El solapamiento indica el desplazamiento sobre la secuencia antes de seleccionar cual será el primer carácter de la siguiente ventana. Es decir, si nuestro solapamiento es dos, la primera ventana empezará en el nucleótido 1 y la siguiente en el nucleótido 3. El tamaño mínimo de solapamiento es uno, donde la siguiente ventana siempre empezará con el segundo nucleótido de la anterior. El solapamiento nunca podrá ser superior al tamaño de la ventana.

El ajuste del tamaño del solapamiento permite simplificar los cálculos a realizar en función del tamaño de las secuencias que se quiere estudiar. Cuando se emplean secuencias grandes, el tamaño de la ventana junto con el del solapamiento hace posible agilizar el proceso de búsqueda de la información más relevante, haciendo posible centrarse en un análisis más fino de las regiones del genoma que tengan más interés.

Secuencia de búsqueda

Corresponde a la secuencia de nucleótidos que deseamos buscar. El sistema localizará todas las ventanas que tengan al menos una ocurrencia de esta secuencia, contando el número de veces que se repite la secuencia en esa ventana.

La secuencia buscada nunca puede ser mayor que el tamaño de la ventana seleccionada. Se limitará el tamaño de la secuencia buscada a 10 nucleótidos, ya que consideramos que éste tamaño es suficiente para la resolución del problema que se aborda, y porque en el caso de secuencias superiores, el número de ocurrencias va a ser muy bajo. Para éstos casos, existen otros sistemas de búsqueda y la presentación de los resultados no requiere de histogramas.



2.3 Alcance

El sistema de análisis, de ahora en adelante Little ADN Analyzer, debe controlar el acceso de los distintos usuarios con nombre de usuario y contraseña, siendo capaz de distinguir a que datos se les da acceso y a cuáles no.

Cada usuario podrá pertenecer a más de un estudio y al menos deberá pertenecer a uno para poder acceder a la aplicación. De ésta forma, tendrá acceso a todos los análisis que ha efectuado él y a todos los análisis efectuados en los distintos estudios en los que participe.

Debe darse al usuario facilidades para la carga de los datos en la aplicación, de forma que se podrá hacer de distintas formas:

- Por consulta directa a la Base de Datos biológica EMBL-BANK, introduciendo el identificador único del nucleótido.
- Mediante la carga de un fichero de tipo FASTA o TXT, que deberá validarse para confirmar si la información que contiene es correcta.
- El usuario introduce la secuencia en formato texto por pantalla, debiendo validarse si la información es correcta.
- La secuencia ya ha sido utilizada en la aplicación y el usuario conoce el identificador que tiene asignado, por lo que podrá realizar una búsqueda para utilizar dicha secuencia.

Todas las secuencias de nucleótidos que se introduzcan para su análisis deben incluir el identificador único para posteriores búsquedas de resultados, dando también la opción de añadir una pequeña descripción del mismo, en caso de que se considere necesario.

Se debe dar la opción al usuario de seleccionar la ventana, solapamiento y secuencia de búsqueda de forma que sobre una misma secuencia de nucleótidos puedan hacerse análisis de distintos casos, identificando los resultados sobre estos parámetros.

Los resultados del análisis serán guardados, de forma que el usuario pueda acceder a ellos posteriormente, pero se dará la opción de eliminarlos, si se considera que no son necesarios en el estudio.

Tras las conversaciones mantenidas con los usuarios que solicitan ésta aplicación y tras el análisis posterior de las herramientas disponibles en el mercado, no se ha encontrado ninguna que cumpla con los criterios y necesidades que se exponen en este proyecto.

Actualmente, estos análisis que son necesarios en las labores del equipo de investigación, se realizan mediante hojas de cálculo, lo que hace la preparación de los datos y el posterior análisis algo tedioso, lento y poco ágil.

De todas las herramientas analizadas y que estaban disponibles para el equipo de análisis, la que más se acerca a cumplir los objetivos propuestos en cuanto a funcionalidad es SeeDNA ^[5]

Básicamente lo que hace es una representación gráfica en dos o en tres dimensiones de la frecuencia de aparición de cualquier secuencia en función de su tamaño.



La salida del histograma es un cuadro con el símbolo de cada nucleótido en el vértice. El número de divisiones del cuadro es dependiente del tamaño de secuencia de búsqueda seleccionada. Por lo tanto, al tener 4 nucleótidos (A, T, G y C), si elegimos tamaño dos, el cuadrado se subdivide en 16, que son las posibles combinaciones de cuatro elementos tomados de dos en dos. Para el caso de tamaño tres, tendremos un total de 64 cuadrados, que son el total de combinaciones de tres elementos tomados de tres en tres. Por lo tanto, cada cuadrado se corresponde con una secuencia de nucleótidos y su tamaño se corresponde con el número de repeticiones en la secuencia.

No permite dividir en ventanas ni seleccionar el desplazamiento en la secuencia, de la misma forma que no permite seleccionar una única secuencia de búsqueda que puede ser la única que nos interesa, ya que por defecto calcula todas.

Esto puede ser una ventana en el caso de secuencias cortas, pero si el análisis se realiza sobre una secuencia larga, no podemos diferenciar en que zona de la secuencia tenemos mayor número de repeticiones de una combinación de nucleótidos.

Se trata de una aplicación que debe ser instalada sobre una plataforma Linux (Ubuntu), pero que actualmente, debido a la poca actualización y abandono de la misma resulta prácticamente imposible de instalar. A pesar de ello, el autor ha puesto a disposición del equipo una nueva versión mejorada, pero cuyo uso no ha satisfecho al equipo, tanto por la complejidad de la configuración como por la escasez de los resultados que ofrece.

Podemos destacar entre las herramientas estudiadas algunas aplicaciones que emplean k-mer^[6] y su aplicación al ensamblado de datos de secuenciación masiva. El término K-mer hace referencia a todas las posibles subcadenas de longitud K que contiene una cadena. Suele utilizarse en genómica computacional, en la que k-meros hace referencia a todas las posibles subsecuencias de longitud k obtenidas a partir de una lectura obtenida tras la secuenciación de ADN. Esto básicamente es la aplicación de la división en ventanas de nuestra secuencia, con un desplazamiento dado.

Las aplicaciones estudiadas que hacen uso de k-mer fundamentalmente se emplean en el análisis de datos que provienen de las técnicas de secuenciación de alto rendimiento, más conocidas como [Next-generation DNA sequencing](#)^[7].

Algunas de éstas aplicaciones son:

- jellyfish-2.2.0
- kebabs_1.0.4
- kmer_counter

Pero aunque tienen ciertas características comunes a las necesidades de éste proyecto, como es la división de las secuencias con el uso de k-mer, sus necesidades cubren otra serie de objetivos que hacen que no se amoldan al completo a las necesidades expuestas en nuestro caso.

Por último destacamos DNA Pattern Find, que ofrece características de búsqueda para secuencias pequeñas. En ésta aplicación es posible introducir una o más secuencias junto a un patrón de búsqueda. El resultado que presenta la aplicación informa del número de repeticiones y de las



posiciones de la secuencia que concuerdan con el patrón. Es decir, esta aplicación cubre la necesidad de encontrar zonas de la secuencia que pueden tener cierto interés por el número de repeticiones de la combinación de nucleótidos buscada, acotando de ésta forma la búsqueda y el estudio de la secuencia.

Sin embargo, el informe de resultados que presenta, es un documento de texto plano con el siguiente formato:

```
Results for 780 residue sequence "sample sequence one" starting "actgttacct"
>match number 1 to "cgc" start=24 end=26 on the direct strand
cgc
>match number 2 to "cgc" start=26 end=28 on the direct strand
cgc
>match number 3 to "cgc" start=557 end=559 on the direct strand
cgc
>match number 4 to "cgc" start=76 end=78 on the reverse strand
cgc
>match number 5 to "cgc" start=25 end=27 on the reverse strand
Cgc
```

Resulta evidente que este formato de resultados no es gráfico ni cómodo a la hora de exportar la información para su tratamiento. Además, en el caso de secuencias largas, resulta poco legible.

2.4 Internacionalización

La internacionalización de la aplicación implica hacerla accesible para distintos idiomas o lenguajes. Básicamente consiste en que, en función de lo que necesite el usuario, la aplicación pueda ser vista en un idioma u otro, sin necesidad de cambiar el código, o cambiándolo lo menos posible.

La aplicación debe estar internacionalizada, al menos con traducción en inglés y español. Por defecto, la aplicación se mostrará en el lenguaje que el usuario tenga configurado en su navegador web. En caso de que el idioma del navegador no sea ninguno de los disponibles en la aplicación, se mostrará en inglés.

2.5 La aplicación

Las aplicaciones que ahora existen para análisis de secuencias de nucleótidos son más complejas y tienen un enfoque completamente distinto al dado en este proyecto.

Generalmente estas aplicaciones que ya existen son aplicaciones nativas que deben ser instaladas en el ordenador del usuario. Esto supone una complicación, ya que por un lado dependen del sistema del operativo del usuario, haciendo necesaria la instalación de máquinas virtuales con sistemas



operativos concretos, es decir, hacen necesario un nivel de conocimientos informáticos altos. Por otro lado, muchas de ellas fueron desarrolladas hace tiempo, lo que complica su instalación por problemas de compatibilidad, además el desarrollo puede llegar a estar obsoleto.

Por último, no permiten el trabajo compartido por parte de varios usuarios ni guardar ésta información para posteriores análisis y comparativas, ya que están basadas en análisis puntuales de secuencias concretas. Con ésta aplicación queremos darle una vuelta al análisis de los datos y adaptar la forma de trabajo real de los laboratorios a la aplicación, dando la posibilidad de que varias personas trabajen sobre un mismo proyecto en distintas secuencias. Además, al tratarse de una aplicación web, no es necesario que todos los usuarios trabajen sobre el mismo ordenador para hacer los estudios, ya que cualquiera con acceso a internet tendrá acceso a la aplicación y por lo tanto a los estudios a los que ha sido asociado.

2.5.1 Los usuarios

Se definen dos roles de usuario:

- Rol Administradores - Administrador
- Rol de Análisis - Usuario

Por el simple hecho de estar dado de alta en la aplicación todos son los usuarios tendrán rol de análisis. Para poder realizar análisis, deberán haber sido dados de alta en un estudio. En caso de que el usuario haya sido dado de alta, pero no pertenezca a ningún estudio, no podrá realizar ninguna acción de análisis desde la aplicación.

El usuario rol de administrador tendrá todas las opciones dadas al usuario con rol de análisis y además recae sobre él la gestión de la aplicación. El usuario con rol de administrador podrá:

- Dar de alta nuevos usuarios
- Modificar la información de usuarios ya existentes
- Reiniciar la contraseña de usuarios ya existentes
- Dar de alta nuevos estudios
- Modificar la información relativa a los estudios así como los usuarios que están incluidos dentro de él.
- Dar por finalizado un estudio, momento en el que se eliminarán todos los datos de respuesta de análisis que se hubieran guardado para dicho estudio.
- Gestión de secuencias existentes

Nombre Rol	Funcionalidad a la que accede
Usuario – Rol análisis	Tiene acceso a la modificación de su propia contraseña. Tendrá acceso a toda la funcionalidad de análisis y de búsqueda de resultados.
Administrador	Tendrá acceso a toda la funcionalidad definida para el Usuario o Rol de Análisis. Además, tendrá acceso a la administración de la aplicación.

2.5.2 Descripción funcional de la aplicación

A continuación, vamos a describir las principales funcionalidades de la aplicación.

Autenticación

Se trata de la pantalla de entrada a la aplicación para todos los usuarios que tengan acceso a la aplicación. Se solicitará al usuario que introduzca el nombre de usuario y la contraseña.

En caso de que el nombre de usuario o la contraseña no sean correctos, no se permitirá al usuario acceder a la aplicación.

Si el usuario y la contraseña son correctos, podrá acceder a la aplicación, pero no podrá realizar ningún análisis.

Puntos comunes

Una vez el usuario ha accedido a la aplicación aparecerá en la parte superior derecha de la pantalla el nombre de usuario. Al lado del nombre de usuario aparecerá un botón que permitirá al usuario abandonar la aplicación.

Debajo de la cabecera arriba explicada, aparecerá un menú con las distintas partes accesibles de la aplicación. Desde este menú, se podrá acceder a cualquier de ellas, siempre y cuando el rol que tenga lo permita.

Administración

La entrada a la parte de administración aparecerá para todos los usuarios en el menú superior de la pantalla. Para el caso de un usuario sin rol de administrador, al entrar desde el menú, se mostrará directamente la pantalla de reinicio de contraseña de usuario, ya que no puede acceder a ninguna otra funcionalidad.

Para el usuario administrador, se mostrará un menú que le dará la posibilidad de acceder a:

- Reinicio de contraseña
- Búsqueda de usuarios
- Búsqueda de estudios
- Gestión de secuencias

Reinicio de contraseña de usuario

Todos los usuarios tendrán disponible esta opción, en la que deberá introducir dos veces la nueva contraseña para permitir el cambio de la misma.



Búsqueda de usuarios

Se presentará por pantalla un formulario de búsqueda en el que aparecerán los siguientes criterios:

Nombre	Descripción
Nombre de usuario	Nombre de usuario mediante el cual el usuario acceder a la aplicación
Nombre	Nombre del usuario
Apellido 1	Primer apellido del usuario
Apellido 2	Segundo apellido del usuario

Se mostrarán los resultados de la búsqueda en una tabla que contendrá la siguiente información:

Nombre	Descripción
Acciones	Aparecerán dos botones: - Editar usuario: Al seleccionarlo se mostrará la pantalla de edición de usuarios - Cambio Contraseña: Al seleccionarlo se mostrará la pantalla de cambio de contraseña.
Nombre de usuario	Nombre de usuario mediante el cual el usuario acceder a la aplicación
Nombre	Nombre del usuario
Apellido 1	Primer apellido del usuario
Apellido 2	Segundo apellido del usuario

Aparecerá también en pantalla el botón de creación de nuevo usuario, que nos permitirá crear un nuevo usuario para la aplicación.



Creación de usuarios

Se mostrará un formulario con todos los datos del usuario:

Nombre	Tipo	Obligatorio	Descripción
Nombre de usuario	Texto	SI	Nombre con el que el usuario accederá a la aplicación.
Nombre	Texto	SI	Nombre del usuario
Apellido 1	Texto	SI	Apellido del usuario
Apellido 2	Texto	NO	Segundo apellido del usuario
Email	Texto	SI	Email de contacto del usuario
Rol Administrador	Check	NO	Indica si el nuevo usuario tiene rol de administrador.

Una vez completados se deberá pulsar el botón aceptar. Por defecto, la contraseña que se dará al usuario será X.

Edición de usuarios

Se presentará una pantalla con toda la información del usuario.

Nombre	Tipo	Editable	Obligatorio	Descripción
Nombre de usuario	Texto	NO	No aplica	Se trata del nombre de usuario, no es posible editarlo
Nombre	Texto	SI	SI	Nombre del usuario
Apellido 1	Texto	SI	SI	Apellido del usuario
Apellido 2	Texto	SI	NO	Segundo apellido del usuario
Email	Texto	SI	SI	Email del usuario
Rol Administrador	Check	SI	NO	Indica si el usuario tiene rol de administrador



Reinicio de contraseña usuarios

Se presentará a nivel informativo el nombre del usuario para el que se va a modificar la contraseña.

El funcionamiento será idéntico al descrito en la pantalla de “cambio de contraseña”.

Búsqueda de estudios

Se presentara por pantalla un formulario de búsqueda con los siguientes criterios:

Nombre	Descripción
Nombre	Nombre del estudio. Podrá incluirse en la búsqueda comodines para realizar búsquedas aproximadas por nombre de estudio.

Los resultados se mostrarán en una tabla con la siguiente información:

Nombre	Descripción
Acciones	<p>Acciones que pueden ejecutarse desde el estudio:</p> <ul style="list-style-type: none"> - Finalización de un estudio: Da por finalizado el estudio, eliminándolo y eliminando todos los resultados guardados para él. Antes de ejecutar la acción, se solicitará la confirmación del usuario mediante un mensaje en un pop-up: “¿Desea realmente finalizar éste estudio? Esta acción no es reversible y se eliminarán todos los datos asociados al mismo.”. En caso afirmativo, se llevará a cabo la eliminación del estudio y todos los resultados asociados. - Edición de estudio: Se mostrará la pantalla de edición del estudio.
Nombre	Nombre del estudio
Fecha de alta	Fecha en la que ha sido dado de alta
Descripción	Descripción del estudio.

En ésta pantalla, se incluirá un botón que redirigirá al usuario a la pantalla de creación de nuevos estudios.



Creación de estudio

Al acceder a ésta pantalla, se mostrará un formulario en el que se solicitarán los datos básicos del estudio, que son los siguientes.

Nombre	Tipo	Obligatorio	Descripción
Nombre	Texto	SI	Se indicará el nombre del estudio. Deberá tener como máximo 10 caracteres.
Fecha	Texto	SI	Será un campo autocompletado con la fecha del día en que se está dando de alta el estudio. No podrá ser editado por el usuario
Descripción	Texto	NO	Breve descripción de como máximo 1000 caracteres del estudio

Debajo de los datos básicos, se mostrará un formulario de búsqueda de usuarios, en la que se podrán seleccionar aquellos usuarios que van a formar parte del estudio. Los criterios de búsqueda serán los siguientes:

Nombre	Tipo	Descripción
Nombre de usuario	Texto	Nombre con el que el usuario acceder a la aplicación
Nombre	Texto	Nombre del usuario
Apellido 1	Texto	Primer apellido del usuario
Apellido 2	Texto	Segundo apellido del usuario
Email	Texto	Email del usuario



Los resultados de la búsqueda se mostrarán en una tabla bajo el título: “Usuarios buscados”. No se mostrarán en ésta tabla los usuarios ya asociados al estudio:

Nombre	Tipo	Descripción
Añadir usuario	Botón	Asocia el usuario al estudio
Nombre de usuario	Texto	Nombre con el que el usuario acceder a la aplicación
Nombre	Texto	Nombre del usuario
Apellido 1	Texto	Primer apellido del usuario
Apellido 2	Texto	Segundo apellido del usuario
Email	Texto	Email del usuario

Al pulsar el botón de añadir usuario, el usuario desaparecerá de la tabla de “Usuarios buscados” y se incluirá en la tabla de “Usuarios asociados”, que tendrá los siguientes campos:

Nombre	Tipo	Descripción
Nombre de usuario	Texto	Nombre con el que el usuario acceder a la aplicación
Nombre	Texto	Nombre del usuario
Apellido 1	Texto	Primer apellido del usuario
Apellido 2	Texto	Segundo apellido del usuario
Email	Texto	Email del usuario
Eliminar Usuario	Botón	Elimina el usuario del estudio

Es obligatorio incluir al menos un usuario en el estudio. En caso de que toda la información del estudio haya sido completada correctamente, al pulsar el botón crear se creará el nuevo estudio y se redirigirá al usuario a la pantalla de “Edición de Estudio”.

En la pantalla de creación existirá el botón “Volver” que llevará al usuario de vuelta a la pantalla en la que se encontrara anteriormente.



Edición de estudio

La pantalla y la funcionalidad serán idénticas a las descritas para la creación de nuevos estudios, con la diferencia de que los datos vendrán precargados con la información del estudio que va a editarse.

La principal diferencia radica en que no es necesario pulsar el botón guardar para que las modificaciones en los usuarios asociados sean efectivas. Es decir, en el momento en que se pulsa el botón añadir usuario en el listado de usuarios buscados y éste aparezca en la lista bajo el título “Usuarios Asociados”, el usuario ya ha sido añadido al estudio. Lo mismo ocurre con el caso de eliminar un usuario del listado de “Usuarios asociados”, en el momento en que no aparezca en la tabla de “Usuarios asociados”, ya no estará en el estudio.

No podrán ser eliminados todos los usuarios asociados a un estudio.

Gestión de secuencias

Se trata de la parte de la aplicación en la que se podrán gestionar las secuencias que han sido utilizadas para los análisis de la aplicación. Se presentará por pantalla un formulario de búsqueda en el que aparecerán los siguientes criterios:

Nombre	Descripción
Nombre	Nombre con la que ha sido guardada la secuencia. Podrán realizarse búsquedas aproximadas, mediante el uso del carácter %

Se mostrarán los resultados de la búsqueda en una tabla que contendrá la siguiente información:

Nombre	Descripción
Acciones	Aparecerán dos botones: - Eliminar. Permite la eliminación de la secuencia, siempre y cuando no exista ningún estudio activo que la esté utilizando
Nombre de usuario	Nombre de la secuencia
Descripción	Descripción de la secuencia
Estudios	Estudios que están utilizando la secuencia



Análisis de datos

Se trata de la pantalla principal sobre la que el usuario trabajará. Aparecerá en el menú superior de la pantalla, para que en caso de que el usuario se encuentre en otra pantalla, pueda volver al inicio.

En caso de que el usuario haya sido asociado a más de un estudio, se mostrará la pantalla de “Selección de estudio”, mientras que si el usuario sólo pertenece a un estudio, se pasará directamente a la pantalla de “Introducción de datos de búsqueda”.

A partir de la pantalla de “Introducción de datos de búsqueda” se mostrará bajo el menú superior el nombre del estudio para el que se está realizando el análisis. Además, aparecerá también en el menú superior la opción de acceder a los resultados del estudio seleccionado, para que puedan consultarse los análisis realizados previamente.

Selección de estudio

En éste caso, se mostrará por pantalla un listado de los estudios a los que el usuario ha sido asociado, con los siguientes campos:

Nombre	Descripción
Entrar en estudio	Botón que nos llevará a la página de “Introducción de datos de búsqueda” para continuar llevando a cabo análisis para éste estudio.
Nombre	Nombre del estudio
Fecha	Fecha en la que fue dado de alta
Descripción	Descripción del estudio
Finalizar estudio	Botón que posibilita al usuario dar por finalizado un estudio. Se solicitará confirmación mediante un pop-up con el mensaje “¿Está seguro de que desea dar por finalizar el estudio? Esta acción no es reversible y se perderán todos los resultados obtenidos.”. En caso de que confirme la acción, se eliminará el estudio así como todos los resultados asociados a él.



Resultados del estudio

Una vez seleccionado el estudio, el usuario podrá acceder a los resultados de todos los análisis realizados en dicho estudio, independientemente de quien haya sido el usuario que los ha hecho. Aparecerá un link en el menú superior desde el que se accederá a la pantalla de resultados, en la que se mostrará una tabla con todos los análisis realizados dentro de un mismo estudio.

Nombre	Tipo	Descripción
Seleccionar	Check	Check que permite seleccionar el resultado para comparación con otros resultados
Usuario	Texto	Nombre de usuario que ha realizado el análisis.
Identificador Secuencia	Texto	Identificador único de la secuencia nucleótidos
Ventana	Texto	Tamaño de la ventana seleccionada
Solapamiento	Texto	Tamaño del solapamiento seleccionado
Secuencia	Texto	Secuencia buscada en las ventanas
Fecha	Texto	Fecha en la que se realizó el estudio
Acciones	Botones	<p>Distintas acciones que se podrán ejecutar sobre el resultado:</p> <ul style="list-style-type: none"> - Ver resultado: Se muestra la pantalla de resumen, para ver en detalle el análisis. - Eliminar resultado: Se pide confirmación al usuario mediante un pop-up con el mensaje: “¿Está seguro de que desea eliminar el resultado? La acción no es reversible.”. En caso afirmativo, eliminará el resultado. - Reutilizar secuencia: Permite realizar un nuevo análisis sobre la secuencia seleccionada. Nos lleva directamente a la pantalla de selección de parámetros.

Introducción de datos de búsqueda

Se solicitará que introduzca la secuencia sobre la que podrán realizarse los análisis. Existen varias formas de introducir éstos datos, que se definen a continuación.

Búsqueda en EMBL-BANK

El usuario deberá introducir en el cuadro de texto el identificador único de la secuencia de nucleótidos.

Al pulsar “Cargar” se obtendrá de la base de datos EMBL-BANK la información solicitada, mediante servicio REST.

En caso de que la carga se haya realizado de forma correcta, se mostrará al usuario la pantalla de parámetros de búsqueda.

Selección de fichero

Debajo del formulario para la búsqueda en EMBL-BANK, se presenta un segundo formulario en el que se permitirá al usuario seleccionar un fichero existente en su propio ordenador y cargarlo en la aplicación.

Al seleccionar esta opción, se redirigirá al usuario a una nueva pantalla, en la que se encontrará el botón de “seleccionar fichero”, que mostrará un cuadro de diálogo en el que el usuario podrá navegar por las carpetas de su ordenador hasta seleccionar el fichero adecuado.

Al pulsar el botón “Cargar”, se comprobará que el formato del fichero es válido y si su contenido también lo es.

Si el contenido es válido, se mostrará el nombre del fichero en pantalla y se dará al usuario la posibilidad de continuar el análisis, pulsando el botón “Analizar” o de cancelar la operación, pulsando el botón “Cancelar”, que lo llevará de nuevo a la pantalla de introducción de datos de búsqueda.

En caso de que no sea válido, se informará mediante un mensaje, no permitiendo continuar con el análisis.

En caso de que continúe con el análisis, se mostrará al usuario la pantalla de parámetros de búsqueda

Introducir secuencia

Debajo del formulario para la selección de fichero, se mostrará una caja de texto cuyo tamaño no estará limitado, donde el usuario podrá pegar la secuencia de nucleótidos sobre la que desea realizar el análisis. Además, podrá indicar el nombre deseado para la secuencia, así como una breve descripción de la misma.

Al pulsar el botón “Cargar Secuencia”, se comprobará que el formato de la cadena es válido. En caso de no ser válido, se mostrará el siguiente mensaje por pantalla: “La secuencia de nucleótidos no es válida, por favor, revise los datos y vuelva a intentarlo”.

En caso de que la carga del fichero finalice correctamente, se mostrará al usuario la pantalla de



parámetros de búsqueda. En caso contrario, se mostrará un mensaje informando del error, no permitiendo continuar con el análisis.

Búsqueda en el sistema

La última opción disponible para la búsqueda de secuencias de análisis, es la de búsqueda en las secuencias existentes en el sistema.

El usuario deberá introducir en el campo “Nombre de secuencia” el nombre de la secuencia que desea buscar. En caso de que exista en el sistema una secuencia cuyo nombre coincida exactamente con la secuencia dada, se cargará automáticamente, pasando a la pantalla de parámetros de búsqueda.

Parámetros de búsqueda

Tras la correcta carga de datos, se presentará la pantalla de los parámetros de búsqueda, en la que se presentará o deberá completar la siguiente información.

Nombre	Tipo	Editable	Obligatorio	Descripción
Identificador	Texto	No	SI	Identificador único de la secuencia de nucleótidos
Descripción	Texto	SI	NO	Descripción de la secuencia de nucleótidos
Ventana	Numérico	SI	SI	Tamaño de ventana que se aplicará en el análisis. El tamaño de la ventana debe ser como mínimo 1 y como máximo el tamaño de la secuencia que se analiza.
Solapamiento	Numérico	SI	SI	Solapamiento que se aplicará en el análisis. El solapamiento nunca puede ser mayor que el tamaño de ventana seleccionado.
Secuencia de búsqueda	Texto	SI	SI	Secuencia de búsqueda de nucleótidos en la que estamos interesados. Sólo podrá contener los caracteres ATCG y N. Como mínimo debe tener un carácter y como máximo el mismo número de caracteres que el tamaño de la ventana elegida. También podrá incluirse el comodín “?”
Incluir N	Check	SI	NO	Indica si en la búsqueda se incluirá o no el carácter N, según lo que se describirá a continuación.



Para la secuencia de búsqueda deberá tenerse en cuenta lo siguiente:

- Check “Incluir N” no marcado:
 - o Si la secuencia de búsqueda no incluye el carácter “?”, se buscará la secuencia tal cual se ha introducido en el cuadro de texto.
 - o Si la secuencia de búsqueda incluye el carácter “?”, se buscarán todas las combinaciones de la secuencia con los nucleótidos A, C, T y G que sustituirán el “?”.
- Check “Incluir N” marcado:
 - o Si la secuencia de búsqueda no incluye el carácter “?”, se buscará la secuencia tal cual se ha introducido en el cuadro de texto.
 - o Si la secuencia de búsqueda incluye el carácter “?”, se buscarán todas las combinaciones de la secuencia de nucleótidos A, C, T, G y N que sustituirán la “?”.

En caso de que todos los datos hayan sido completados correctamente, se iniciará el análisis y al finalizar se mostrará al usuario la pantalla de Resumen de Resultados. El sistema dividirá la secuencia inicial según el tamaño de ventana indicado y teniendo en cuenta el desplazamiento. Sobre esto, buscará aquellas ventanas que contengan la secuencia de búsqueda indicada y hará recuento del número de veces que aparece en ellas.

Resumen de resultados

En la parte superior se mostrará a modo informativo el identificador de la secuencia así como la descripción de la misma, si la tuviera.

Además se indicará cual es la ventana, solapamiento y secuencia de búsqueda indicados.

Una vez finalizado el análisis seleccionado, se mostrará por pantalla el resumen de los resultados obtenidos.

Se mostrará una tabla en la que se incluirán el 20% de las ventanas con mayor número de ocurrencias, hasta un máximo de 100 resultados.

Se dividirán los resultados en conjuntos de 20 resultados, que serán mostrados en distintas tablas hasta un máximo de 5 tablas.

Cada tabla indicará el número de ventana, el número de ocurrencias y la página de resultados en la que se puede encontrar dicha ventana, dentro de los resultados totales, a los que puede accederse desde ésta pantalla. Al dejar el cursor del ratón encima de la celda de resultado, se mostrará el contenido de la ventana analizada.

Se incluirá igualmente un histograma en el que se representará el número de ocurrencias de las ventanas de la tabla, ordenando el número de ventanas de menos a mayor.

Podrá accederse a la reutilización del análisis, que se explicará más adelante.

Desde ésta pantalla podrá descargarse los resultados de la tabla en formato excel y la gráfica como PDF, PNG, JPG o SVG



Resultados Totales

Al igual que en la pantalla de resumen de resultados, en la parte superior se mostrará a modo informativo el identificador de la secuencia así como la descripción de la misma, si la tuviera.

También se indicará cual es la ventana, solapamiento y secuencia de búsqueda indicados.

En caso de que el análisis tenga más de 2000 resultados, se paginará incluyendo como máximo 2000 resultados por página.

Los resultados se dividirán en tablas de 20 resultados por tabla, hasta un máximo de 100 tablas por página.

Cada tabla indicará el número de ventana y número de ocurrencias de la secuencia buscada. Al dejar el cursor encima de la celda de resultado, se mostrará el contenido de la ventana analizada.

Debajo de la tabla de resultados, se incluirá un histograma en el que se mostrarán las ventanas y el número de ocurrencias de cada una de ellas.

Podrá descargarse la totalidad de los resultados, sin paginar en formato excel. Podrá descargarse el histograma de cada página como PDF, PNG, JPG o SVG.

Desde ésta pantalla, también podrá accederse a la pantalla de reutilización del análisis.

En caso de que el usuario no indique nada, los resultados se guardarán, pero existe el botón “Eliminar Resultado” que el usuario podrá utilizar en caso de que considere que los resultados obtenidos no son relevantes para el estudio y no deben ser tenidos en cuenta.

Reutilizar búsqueda

Se trata de una opción que podemos encontrar en la pantalla de resumen de resultados, en la de resultados totales y en la de resultados del estudio. Gracias a esta funcionalidad, se permite al usuario realizar un nuevo análisis sobre una secuencia que ya ha sido utilizada. En caso de seleccionarla, se presentará al usuario la pantalla de parámetros de búsqueda, en la que deberá introducir los nuevos parámetros de análisis.

2.5.4 Especificación de los casos de uso y diagramas de actividad

A continuación van a describirse los distintos casos de uso, acompañándolos de los diagramas de actividad correspondientes:

Seleccionar Estudio

Nombre	Seleccionar estudio
Actores	Usuario
Precondiciones	El usuario debe haber sido asociado a más de un estudio para poder seleccionar sobre cual quiere trabajar.
Postcondiciones	El usuario trabajará sobre el estudio seleccionado, hasta que vuelva a ejecutarse este caso de uso y seleccione otro.
Descripción	El caso de uso de “seleccionar estudio” sólo estará disponible para aquellos usuarios que tengan asociado más de un estudio. Permitirá al usuario seleccionar sobre qué estudio quiere trabajar. En caso de que el usuario sólo pertenezca a un sólo estudio, será la aplicación la que se encargue de seleccionar el estudio sobre el que trabajará el usuario, no dando opción a cambio.

Diagrama de actividad del caso “Seleccionar Estudio”



**Realizar Análisis**

Nombre	Realizar Análisis
Actores	Usuario
Precondiciones	Debe haberse ejecutado previamente el caso de uso “Selección de estudio”
Descripción	El caso de uso “realizar análisis” es el encargado del análisis de las secuencias según los parámetros introducidos, para mostrar el resumen y los resultados.

Los siguientes casos de uso están incluidos en el caso de uso “Realizar Análisis”

Nombre	Seleccionar Secuencia
Actores	Usuario
Descripción	Permite seleccionar el origen de la secuencia de nucleótidos analizada, obteniéndola gracias a los servicios de EMBL-BANK, subiendo un fichero desde el ordenador del usuario o introduciéndolo como texto libre.

Nombre	Servicio EMBL-BANK
Actores	Usuario
Descripción	Permite obtener un fichero tipo FASTA desde EMBL-BANK con la secuencia que se desea analizar gracias al identificador único de la secuencia.

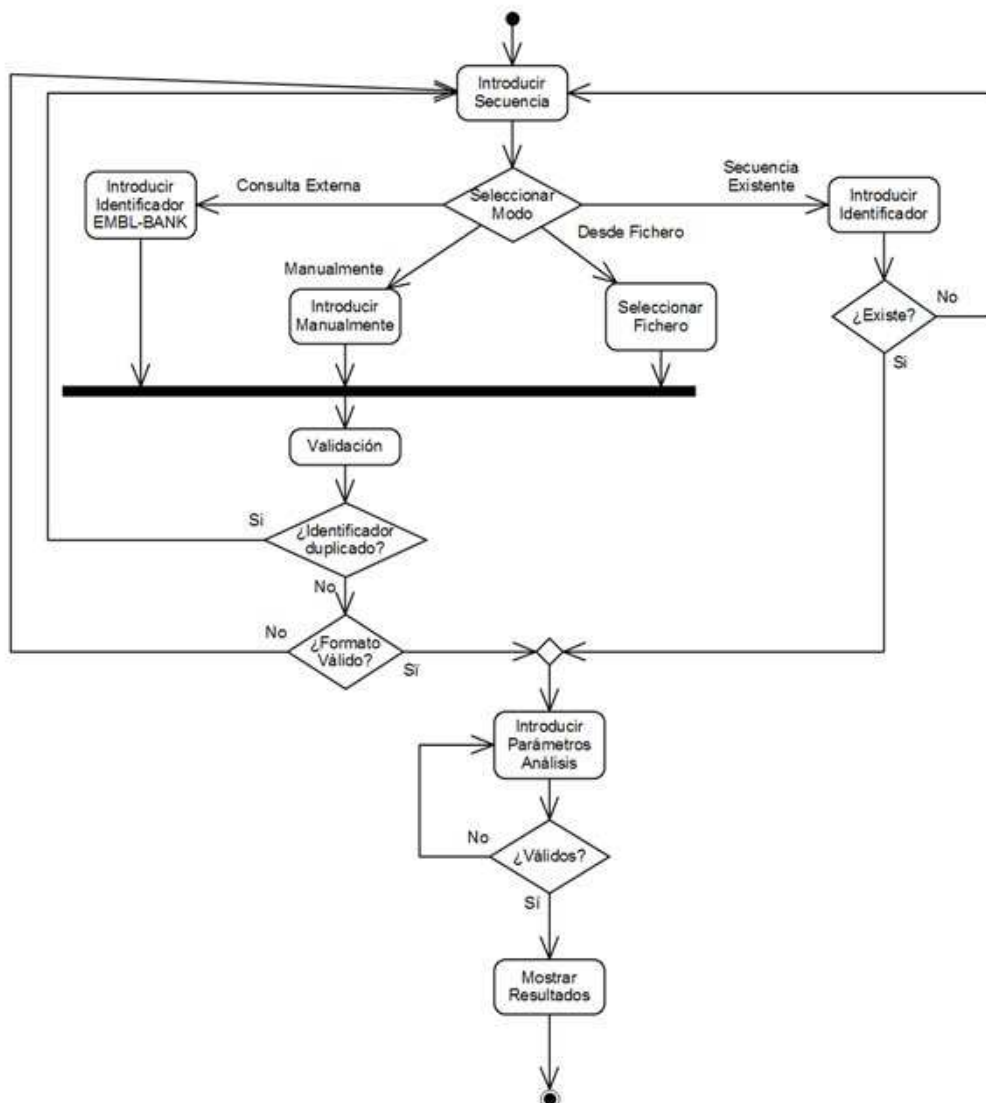
Nombre	Carga de Fichero
Actores	Usuario
Descripción	Permite cargar el contenido de un fichero a la aplicación, de tipo FASTA o TXT. Se validará su contenido para realizar el análisis sobre su contenido

Nombre	Carga Secuencia Texto Libre
Actores	Usuario
Descripción	El usuario puede introducir una secuencia corta, con su nombre y descripción.

Nombre	Cargar secuencia existente
Actores	Usuario
Descripción	Permite cargar una secuencia existente gracias a su identificador.

Nombre	Introducir Parámetros
Actores	Usuario
Descripción	Introducción de los parámetros que se aplicaran al análisis.

Diagrama de actividad del caso “Realizar Análisis”



**Reutilizar Secuencia**

Nombre	Reutilizar Secuencia
Actores	Usuario
Precondiciones	Debe haberse ejecutado previamente el caso de uso “Realizar Análisis”
Descripción	El caso de uso de “reutilizar secuencia” es el que permite realizar un nuevo análisis sobre la misma secuencia que el último que acaba de realizarse.

Diagrama de actividad del caso “Reutilizar Secuencia”





Búsqueda de Resultados

Nombre	Búsqueda de resultados
Actores	Usuario
Precondición	Previo a este caso de uso es necesario haber ejecutado el caso de uso de “Seleccionar Estudio” y haber ejecutado alguna vez el caso de uso “Realizar Análisis”
Descripción	El caso de uso de “búsqueda de resultados” permite al usuario buscar resultados de análisis realizados previamente.

Diagrama de actividad del caso “Búsqueda de Resultados”



**Borrar Resultados**

Nombre	Borrar Resultado
Actores	Usuario
Precondición	Para poder borrar un resultado, previamente ha debido ejecutarse el caso de uso de “Búsqueda de Resultados”.
Descripción	El caso de uso de “borrar resultado” es el encargado de borrar los resultados de algún análisis ya realizado.

Diagrama de actividad del caso “Borrar Resultado”

**Gestión de Contraseña**

Nombre	Gestión Contraseña
Actores	Usuario
Descripción	El caso de uso de “gestión contraseña” es el que permite a los usuarios modificar su contraseña de acceso a la aplicación.



Diagrama de actividad del caso “Gestión Contraseña”

**Gestión de Usuarios**

Nombre	Gestión de usuarios
Actores	Administrador
Descripción	Este caso de uso se encarga de gestionar los usuarios que tienen acceso a la aplicación. Se da la opción de realizar búsquedas de los usuarios, crear nuevos usuarios, editar los ya existentes o eliminarlos.

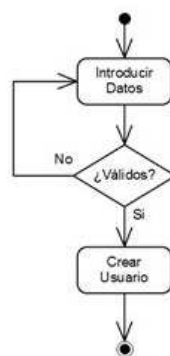
Nombre	Búsqueda de usuarios
Actores	Administrador
Descripción	Este caso de uso facilita la búsqueda de usuarios dados de alta en la aplicación, mediante los parámetros de filtro definidos

Diagrama de actividad del caso “Búsqueda de Usuarios”



Nombre	Crear Usuario
Actores	Administrador
Descripción	Este caso de uso permite crear un nuevo usuario en la aplicación.

Diagrama de actividad del caso “Crear Usuario”



Nombre	Editar Usuario
Actores	Administrador
Precondiciones	Para poder ejecutar este caso de uso, previamente es necesario ejecutar el caso de uso de “búsqueda de usuarios”
Descripción	Para poder ejecutar el caso de uso de “editar usuario”, es necesario previamente realizar una búsqueda de usuarios, sobre cuyo resultado se seleccionará el usuario que se desea modificar.

Diagrama de actividad del caso “Editar Usuario”





Nombre	Gestionar Contraseña usuario
Actores	Administrador
Precondiciones	Para poder ejecutar este caso de uso, previamente es necesario ejecutar el caso de uso de “búsqueda de usuarios”
Descripción	Para poder ejecutar el caso de uso de “gestionar contraseña usuario”, es necesario previamente realizar una búsqueda de usuarios, sobre cuyo resultado se seleccionará al usuario al que se desea reiniciar la contraseña..

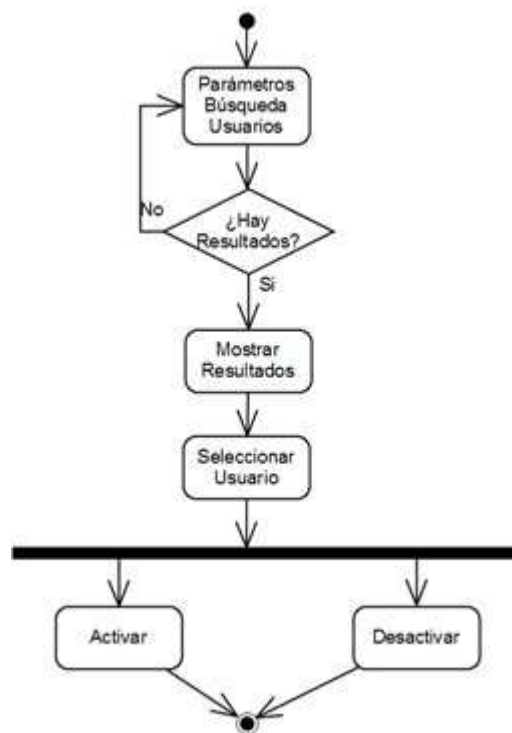
Diagrama de actividad del caso “Gestionar Contraseña Usuario”





Nombre	Desactivar/Activar Usuario
Actores	Administrador
Precondiciones	Para poder ejecutar este caso de uso, previamente es necesario ejecutar el caso de uso de “búsqueda de usuarios”
Descripción	Para poder ejecutar el caso de uso de “activar/desactivar usuario” es necesario previamente realizar una búsqueda de usuarios, sobre cuyo resultado se seleccionará al usuario que se desea activar o desactivar.

Diagrama de actividad del caso “Desactivar/Activar Usuario”





Gestión de Estudios

Nombre	Gestión de estudios
Actores	Administrador
Descripción	Este caso de uso se encarga de gestionar los estudios existentes en la aplicación. Se da la opción de realizar búsquedas de los estudios, crear nuevos estudios, editar los ya existentes o eliminarlos, así como de añadir o eliminar usuarios asociados a los mismo y eliminar análisis realizados dentro de un estudio

Nombre	Búsqueda de estudios
Actores	Administrador
Descripción	Este caso de uso facilita la búsqueda de estudios dados de alta en la aplicación, mediante los parámetros de filtro definidos

Diagrama de actividad del caso “Búsqueda de estudios”





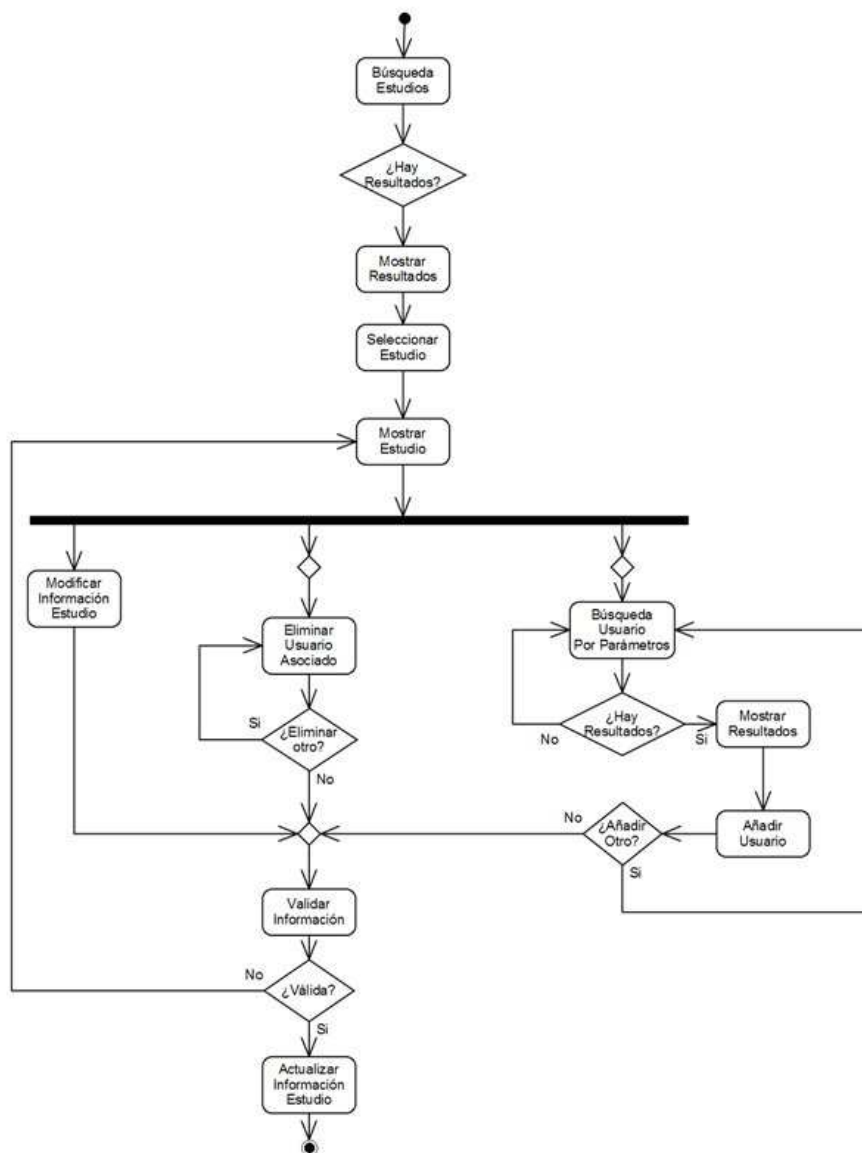
Nombre	Crear Estudio
Actores	Administrador
Descripción	Este caso de uso permite crear un nuevo estudio en la aplicación

Diagrama de actividad del caso “Crear Estudio”



Nombre	Editar Estudio
Actores	Administrador
Precondiciones	Para poder ejecutar este caso de uso, previamente es necesario ejecutar el caso de uso de “búsqueda de estudios”
Descripción	Para poder ejecutar el caso de uso de “editar estudio”, es necesario previamente realizar una búsqueda de estudios, sobre cuyo resultado se seleccionará el usuario que se desea modificar.

Diagrama de actividad del caso “Editar Estudio”





Nombre	Borrar Estudio
Actores	Administrador
Precondiciones	Para poder ejecutar este caso de uso, previamente es necesario ejecutar el caso de uso de “búsqueda de estudios”
Descripción	Para poder ejecutar el caso de uso de “borrar estudio” es necesario previamente realizar una búsqueda de estudios, sobre cuyo resultado se seleccionará el estudio que se desea eliminar.

Diagrama de actividad del caso “Borrar Estudio”



**Gestión de Secuencias**

Nombre	Gestión de Secuencias
Actores	Administrador
Descripción	Este caso de uso se encarga de gestionar las secuencias existentes en la aplicación, porque ya han sido utilizadas en algún estudio. Se da opción de realizar búsquedas y eliminar secuencias.

Nombre	Búsqueda de secuencias
Actores	Administrador
Descripción	Este caso de uso facilita la búsqueda de secuencias utilizadas para los análisis de la aplicación.

Diagrama de actividad del caso “Búsqueda de secuencias”





Nombre	Borrar Secuencia
Actores	Administrador
Precondiciones	Para poder ejecutar este caso de uso, previamente es necesario ejecutar el caso de uso de “búsqueda de secuencias”
Descripción	Para poder ejecutar el caso de uso de “borrar secuencia” es necesario previamente realizar una búsqueda de secuencias, sobre cuyo resultado se seleccionará el estudio que se desea eliminar. Sólo podrá eliminarse la secuencia si no está siendo utilizada en ningún estudio activo.

Diagrama de actividad del caso “Borrar secuencia”



3. Plan de pruebas

Cualquier proyecto de desarrollo requiere de la realización e implementación de pruebas, que deben ser capaces de cubrir diferentes propósitos, dependiendo si se trata de pruebas unitarias, de integración, funcionales, de regresión, de estrés...

El éxito de las pruebas dependerá del diseño de las mismas así como de una adecuada elección de las herramientas utilizadas.

Es importante tener en cuenta que la calidad de las pruebas, no sólo nos ayudará a detectar posibles errores de desarrollo, si no que influye directamente en la calidad final del producto y formará parte fundamental de su evolución y mantenimiento.

Disponer de un conjunto de pruebas adecuado, permitirá reducir el riesgo de errores en la evolución de los proyectos, ya sea por la implementación de nuevas funcionalidades o por mejora en las ya existentes.

3.1 Pruebas Unitarias

Las pruebas unitarias son las que realizamos individualmente en cada una de las partes estructurales de la aplicación.

Se ha diseñado un plan de pruebas unitarias para cada módulo funcional de la aplicación, centrándonos en los posibles errores que pueden producirse, debidos, por ejemplo, a campos no informados o informados incorrectamente e los distintos formularios de la aplicación.

En nuestro caso, las pruebas unitarias se han efectuado manualmente.

3.2 Pruebas Funcionales

Las pruebas funcionales están enfocadas a garantizar que los requerimientos funcionales de la aplicación hayan sido cubiertos. Este tipo de pruebas se crean desde el punto de vista del usuario, para confirmar que la aplicación realmente hace lo que se espera. Para poder realizar estas pruebas es necesario un documento funcional exhaustivo en el que se refleje fielmente tanto la funcionalidad como los mensajes y errores obtenidos en los casos que corresponda. Con un buen conjunto de pruebas funcionales, podremos cubrir además gran parte de las pruebas unitarias necesarias.

Las pruebas de regresión tienen como objetivo garantizar que ante cualquier modificación, por mantenimiento o inclusión de nuevas funcionalidades, no se vean afectados el resto de módulos de la aplicación.

Ambos tipos de pruebas pueden ejecutarse de forma manual o pueden ser automatizada, como ha sido nuestro caso, utilizando la herramienta Selenium^[8]. Al hacerlo así debemos tener en cuenta:

- Si hubiera un cambio en los requerimientos funcionales de la aplicación, será necesario revisar y modificar las pruebas para adaptarlas a las nuevas necesidades, de forma que las

pruebas deberán crecer a la par que el proyecto.

- Al generar una nueva versión del proyecto, se debe garantizar que las pruebas diseñadas se ejecutan de forma exitosa en su totalidad (pruebas de regresión).

Selenium es un conjunto de utilidades que facilita la tarea de crear juegos de pruebas para aplicaciones web. Permite grabar, editar y depurar casos de prueba, que posteriormente podrán ser ejecutados de forma automática e iterativa. Entre las principales características de Selenium destacamos:

- Facilidad de registro y ejecución de los test.
- Referencia a objetos DOM en base al ID, nombre o a través de XPath.
- Auto-completado para todos los comandos.
- Las acciones pueden ser ejecutadas paso a paso.
- Herramientas de depuración y puntos de ruptura.
- Los test pueden ser almacenados en diferentes formatos.

De esta forma, hemos diseñado un conjunto de pruebas que puede ser utilizado como pruebas funcionales de la aplicación durante el desarrollo y adaptación de la misma y como pruebas de regresión que deberán ser ejecutadas en el caso de futuras modificaciones en el proyecto.

3.3 Pruebas de Rendimiento

La aplicación realizada en este proyecto, inicialmente no está pensada para su uso masivo por un gran número de usuarios, pero debemos estar preparados para que su uso pueda generalizarse y no vernos desbordados por ello. Por lo tanto, tras finalizar el desarrollo vamos a realizar unas pruebas de carga utilizando la herramienta Jmeter^[9], simulando el uso de la aplicación por parte de varios usuarios concurrentes.

Este tipo de análisis del rendimiento de la aplicación se conoce comúnmente como *profiling*^[10]. Básicamente es una investigación sobre el comportamiento de la aplicación, utilizando la información obtenida a partir de un análisis dinámico. Se trata, en resumen, de ejecutar la aplicación y observar su comportamiento.

Se deben ejecutar los suficientes casos de prueba como para producir algún comportamiento interesante y para que el estudio resulte efectivo.

Jmeter es un proyecto de apache para analizar y medir el desempeño de una variedad de servicios y que está muy orientado al análisis de aplicaciones web.

Aunque puede ser utilizado también en pruebas unitarias para conexiones de bases de datos, servicios rest, servicios web, etc... nosotros vamos a utilizarlo únicamente para el caso de las pruebas de carga, ya que creemos que es la herramienta que más se ajusta a nuestras necesidades. Está considerado como una herramienta de generación de carga, aunque las posibilidades que ofrece este proyecto son bastante más amplias.

Para poder realizar un análisis de carga, deben definirse los casos que se desean probar o *testplan*, generando un fichero .jmx que posteriormente podrá ser ejecutado desde Jmeter.

Existen dos formas para generar estos testplan, la básica es la manual, en la que se crea cada componente, generando el árbol de ejecución y configurando el componente según las necesidades, y la automatizada, utilizando una herramienta que capture las peticiones HTTP hechas por el navegador mientras se utiliza la aplicación. Con estas peticiones, se genera un componente en el árbol de ejecución que posteriormente podrá ser ejecutado desde Jmeter. Aunque esta segunda estrategia no genera al completo el testplan, ya que posteriormente habrá que revisarlos y ajustarlos, si deja una estructura de las llamadas realizadas facilitando su creación. Nosotros vamos a seguir esta segunda estrategia utilizando BadBoy^[11]. Se trata de una poderosa herramienta diseñada para ayudar al desarrollo y las pruebas sobre el mismo, haciendo que éstas últimas sean más sencillas.

Existen, principalmente dos tipos de pruebas de rendimiento a tener en cuenta:

Las pruebas de carga, en las que se observa el comportamiento de la aplicación bajo una cantidad de peticiones esperada, en la que se realiza un número determinado de usuarios realizan un número concreto de peticiones durante el tiempo de la prueba. Su objetivo es mostrar los tiempos de respuesta de todas las transacciones importantes.

Las pruebas de estrés, cuyo objetivo es "romper" la aplicación. La idea es aumentar el número de usuarios y de peticiones realizadas hasta llegar a un punto en que la aplicación no lo soporte. Este tipo de pruebas se hace para conocer la solidez de la aplicación y determinar cuáles son sus límites en caso de un uso intensivo de la misma.

Vamos a realizar tres testplan, para poder analizar el rendimiento de los casos de los distintos tipos de entrada de datos a la aplicación. Por lo tanto, un test plan incluirá el estudio de secuencias obtenidas desde EMBL-BANK y otro el estudio de secuencias obtenidas desde fichero. También se realizará un análisis para el caso de secuencias introducidas manualmente en la aplicación.

Todas las secuencias utilizadas van a ser de tipo medio o alto y los parámetros definidos serán ventanas medias/altas, con desplazamientos cortos y secuencias de búsqueda pequeñas.

En primer lugar, vamos a analizar el caso base, en el que tendremos 5 usuarios concurrentes, que entrarán en la aplicación en una rampa de tiempo para la entrada de cada usuario será de 1 segundo. Cada usuario ejecutará 10 peticiones.

En segundo lugar, vamos a ejecutar el caso de carga de usuarios, en el que se eleva el número de usuarios que entrará en la aplicación para la ejecución de la prueba. Elevamos el número de usuarios a 50, con una rampa de entrada de 2 segundos, cada usuario ejecutará 15 peticiones.

Por último, vamos a ejecutar el caso de carga de peticiones, en el que se eleva el número de peticiones de cada usuario que hay en la aplicación. El número de usuarios será 25, con una rampa de entrada de 2 segundos y cada uno ejecutará 75 peticiones.

Por último, se va a realizar un análisis utilizando distintos parámetros para ventana, desplazamiento y secuencia de búsqueda, sobre una secuencia considerada como muy larga, para evaluar los tiempos de análisis, sin tener en cuenta la forma de cargar la secuencia.

4. Gestión del proyecto

A continuación se detalla la planificación del proyecto, así como la metodología seguida para el desarrollo del mismo.

Este proyecto nace con la necesidad de compaginarse con otros trabajos por parte del equipo de desarrollo, por lo que se ha elegido una metodología ágil, que permitirá al equipo un ajuste de tiempo y corrección de necesidades a lo largo del desarrollo del sistema, ya que se plantea la opción de que el usuario pueda revisar el proyecto y concretar durante el propio desarrollo las necesidades del mismo, siempre dentro de unos mínimos que garanticen el avance y la finalización del desarrollo en el plazo de tres meses como máximo.

4.1 Metodologías ágiles. Introducción a SCRUM.

Se conoce como metodología ágil^[12] al conjunto de métodos de ingeniería del software que basan el desarrollo del proyecto en un esquema iterativo e incremental. Gracias a este tipo de metodologías se permite que los requisitos y las soluciones vayan evolucionando junto con el desarrollo, gracias a la colaboración de los distintos grupos que integren el equipo de desarrollo del proyecto.

Las metodologías ágiles tienen su punto fuerte en la comunicación directa entre las distintas partes, más que en la documentación, siempre teniendo en cuenta que ésta es imprescindible en según qué partes del desarrollo.

Existen múltiples metodologías de éste tipo, pero para el desarrollo de éste proyecto se ha elegido SCRUM^[13]. Se trata de una forma de trabajo que surgió en empresas de productos tecnológicos, especialmente apropiada para proyectos con requisitos inestables y que requieren de rapidez y flexibilidad. Estos tres puntos suelen aparecer siempre en el desarrollo de prácticamente cualquier aplicación.

La metodología SCRUM separa el desarrollo del proyecto en distintas iteraciones, de ahora en adelante SPRINTS. Se define como SPRINT el periodo en el cual se lleva a cabo el trabajo real del proyecto. Su duración debe oscilar entre 2 y 4 semanas, adaptándolo siempre a la experiencia y ritmo del equipo así como del alcance del propio proyecto. Al final de cada SPRINT se debe obtener un producto potencialmente entregable, es decir, que realmente sea funcional.

Por lo tanto, y a modo de resumen, SCRUM permite:

- Adoptar una estrategia de desarrollo incremental, en vez de una planificación cerrada del producto.
- Basar la calidad del resultado en el conocimiento del equipo.
- Solapar distintas fases del desarrollo, en vez de realizar una tras otra e un ciclo secuencial o de cascada.
- Gestionar las expectativas del proyecto, con resultados anticipados, flexibilidad y adaptación.
- Reducir los riesgos, lo que hará aumentar la productividad y la calidad del producto.

- Alinear las necesidades del cliente con la información manejada por el equipo.

La metodología SCRUM define los siguientes roles que deben estar presentes durante el desarrollo del proyecto

- Dueño del producto, más conocido por su nombre inglés *Product Owner*. Es la representación del cliente y verifica que se trabaja de forma adecuada desde la perspectiva del negocio.
- Facilitador, más conocido por su nombre inglés *ScrumMaster*. Su trabajo consiste en eliminar los obstáculos o impedimentos que pueden dificultar el alcanzar el objetivo del SPRINT. No se trata de un líder, sino de una burbuja que aísla al equipo de posibles distracciones.
- El equipo: Tiene la responsabilidad de entregar el producto. Deben ser un equipo multidisciplinar, con habilidades transversales para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación...)
- Roles auxiliares. No tiene un rol formal y no se suelen involucrar en el proceso. Pueden ser los clientes, proveedores, vendedores o los Administradores o *Managers*.

Para el buen funcionamiento de la metodología SCRUM se definen una serie de reuniones, no todas de carácter obligatorio, pero sí recomendables en cualquier caso.

- Reunión diaria o DAILY. Cada día a la misma hora, en el mismo sitio, el equipo se reúne para ver el estado del proyecto. Cada uno de los integrantes debe responder a las preguntas:
 - o ¿Qué has hecho desde la DAILY de ayer?
 - o ¿Qué vas a hacer hasta la DAILY de mañana?
 - o ¿Ha surgido algún impedimento que haya impedido alcanzar el objetivo?
- Scrum de Scrum. En caso de tratarse de varios equipos trabajando sobre un mismo proyecto, se debe realizar una reunión para discutir posibles solapamientos en el desarrollo por parte de los distintos grupos. Se deben responder, por parte de cada equipo, a las siguientes preguntas:
 - o ¿Qué ha hecho el equipo desde la última reunión?
 - o ¿Qué hará el equipo hasta la próxima reunión?
 - o ¿Ha habido algún problema que demore al equipo?
 - o ¿Vas a cruzarte en el camino de otro equipo?
- Reunión de planificación del SPRINT

Al inicio de cada SPRINT, tiene lugar una reunión de planificación en la se define el trabajo que se hará, preparando un resumen de las tareas en el que se detalla el tiempo que llevará hacer el trabajo. Este resumen de tareas se conoce como Sprint Backlog.

- Reunión de revisión del SPRINT

Se trata de una reunión en la que se revisa al final del SPRINT que trabajo no fue completado. De igual forma, se presenta el trabajo completado.

- Retrospectiva del SPRINT:

Es otra reunión que tiene lugar al finalizar el SPRINT, en la que el equipo deja sus impresiones sobre el ciclo que acaba de terminar, con el propósito de mejorar o no cometer los mismos errores en el siguiente.

Todo esto no es un más que un resumen a grandes rasgos de lo que ésta metodología ofrece, aunque hay mucho más detrás de ella.

Es difícil aplicar al 100% cualquier metodología ágil a un proyecto, ya que cada desarrollo es un mundo y tiene distintos requerimientos. Es tan importante ser capaz de aplicar una metodología ágil como ser capaz de adaptarla para sacarle el máximo provecho en cada situación.

Igualmente, cabe destacar que se hace especial hincapié en la flexibilidad respecto a los requisitos y alcance del proyecto que aporta SCRUM. Conviene igualmente dejar constancia de que ésta flexibilidad debe ser controlada, ya que incluir ciertos aspectos o mejoras al proyecto no quiere decir que se modifiquen por completo todas las necesidades y funcionalidades de la aplicación durante el desarrollo. La idea siempre es aportar valor al proyecto, dentro de las posibilidades del desarrollo, intentado siempre que los objetivos generales del proyecto no se modifiquen. No se debe en ningún caso, permitir que se dé la vuelta al proyecto cuando ya se ha iniciado su desarrollo. Para ello, por lo tanto, es necesaria una buena ingeniería de requisitos.

4.2 Adaptación de SCRUM al caso concreto

En éste proyecto partimos de una planificación inicial para el desarrollo, que ha sido presentada anteriormente. No se trata de una planificación cerrada al 100%, ya que en nuestro caso hay una necesidad imperiosa de flexibilidad para el equipo, por la necesidad de compatibilizar el desarrollo del proyecto con otros trabajos. Es por ésta razón por la que se ha elegido como metodología ágil SCRUM. Además, al tratarse en cierto modo de un pequeño proyecto de investigación, se abre así la posibilidad de incluir en alguno de los SPRINTS, alguna nueva funcionalidad o mejora sobre lo inicialmente incluido en el alcance de la aplicación. Eso sí, no perdiendo de vista el foco principal del proyecto así como el alcance y las necesidades del mismo.

Se define que los SPRINTS serán de dos semanas, tiempo estimado como suficiente para producir un entregable del proyecto. No se puede establecer un número de horas fijo para cada día de desarrollo, por lo que en función de cada semana se establecen las horas mínimas para cada día que el equipo deberá trabajar, que será de 4 horas por jornada.

Resulta evidente que el equipo de desarrollo de éste proyecto no está formado por tantas personas como para asignar a cada una de ellas uno de los roles detallados anteriormente en la definición de SCRUM, por lo que será necesaria una adaptación de la metodología que pueda aportar valor durante la evolución del proyecto.

El grupo de trabajo de éste proyecto deberá regirse por los siguientes roles adaptados a la metodología SCRUM:

- Por el tutor del proyecto de fin de carrera, que actuará las veces como *Product Owner*.
- El *Scrum Master*, será en unas ocasiones el tutor del proyecto y en otras ocasiones el propio alumno.
- El equipo de desarrollo, que en éste caso estará compuesto por un sólo desarrollador, que deberá llevar a cabo las labores de definición, documentación, análisis y desarrollo del proyecto.
- Por otro lado, al no haber más de un equipo de desarrollo, se fijan como imprescindibles las siguientes reuniones:
- Definición de Sprint. Definiendo al inicio de cada Sprint el alcance del mismo.
- *Daily*. Para poder forzar el trabajo diario del desarrollador, ya que se encuentra trabajando en otros proyectos, se tendrá a diario una *daily* en la que se revisará la evolución del mismo, para ver el avance real del proyecto. En caso de encontrar impedimentos, relacionados con el proyecto y que retrasen la entrega del SPRINT, se tendrán en cuenta y se dejará constancia de ello en éste proyecto. Al tratarse de una reunión en la que sólo estará el desarrollador, se creará un “diario” de proyecto en el que se incluyan los avances, impedimentos... Que podrá ser revisado una vez finalizado cada SPRINT o el proyecto.
- Revisión de SPRINT. Una vez finalizado el SPRINT se revisará el resultado final, para constatar que es el esperado.
- Retrospectiva. Tras la revisión del SPRINT, se enumerarán los posibles impedimentos que hubo durante el desarrollo y las posibles mejoras a aportar tanto para los siguientes SPRINT como otros proyectos.

4.3 Sprint

A continuación van a definirse y describirse los *sprint* y los resultados obtenidos en los mismos.

4.4.1 Sprint 1

Definición de los objetivos del SPRINT.

El objetivo del primer *sprint* es configurar el entorno de trabajo para el desarrollo de la aplicación, integrando las distintas librerías y *frameworks* que van a utilizarse en el proyecto:

- Maven
- Spring MVC
- Spring Security
- Hibernate y la conexión a Base de datos MySQL
- Tiles
- Junit

Será también objetivo de éste *sprint* dejar definido el aspecto visual de la aplicación así como completar la definición de los roles de usuario que podrán acceder a la aplicación.



Product Backlog

Estas son las tareas y los tiempos definidos para el *sprint*. La fecha de inicio del mismo fue el 20 de febrero y la de fin el 8 de marzo.

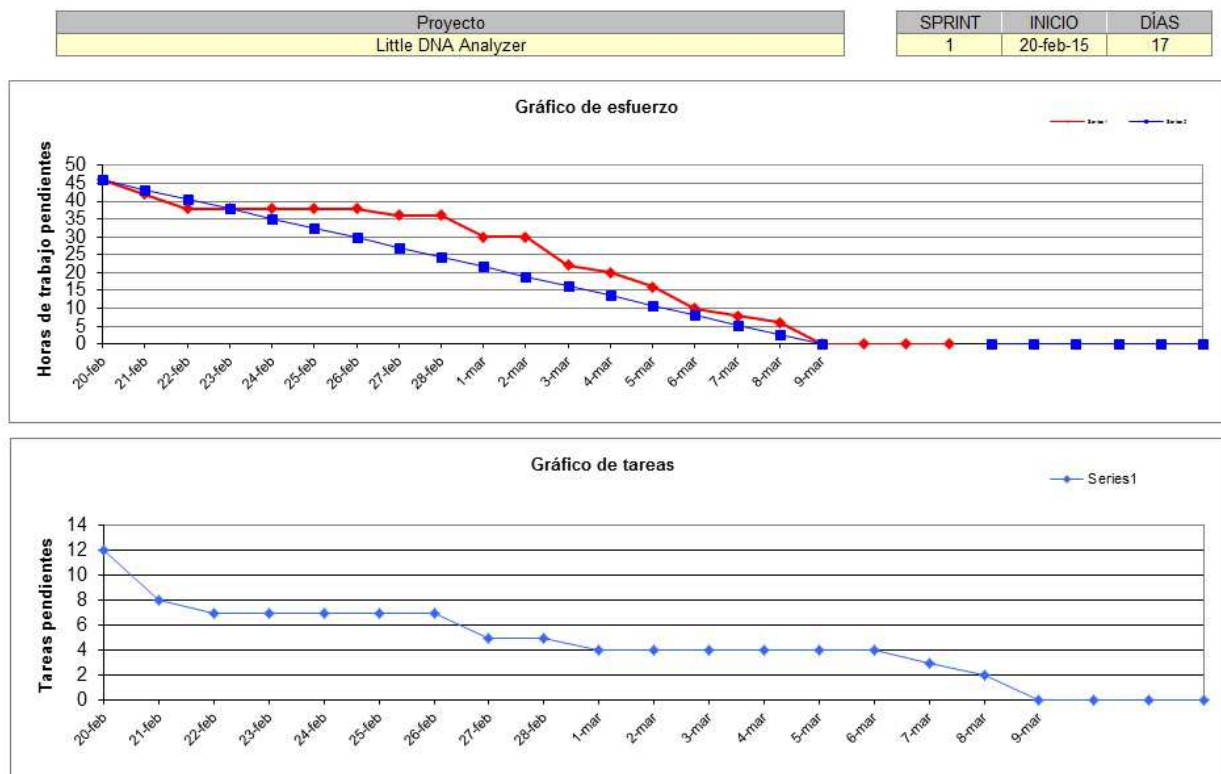
SPRINT		INICIO	DURACIÓN
1		20-feb-15	17

Tareas pendientes		20-feb	21-feb	22-feb	23-feb	24-feb	25-feb	26-feb	27-feb	28-feb	1-mar	2-mar	3-mar	4-mar	5-mar	6-mar	7-mar	8-mar	9-mar
Horas de trabajo pendientes		12	8	7	7	7	7	7	5	5	4	4	4	4	4	4	3	2	
ESFUERZO		46	42	38	38	38	38	38	36	36	30	30	22	20	16	10	8	6	

Backlog ID	PILA DEL SPRINT	Tipo	Estado	Responsab
	Instalación BBDD MySQL	Configuración	Terminada	Dolores García
	Creación esquema BBDD Gestión	Codificación	Terminada	Dolores García
	Instalación BBDD MongoDB	Configuración	Terminada	Dolores García
	QuickStart del proyecto	Configuración	Terminada	Dolores García
	Configuración del proyecto	Codificación	Terminada	Dolores García
	Integración JUNIT	Codificación	Terminada	Dolores García
	Login Spring Security	Codificación	Terminada	Dolores García
	Definición de usuario y roles de usuario	Codificación	Terminada	Dolores García
	Definición Servicios de autenticación	Codificación	Terminada	Dolores García
	Vista controlador login	Codificación	Terminada	Dolores García
	Diseño General de la aplicación	Codificación	Terminada	Dolores García
	Integración con Tiles	Codificación	Terminada	Dolores García

Gráfica de quemado.

En la gráfica que mostramos a continuación, vemos que al inicio del *sprint*, la línea roja va por debajo de la azul, lo que indica que el tiempo en el que se realizaron las tareas fue menor al esperado. Desde ahí hasta el 26 de febrero, la línea roja está por encima de la azul, lo que nos indica que no se ejecutaron las tareas al ritmo esperado, retrasando por tanto el desarrollo del proyecto.



A partir del día 26, la línea roja desciende más rápidamente que la azul, lo que demuestra que se hizo un esfuerzo para completar la totalidad de las tareas descritas en el tiempo previsto, ya que finalmente, se finaliza el proyecto en la fecha estimada de fin.

En la segunda gráfica se muestran las tareas pendientes y el ritmo en el que los desarrolladores las van asignando. Podemos observar dos periodos en los que el desarrollador permanece con la misma tarea durante varios días. La tarea que se desarrolla desde el 1 al 6 de marzo es la de “diseño general de la aplicación”, estimada en 24 horas. Se trata de una tarea larga en la que se estuvo trabajando en dicho periodo. Si observamos la gráfica de quemado, vemos que en este periodo sí desciende la línea roja, que indica que se han empleado horas en el trabajo del proyecto.

En cambio, el periodo que vemos entre el 22 y el 26 de febrero, nos indica que no se avanzó con el desarrollo de la aplicación, ya que no se completaron tareas. Si se observa la gráfica de quemado, observamos que este periodo coincide con el periodo en el que no se emplearon horas en el desarrollo del proyecto y que retrasaron el mismo.

Impedimentos

No se han encontrado impedimentos durante el desarrollo de éste *sprint*

Revisión del SPRINT

Se ha podido completar la totalidad de las tareas del *sprint*, sin encontrar impedimentos en el desarrollo de ésta primera fase.

Al finalizar se ha podido presentar al usuario el aspecto de la aplicación, dando el visto bueno al diseño elegido.

Cómo única funcionalidad tras este *sprint*, se ha presentado la autenticación de la aplicación.

Retrospectiva

El mayor problema que se presenta durante éste primer *sprint* es la imposibilidad de dedicarle al proyecto todo el tiempo deseado, alargándose ciertas tareas más de lo imprescindible.

4.4.2 Sprint 2

Definición de los objetivos del SPRINT.

No se ha introducido en éste *sprint* ninguna mejora que se encuentre fuera del alcance inicial del proyecto. El objetivo de éste *sprint* es el de completar la totalidad de la funcionalidad de gestión y administración de usuarios y estudios de la aplicación, lo que incluye:

- Gestión y Administración de usuarios
- Gestión y Administración de estudios
- Gestión de contraseña tanto para usuario como para administradores.



Product Backlog.

Estas son las tareas y los tiempos definidos para el *sprint* 2. La fecha de inicio del mismo fue el 9 de marzo y la de fin el 23 de marzo.

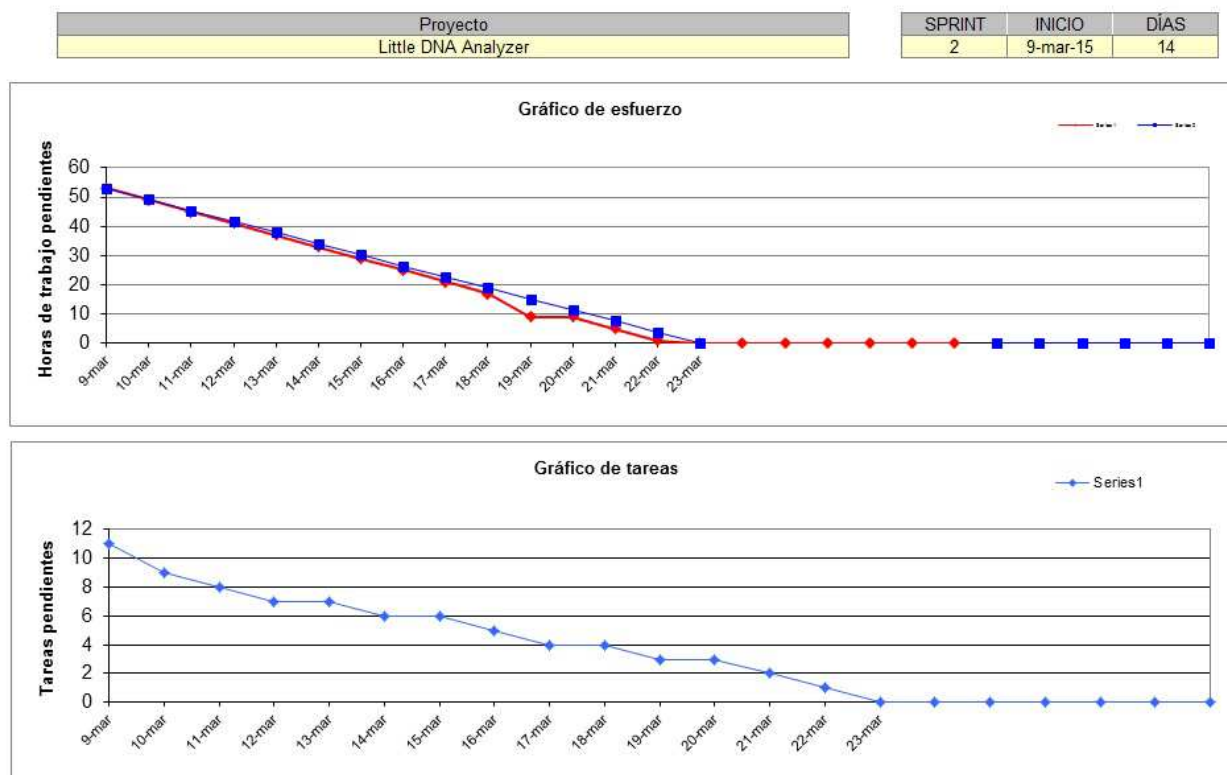
SPRINT		INICIO	DURACIÓN
2		9-mar-15	14

L	M	X	J	V	S	D	L	M	X	J	V	S	D	L
9-mar	10-mar	11-mar	12-mar	13-mar	14-mar	15-mar	16-mar	17-mar	18-mar	19-mar	20-mar	21-mar	22-mar	23-mar
Tareas pendientes	11	9	8	7	7	6	6	5	4	4	3	3	2	1
Horas de trabajo pendientes	53	49	45	41	37	33	29	25	21	17	9	9	5	1

Backlog ID	PILA DEL SPRINT	Tipo	Estado	Responsab	ESFUERZO											
13	Definición de servicios de usuario	Codificación	Terminada	Dolores García	2											
14	Definición de servicios de estudios	Codificación	Terminada	Dolores García	2											
15	Vista Controlador Búsqueda de usuarios	Codificación	Terminada	Dolores García	8	8	8	8	4							
16	Vista controlador Creación de usuarios	Codificación	Terminada	Dolores García	8	8	8	8	8	8						
17	Vista controlador Edición de usuarios	Codificación	Terminada	Dolores García	4	4	4	4	4	4	4					
18	Vista Controlador Cambio de contraseña Usuario	Codificación	Terminada	Dolores García	4	4										
19	Vista Controlador Cambio de contraseña Admin	Codificación	Terminada	Dolores García	4	4	4									
20	Vista controlador búsqueda de estudios	Codificación	Terminada	Dolores García	8	8	8	8	8	8	8	8	4			
21	Vista controlador Creación de estudios	Codificación	Terminada	Dolores García	8	8	8	8	8	8	8	8	4	4		
22	Vista controlador edición de estudios	Codificación	Terminada	Dolores García	4	4	4	4	4	4	4	4	4	4	4	
23	Vista Controlador Eliminar estudios	Codificación	Terminada	Dolores García	1	1	1	1	1	1	1	1	1	1	1	1

Gráfica de quemado

Esta es la gráfica de quemado del segundo *sprint*. La línea azul representa el quemado ideal y continuo de las tareas, mientras que el rojo representa el quemado real en el desarrollo del proyecto.



Puede observarse que el tiempo en que se han realizado las tareas ha sido inferior al estimado.

En la segunda gráfica puede analizarse el volumen de las tareas pendientes desde el inicio al final del *sprint*, en la que vemos que prácticamente en la totalidad de los casos, se ha completado una tarea por día de desarrollo.

Impedimentos

No se han encontrado impedimentos en el desarrollo de éste *sprint*.

Revisión del SPRINT

Al finalizar este *sprint* se han completado la totalidad de las funcionalidades definidas.

Se llevó a cabo una demostración del estado del desarrollo al usuario, en la que surgieron ciertas dudas y se plantearon mejoras. Se incorporaron las mejoras, lo que hizo necesario modificar las tareas inicialmente definidas en el siguiente *sprint*.

Se incluyó una modificación en las secuencias de búsqueda de la aplicación. Inicialmente, sólo se tendrían en cuenta los caracteres A, T, G y C en las búsquedas. Se incluye en éstos el carácter N. Igualmente se incluye la posibilidad de incluir un comodín en la secuencia de búsqueda (el carácter interrogante), que permitirá realizar búsquedas sustituyendo dicho carácter por todos los válidos de la secuencia de búsqueda.

Igualmente, se propone al usuario una modificación de la representación de los resultados. Inicialmente sólo iba a presentarse una página de resultados en la que se mostraría toda la información obtenida. Se propone mostrar inicialmente un resumen de los resultados más relevantes encontrados. Además de éste resumen, el usuario tendrá acceso a los resultados totales del estudio, que en vez de mostrarse en una sola pantalla, se paginarán de forma que sean más legibles.

Para poder incluir estas nuevas funcionalidades sin alterar la fecha de fin del proyecto, se considera oportuno eliminar la funcionalidad de comparación de dos estudios que se había definido inicialmente y dejarla como posible mejora futura, ya que requiere de un análisis exhaustivo para que el resultado de la comparativa tenga sentido.

Retrospectiva

Tras la finalización de éste *sprint* y la reunión con el usuario, podemos tomar como muy positivo la aplicación a éste proyecto de ésta metodología de trabajo, ya que nos ha permitido corregir un error de base que el usuario no tuvo en cuenta, llegando a un acuerdo a nivel de reducción de funcionalidades, ya que consideró el propio usuario que no era necesario actualmente la comparativa de resultados.



4.4.3 Sprint 3

Definición de los objetivos del SPRINT.

Dentro de este *sprint* se contempla inicialmente la siguiente funcionalidad:

- Lectura y validación de secuencias para el análisis desde ficheros y desde pantalla.
- Parámetros de búsqueda y validación
- Análisis de secuencias en función de los parámetros dados
- Presentación de resultados por pantalla, en tabla y con representación gráfica
- Comparación de resultados por pantalla, en tabla y con representación gráfica.

Se han introducido las siguientes mejoras que inicialmente no entraban dentro de la definición de la aplicación:

- Búsquedas con N y con ?
- Pantalla de resumen de resultados, descarga en Excel de su información

Para poder incluir estas funcionalidades, se ha eliminado la comparación de resultados la comparación de resultados, que era el último punto incluido en la definición del *sprint*.

Product Backlog

Estas son las tareas y los tiempos definidos para el *sprint* 3.

La fecha de inicio del mismo fue el 25 de marzo y la de fin el 9 de abril.

SPRINT	INICIO	DURACIÓN
3	25-mar-15	16

	X	J	V	S	D	L	M	X	J	V	S	D	L	M	X	J	V
25-mar																	
26-mar																	
27-mar																	
28-mar																	
29-mar																	
30-mar																	
31-mar																	
1-abr																	
2-abr																	
3-abr																	
4-abr																	
5-abr																	
6-abr																	
7-abr																	
8-abr																	
9-abr																	
10-abr																	

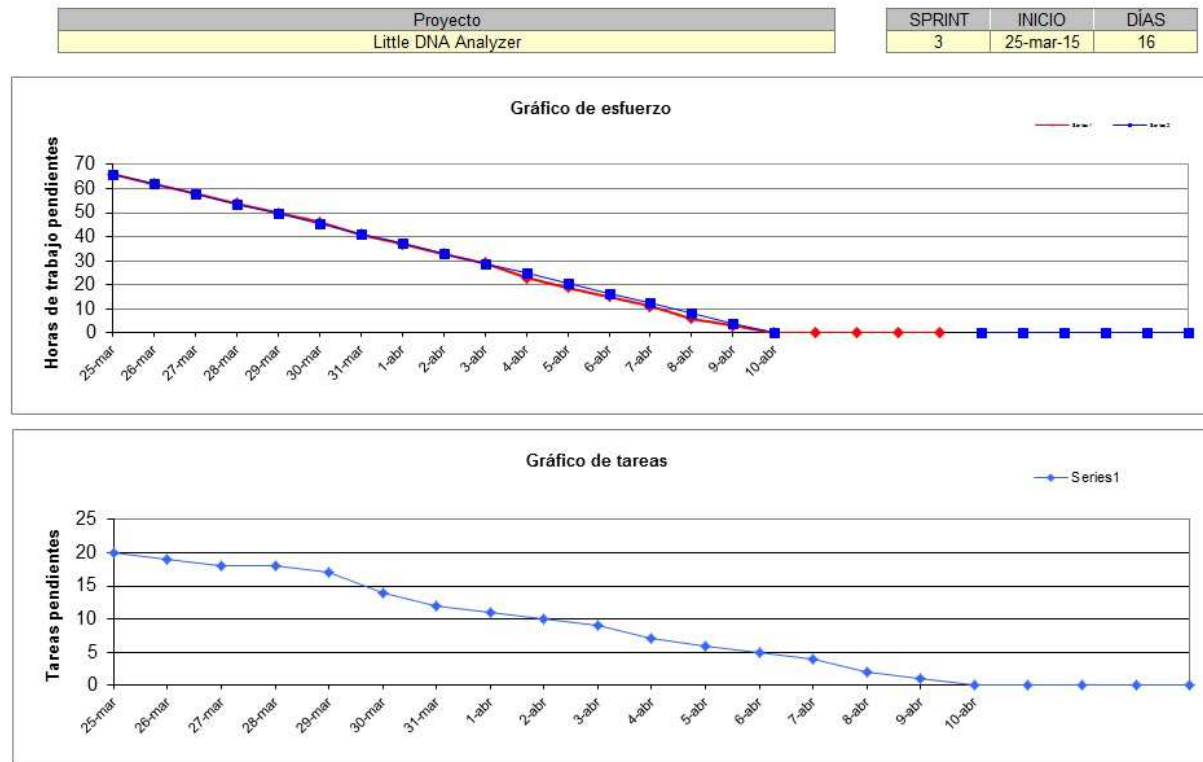
Tareas pendientes	20	19	18	18	17	14	12	11	10	9	7	6	5	4	2	1
Horas de trabajo pendientes	66	62	58	54	50	46	41	37	33	29	23	19	15	11	6	3

Backlog ID	PILA DEL SPRINT	Tipo	Estado	Responsab	ESFUERZO
24	Vista Controlador Selección de estudio	Codificación	Terminada	Dolores García	4
26	Vista Controlador Introducir Parametros	Codificación	Terminada	Dolores García	4
28	Lectura de secuencia desde fichero	Codificación	Terminada	Dolores García	8
30	Introducir secuencia por pantalla	Codificación	Terminada	Dolores García	2
31	Validación de secuencias	Codificación	Terminada	Dolores García	1
32	Limpieza de secuencias	Codificación	Terminada	Dolores García	1
34	Vista Controlador Parametros de búsqueda	Codificación	Terminada	Dolores García	4
35	Validación de parámetros de búsqueda	Codificación	Terminada	Dolores García	1
50	Secuencia Búsquedas N	Codificación	Terminada	Dolores García	3
51	Secuencia Búsquedas ?	Codificación	Terminada	Dolores García	3
38	Servicio creación nuevos de datos	Codificación	Terminada	Dolores García	4
52	Resumen Análisis	Codificación	Terminada	Dolores García	4
53	Paginación resultados	Codificación	Terminada	Dolores García	4
40	Vista Controlador Presentación resultados	Codificación	Terminada	Dolores García	4
41	Vista controlador Resumen resultados	Codificación	Terminada	Dolores García	4
44	Presentación datos en tabla	Codificación	Terminada	Dolores García	2
45	Presentación datos en histograma	Codificación	Terminada	Dolores García	4
46	Exportación de datos a excel	Codificación	Terminada	Dolores García	4
54	Exportación de resumen a excel	Codificación	Terminada	Dolores García	4
48	Reutilización de estudio	Codificación	Terminada	Dolores García	1

Gráfica de quemado

Se observa perfectamente que la línea de trabajo ha ido a la par con la línea estimada, incluso en ciertos momentos ligeramente por debajo, lo que indica que la estimación del trabajo fue buena y que pudieron emplearse en el desarrollo las horas adecuadas.

En la gráfica de las tareas pendientes, vemos que el ritmo de finalización de las tareas fue bueno y que en los casos en los que se mantenía el número de tareas pendientes coincidía con tareas de ejecución más larga.



Impedimentos

No se han encontrado impedimentos a lo largo de la ejecución de éste *sprint*

Revisión del SPRINT

En la reunión y demostración del desarrollo mantenida con el usuario, no se ha planteado ningún problema ni modificación sobre la funcionalidad desarrollada ni la que falta por desarrollar.

Retrospectiva

Nada destacable al respecto del tercer *sprint*

4.4.4 Sprint 4

Definición de los objetivos del SPRINT.

El objetivo de este *sprint* es la persistencia de los parámetros sobre los que se realizan los análisis así como los resultados obtenidos en la Base de Datos de MongoDB.

Product Backlog.

Estas son las tareas y los tiempos definidos para el *sprint* 4.

La fecha de inicio del mismo fue el 13 de abril y la de fin el 23 de abril.

SPRINT		INICIO	DURACIÓN
4		13-abr-15	10

Tareas pendientes		13-abr	14-abr	15-abr	16-abr	17-abr	18-abr	19-abr	20-abr	21-abr	22-abr	23-abr	24-abr	25-abr	26-abr	27-abr	28-abr	29-abr	30-abr
Horas de trabajo pendientes		30	29	29	29	25	25	25	23	23	19	15	15	15	15	15	11	7	3

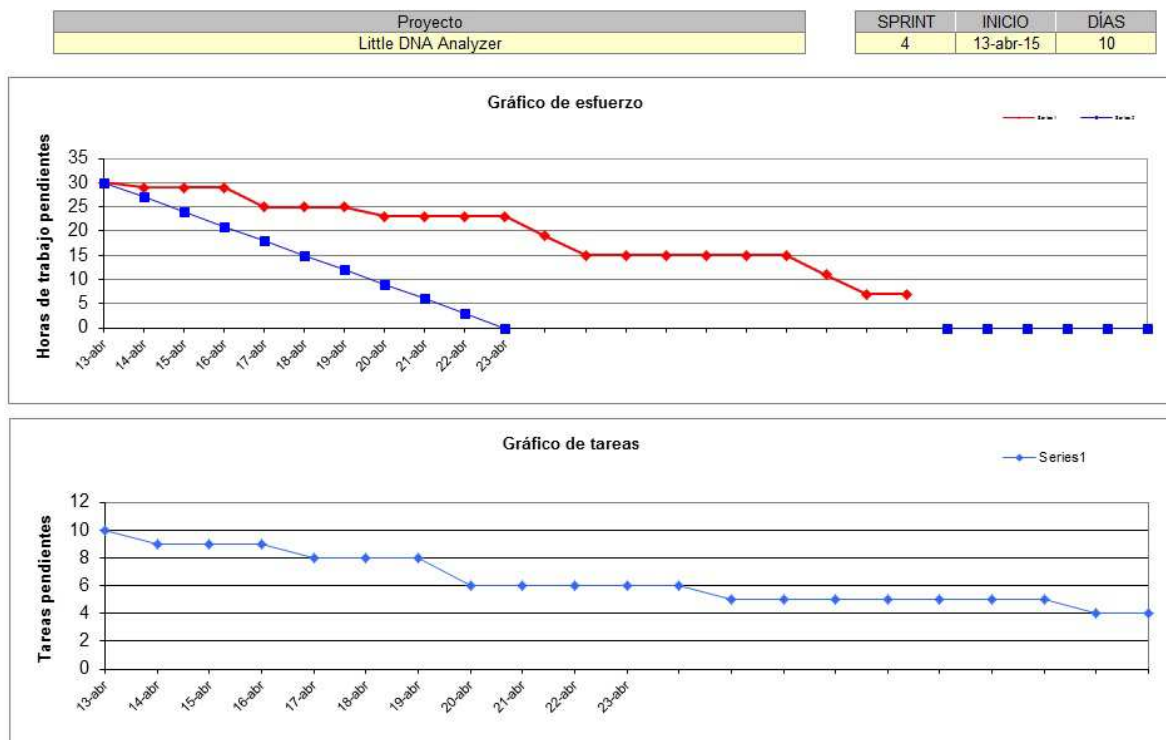
PILA DEL SPRINT		Tipo	Estado	Responsable	13-abr	14-abr	15-abr	16-abr	17-abr	18-abr	19-abr	20-abr	21-abr	22-abr	23-abr	24-abr	25-abr	26-abr	27-abr	28-abr	29-abr	30-abr
Backlog ID	25	Revisión mensajes Pantalla Selección estudio	Codificación	Terminada	Dolores García	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	27	Integración Servicios Rest	Codificación	Terminada	Dolores García	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	29	Seleccionar Fichero	Codificación	Terminada	Dolores García	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	33	Revisión Mensajes pantalla secuencia	Codificación	Terminada	Dolores García	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	37	Conexion BBDD Mongo	Codificación	Terminada	Dolores García	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	39	Revisión mensajes pantalla parametros	Codificación	Terminada	Dolores García	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	42	Consultas BBDD Mongo	Codificación	Terminada	Dolores García	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	43	Borrado de datos	Codificación	Terminada	Dolores García	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	47	Donar Resultado tras analisis	Codificación	Terminada	Dolores García	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	49	Revisión mensajes pantalla resultados y resumen	Codificación	Terminada	Dolores García	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Gráfica de quemado

Se deduce tras ver la gráfica, que la finalización del desarrollo no tuvo lugar en la fecha estimada, que era el 23 de abril, si no que tuvo lugar el 4 de Mayo. En los periodos entre el 17 y 19 de abril, 20 y 23 de abril y 25 y 30 de abril no hubo avance ni se finalizó ninguna tarea.

No se debe a ningún tipo de impedimento relacionado con el proyecto, sino más bien con falta de tiempo para dedicar al desarrollo.

En la gráfica de las tareas pendientes, vemos que el ritmo de finalización de las tareas fue lento y que durante tres periodos concretos no se finalizó ni avanzó con ninguna tarea, lo que, evidentemente, retrasó la finalización del *sprint*.





Impedimentos

Los únicos impedimentos encontrados durante éste periodo, son los relativos a la falta de tiempo para dedicarlo al desarrollo.

Revisión del SPRINT

En la reunión y demostración del desarrollo, no se encontró ningún tipo de mejora ni modificación sobre el desarrollo realizado.

Sí se recogieron distintos puntos de mejora que pueden ser tenidos en cuenta y que se desarrollan en este documento, en el punto 10 sobre futuras líneas de trabajo.

Retrospectiva

Se destaca en la retrospectiva la falta de tiempo dedicado al proyecto, tomando como determinación dedicar más tiempo al día.

5. Entorno y Tecnología

A continuación vamos a introducir el entorno y las tecnologías utilizadas en este proyecto, así como la arquitectura y los modelos de datos utilizados.

5.1 Introducción

Por todo lo expuesto anteriormente, se decide que la aplicación será una aplicación web, que ahorrará tiempo al usuario, ya que no requiere ningún tipo de instalación previa de programas en sus equipos. Además, cualquier actualización de la aplicación será totalmente transparente a él y no ocupará espacio en su disco duro. Cabe destacar igualmente, que podrán utilizar cualquier sistema operativo y no encontrarán problemas de compatibilidad al respecto y que será accesible desde cualquier ordenador con acceso al servidor en el que esté desplegada la aplicación.

Aunque la aplicación será accesible desde cualquier navegador, si cabe destacar y recomendar su uso desde Chrome, ya que el desarrollo estará optimizado para dicho navegador.

La aplicación estará desplegada en un servidor web desde el que se servirán las distintas páginas con la información solicitada por los usuarios. El protocolo utilizado para dichas transacciones es el protocolo de transferencia de hipertexto, del inglés Hypertext Transfer Protocol (de ahora en adelante HTTP.)

El protocolo HTTP se utiliza en todas las transacciones de la red informática mundial (World Wide Web). HTTP define la sintaxis y semántica utilizada en los elementos web para comunicarse. Se trata de un protocolo orientado a transacciones, que sigue un esquema petición- respuesta entre el cliente y el servidor. Esto quiere decir, que el funcionamiento básico de cualquier aplicación web es el siguiente:

- El usuario hace una solicitud desde el navegador, mediante el protocolo HTTP.
- El servidor web recibe dicha petición y localiza la página que debe servir, devolviéndola al navegador que la solicitó.

Las aplicaciones web se basan en el modelo Cliente/Servidor, en el que se gestionan servidores Web y que utilizan como interfaz páginas web. Se trata de un modelo de aplicación distribuida donde hay un reparto de tareas entre los servidores y los clientes.

Podemos decir que las páginas web, ya sean almacenadas o creadas de forma dinámica con la información de los servidores, son el componente principal de las aplicaciones web.

Se seguirá para el desarrollo de ésta aplicación el modelo Vista-Controlador, en adelante MVC. Se trata de un patrón de arquitectura software en el que los datos y la lógica de negocio se separan de la interfaz de usuario y de la gestión de los eventos. Se consigue mediante la separación en tres capas o componentes que son el modelo, la vista y el controlador. Básicamente, se define por un lado cómo se presenta la información y por otro cómo se interacciona con el usuario.

Se definen por lo tanto, los siguientes componentes:

- El modelo o representación de la información del sistema. Gestiona los accesos de la información y controla los accesos a la propia aplicación. La información le llega a través del controlador.
- La vista, donde se presentan los datos del modelo de forma que el usuario pueda interactuar con ella.
- El controlador que responde a los eventos o acciones por parte, normalmente del usuario, invocando peticiones al modelo. Se trata, por así decirlo de un intermediario entre la vista y el modelo.

Por lo general, el usuario es el encargado de interactuar con la interfaz de usuario, ejecutando ciertas acciones. El controlador recibe desde la vista la notificación de dicha acción. Tras recibir la notificación, el controlador accede al modelo, realizando las acciones correspondientes (consulta, actualización...) según lo solicitado. El controlador se encarga de notificar de algún modo estas acciones realizadas sobre el modelo al usuario, a través de la vista.

Inicialmente MVC se desarrolló para aplicaciones de escritorio, pero ha sido adaptado para el diseño de aplicaciones web. Existen varios frameworks que implementan este patrón. Podemos decir que un framework son un conjunto de clases que nos facilitan el trabajo cotidiano en las aplicaciones, utilizándolo para crear el conjunto de objetos que nuestra aplicación va a necesitar. El elegido para el desarrollo de éste proyecto es Spring, que describiremos en los siguientes puntos de ésta memoria.

5.2 Tecnología Hardware

A continuación vamos a detallar la tecnología hardware necesaria para el uso de la aplicación, así como la arquitectura de la misma.

5.2.1 Tecnología Hardware en el Servidores

La Raspberry pi^[14] es un ordenador de placa reducida, es decir, un ordenador completo en un sólo circuito. El diseño se centra en un sólo microprocesador con la RAM, todo en una sola tarjeta de tamaño reducido y que contiene todo lo que necesita en la placa base.

Nace con la idea de estimular la enseñanza de las ciencias de la computación en las escuelas.

En nuestro caso, vamos a configurar la Raspberry como servidor de aplicaciones web, y utilizaremos el modelo Raspberry pi 2, model B. Se trata de la segunda generación de Raspberry, y reemplaza el original desde febrero de 2015.

Sus características básicas son las siguientes:

- Quad Core ARM Cortex-A7 CPU a 900MHz
- 1 GB de RAM
- 4 Puertos USB
- Puerto Ethernet

Sobre ella puede instalarse un amplio rango de distribuciones de Linux, así como Windows 10.

Como método de almacenamiento tendremos una tarjeta SD de 64 gigas y además, para almacenamiento de Base de datos, dispondremos de un disco duro externo autoalimentado de 2TB.

5.2.2 Tecnología Hardware en las estaciones de trabajo

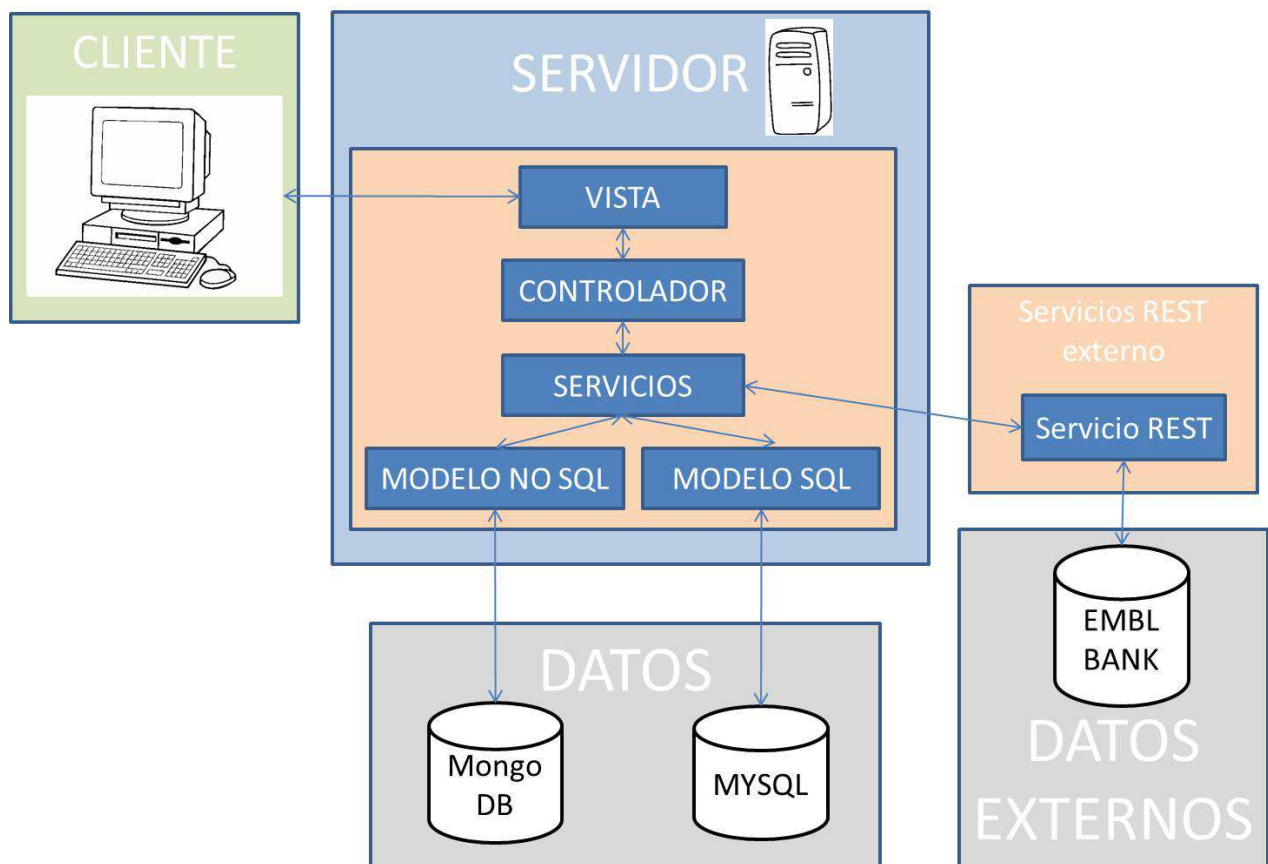
La aplicación podrá ser utilizada desde cualquier ordenador con conexión a internet y con un navegador web, preferiblemente Chrome, ya que la aplicación ha sido optimizada para éste navegador.

5.3 Tecnología Software

A continuación vamos a detallar la arquitectura definida para la aplicación, así como las tecnologías elegidas para el desarrollo de la misma.

5.3.1 Arquitectura

Diagrama de arquitectura





Descripción de la aplicación.

A continuación pasamos a describir las distintas tecnologías y *frameworks* utilizados en el desarrollo de la aplicación y porqué se han elegido.

5.3.2 Tecnologías

LENGUAJE

El lenguaje de programación elegido para el desarrollo de ésta aplicación es JAVA. Se decide así por el alto conocimiento del desarrollador en él y porque es el que mejor se adapta a la arquitectura y *frameworks* elegidos para el desarrollo.

FRAMEWORK SPRING^[15]

Para el desarrollo de ésta aplicación elegimos como *framework* Spring, más concretamente su módulo Spring MVC. Spring es un *framework* ligero y no intrusivo, ya que generalmente los objetos que programamos no tienen dependencias de clases específicas de Spring. Sus características principales son la inyección de dependencias y la programación orientada a aspectos:

- Inyección de dependencias: Su objetivo es lograr un bajo acoplamiento entre los objetos de la aplicación. Con este patrón los objetos no crean ni buscan sus dependencias, si no que éstas vienen dadas al objeto. El contenedor, o entidad que coordina cada objeto, se encarga de hacer este trabajo en el momento en que se instancia el objeto. Se cambia la responsabilidad en cuanto a la forma en que un objeto obtiene la referencia de otro. Los objetos pueden conocer sus dependencias a través de su interfaz, de forma que la dependencia puede intercambiarse por distintas implementaciones desde el contenedor. Es decir, que se programará orientando a interfaces e inyectando implementaciones desde el contenedor.
- Programación orientada a aspectos: Es un paradigma de programación que intenta separar funcionalidades secundarias de la lógica de negocio. Existen ciertas funcionalidades como las trazas, la seguridad o la forma de manejar la transacción que son transversales a nuestro programa. Por lo tanto, es muy fácil caer en la duplicidad de código y acoplamiento de la lógica de negocio. Mediante la programación orientada a aspectos se busca separar en módulos estos servicios, de forma que puedan aplicarse a los componentes que deban afectar.

Spring cuenta con distintos módulos, parte de los cuales serán utilizados en nuestra aplicación:

Módulos de Spring Framework



- Utilizaremos el núcleo o core, donde se encuentra el contenedor fundamental de Spring o *BeanFactory*.
- Spring Web, que aporta clases orientadas al desarrollo web. Cuenta con el paquete Spring MVC, que es en concreto un módulo basado en HTTP y *Servlets* que provee de herramientas para la extensión y personalización de aplicaciones y servicios web REST (Transferencia de Estado Representacional, del inglés Representational State Transfer). Un *servlet* es una clase en lenguaje java que es utilizada en la ampliación de las utilidades de un servidor. Inicialmente se refería al término REST como un conjunto de principios de arquitectura, pero actualmente se utiliza para describir una interfaz web simple que utiliza XML y HTTP.

Utilizaremos también Spring Security, para la gestión de la seguridad y autenticación en la aplicación.

MAVEN^[16]

Para la gestión de las dependencias del proyecto así como la construcción del mismo vamos a utilizar Maven. Se trata de una herramienta que nos ayuda a ambas cosas. Maven utiliza un *Project Object Model*, de ahora en adelante POM, para describir el proyecto a construir, definiendo en las dependencias de otros módulos y los componentes externos, así como el orden de construcción de los elementos. Facilita la compilación del código y su empaquetado. El motor que incluye el núcleo de Maven puede descargar dinámicamente *plugins* de un repositorio. En éste repositorio se tiene acceso a distintas versiones de muchos proyectos *Open Source* (de código abierto) en Java, de

apache y otras organizaciones de desarrolladores. Utilizando Maven es posible no sólo proveerse de archivos de su repositorio, sino subir artefactos al repositorio al terminar la construcción de la aplicación, dando acceso al resto de usuarios a ella.

TILES^[17]

En la parte de la vista vamos a utilizar Tiles. Se trata de un *framework* de plantillas o patrones, mediante los cuales podemos simplificar la presentación de la interfaz de usuario. Tiles permite a los desarrolladores definir fragmentos o trozos de páginas que pueden ir montándose hasta conseguir páginas completas en tiempo de ejecución. Estos fragmentos evitan la duplicidad de elementos comunes dentro de una página, incrustándose en otras páginas originando así una serie de plantillas reutilizables. De ésta forma, se agiliza el desarrollo de la interfaz de usuario, logrando una apariencia de aplicación consistente y similar en toda la aplicación.

JQUERY

Se trata de una biblioteca *Javascript*, que permite simplificar la manera de interactuar con los documentos HTML, así como manejar eventos y agregar interacción con AJAX a páginas web.

Es software libre y de código abierto, ofrece funcionalidades basadas en Javascript que de otra forma requerirían de mucho más código, es decir, con las funciones propias se logran grandes resultados en menos tiempo y espacio.

AMCHARTS^[18]

Se trata de un componente *Javascript* utilizado para la generación de gráficos. El parámetro que recibe para pintarla es un JSON.

MONGODB^[19]

En lo que respecta a las bases de datos, se ha decidido dividir la solución, ya que así será posible aprovechar al máximo las características de las bases de datos SQL y NO SQL.

Para la parte de análisis de la aplicación elegimos MongoDB. Se trata de un sistema de Base de Datos No SQL orientado a documentos. En vez de guardar los datos en tablas como se hace habitualmente en las bases de datos relacionales, se guardan estructuras de datos en documentos JSON (de *Javascript Object Notation*), con un esquema dinámico, llamado BSON. JSON es un formato ligero para el intercambio de datos, es un subconjunto de la notación literal de objetos de *Javascript* que no requiere el uso de XML. Utilizando esto, la integración de los datos en ciertos casos es más fácil y rápida.

Las principales características de MongoDB son las siguientes:

- Consultas Ad hoc.
- Indexación: Su concepto es similar al de las bases de datos relacionales.
- Replicación: soporta la replicación maestro-esclavo, mientras el maestro puede ejecutar comandos de lectura y escritura, el esclavo puede copiar los datos del maestro y usarlos para lectura o copia de seguridad. En caso de caída del servicio, el esclavo tiene el poder de elegir



un nuevo maestro.

- Balanceo de carga: es posible escalar de forma horizontal, usando un maestro con uno o más esclavos. Se elige una clave que determinará cómo se van a distribuir los datos de la colección. MongoDB es capaz de ejecutarse en varios servidores, balanceando la carga o duplicando los datos para mantener el sistema en funcionamiento en caso de fallo hardware.

Por otro lado, cuenta con ciertas desventajas que hay que tener en cuenta, cómo es que durante la escritura se bloquea la base de datos al completo, lo que reduce la concurrencia de forma drástica. Éste caso es punto de mejora en MongoDB y se aparecerá en futuras versiones. Otro problema en el momento de la escritura, es que retorna cuando la información aún no ha sido escrita en el espacio de almacenamiento permanente, lo que puede ocasionar ciertas pérdidas de información en caso de fallo durante la escritura. Y para finalizar, el rendimiento de MongoDB se ve reducido cuando el volumen supera los 100GB, lo que causa problemas de escalabilidad.

MySQL

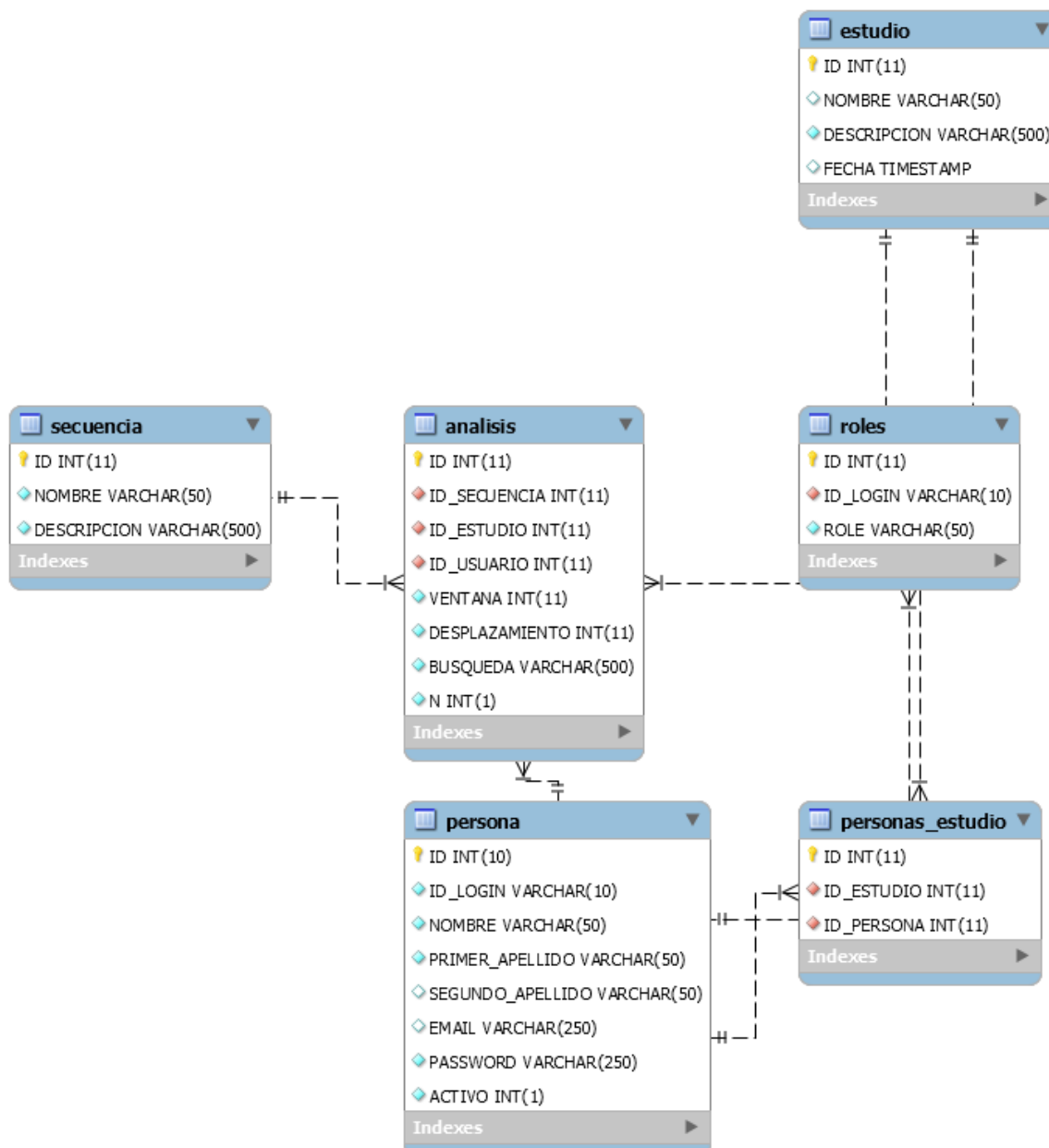
Se trata de la elección que haremos para la parte de gestión de la aplicación. La parte de análisis propiamente dicha y la parte de gestión de usuario son independientes y sus requerimientos son totalmente distintos. Se trata de algo típico en la mayor parte de las aplicaciones, por lo que se ha elegido una Base de Datos tradicional para aislarlo de la parte de análisis, aislando ambas partes y aprovechando mejor lo que cada tipo de Base de Datos nos ofrece.

5.4 Diagramas de Base de datos

Como se ha explicado anteriormente, la aplicación tendrá dos bases de datos, cada una con características muy diferentes. A continuación, vamos a detallar la estructura de cada una de ellas.

5.5.1 Base de datos SQL

Este es el diagrama de la base de datos MySQL de la aplicación Little DNA Analyzer





5.5.2 Base de datos NO SQL

Uno de los principales motivos por los que se elige utilizar MongoDB como base de datos, es que la estructura de los datos guardados puede ser modificada de forma sencilla, sin necesidad de modificar los datos ya existentes ni de ampliar o editar las tablas.

Es decir, no existe un modelo como tal, ya que los datos se adaptan a las necesidades existentes.

Sí se han definido unas bases mínimas que deben cumplirse para los datos que ya se están guardando, pero nos da libertad para futuras mejoras y para la incorporación de nuevas búsquedas más específicas en adelante.

Se han definido tres colecciones básicas y necesarias:

- Resultado: Se trata del resultado total del análisis. En función del tamaño del resultado, es posible que sea necesario dividirlo en varios documentos para guardarlos. La estructura básica definida para los resultados incluye los siguientes datos:
 - o `_id` único de documento, asignado al crearlo.
 - o `_class`: nombre de la clase java que contiene el modelo del documento
 - o `idEstudio`: referencia al estudio al que pertenece el documento
 - o `idAnalisis`: referencia al análisis creado, para poder agruparlos en caso de división
 - o `pagina`: número de página del análisis, para que en caso de división se pueda reordenar.
 - o `json`: documento *json* que recoge los resultados para poder presentarlos directamente en pantalla de resultados
- Resumen: Se trata del resumen de resultados más relevantes del estudio. Coincide con la información mostrada en la pantalla de Resumen. Se ha definido la siguiente estructura básica de datos:
 - o `_id`: único de documento, asignado al crearlo.
 - o `_class`: nombre de la clase java que contiene el modelo del documento
 - o `idEstudio`: referencia al estudio al que pertenece el resumen
 - o `idAnalisis`: referencia al análisis al que pertenece el resumen
 - o `identificador`: identificador de la secuencia analizada
 - o `json`: documento *json* que recoge el resumen de los resultados más relevantes, para poder presentarlos directamente en la pantalla de resumen.
- Secuencia Partida: Secuencia analizada al completo. En función del tamaño de la secuencia, es posible que sea necesario dividirla en varios documentos. Se ha definido la siguiente estructura básica de datos para la secuencia:
 - o `_id`: único de documento, asignado al crearlo.
 - o `_class`: nombre de la clase java que contiene el modelo del documento
 - o `nombre`: identificador de secuencia
 - o `orden`: Orden para reordenar la secuencia, en caso de división
 - o `idSecuencia`: identificador único de secuencia



- Secuencia: Lista que contiene cadenas de caracteres con trozos de la secuencia analizada. Se decidió este tipo de guardado de la secuencia, en lugar de utilizar una cadena de caracteres (*String*) más largo, por la posibilidad y facilidades que ofrece para el posterior análisis de la secuencia en caso de reutilización de la misma. En el momento de dividir en función de una nueva ventana y desplazamiento, resulta más eficaz y rápido hacerlo sobre cadenas de caracteres cortas que sobre una cadena de caracteres muy larga, ya que a la aplicación resulta más lenta en este último caso.



6. Presupuesto

A continuación vamos a detallar los costes del proyecto de forma detallada.

Todas las tecnologías que se han utilizado en el proyecto son de fuente abierta, también conocidas como *opensource*, así como las herramientas utilizadas para la generación de la documentación, por lo que en el presupuesto no se incluye ninguna licencia.

Únicamente ha intervenido un recurso en el análisis, diseño y construcción del proyecto, ya que el resto de personas implicadas pueden ser consideradas como “clientes” del producto generado. Se considerará que se trata de un Ingeniero Junior a 30 euros la hora.

Como se ha indicado anteriormente, la aplicación va a desplegarse en una Raspberry PI 2 Model , a la que se añadirá una tarjeta SD de 64 gigas y un disco duro de 2 TB.

Concepto	Horas	Precio/Hora	Total
Análisis, Diseño, Implementación y Pruebas Unitarias	232	30	6960
Reuniones Cliente	3	30	90
Plan de pruebas – Implementacion pruebas	56	30	1680
Documentación	45	30	1350
TOTAL			10080

Concepto	Unidades	Precio Unidad	Total
Portatil Desarrollo	1	800	800
Raspberry PI 2 Model	1	39,99	72,48
Caja y accesorios Raspberry PI	1	15,00	15,00
Tarjeta SD 64 GB	1	21,20	21,20
Disco duro 2TB	1	80,50	80,50
TOTAL			989,18

Por lo que el coste total del proyecto es de 11069,18



7. Manual de Referencia Técnica

7.1 Instalación del proyecto en el entorno de desarrollo-

Generación del proyecto con MAVEN

En el caso de que se desee montar el entorno de desarrollo sobre el IDE Eclipse, deberán seguirse los siguientes pasos:

- Descargar o descomprimir el código fuente del proyecto en la carpeta deseada.
- Comprobar que el ordenador en el que va a crearse el proyecto tiene instalado maven. Para ello deberá ejecutar por línea de comandos:

```
1. mvn --version
```

En caso de que maven ya haya sido instalado en el ordenador, deberá obtener una salida similar a ésta:

```
1. Apache Maven 3.0.5 (r01de14724cdef164cd33c7c8c2fe155faf9602da; 2013-02-19
14:51:28+0100)
```

```
2. Maven home: D:\apache-maven-3.0.5\bin\..
```

```
3. Java version: 1.6.0_25, vendor: Sun Microsystems Inc.
```

```
4. Java home: C:\Program Files\Java\jdk1.6.0_25\jre
```

```
5. Default locale: nl_NL, platform encoding: Cp1252
```

```
6. OS name: "windows 7", version: "6.1", arch: "amd64", family: "windows"
```

En la que se indicará cual es la versión de maven instalada.

En caso contrario, deben seguirse los pasos que se encontrará en el enlace para instalar maven en su equipo antes de continuar:

<https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>

Una vez haya finalizado la instalación de maven, deberá acceder por línea de comandos a la carpeta que contenga el fichero pom.xml. Una vez en ella, ejecutará el siguiente comando:

```
1- mvn clean install eclipse:eclipse
```

Una vez ejecutado esto, podrá importar el proyecto en Eclipse para comenzar a trabajar con él.

Instalación de las BBDD

Como ya se ha descrito anteriormente, la aplicación dispone de dos bases de datos que funcionan conjuntamente para explotar mejor sus capacidades y las de la aplicación. Por lo tanto, para trabajar sobre la aplicación, deberá disponerse de ambas.

Puede encontrarse información para la instalación de las bases de datos utilizadas en los siguientes enlaces:

<https://dev.mysql.com/downloads/installer/>

<http://docs.mongodb.org/manual/installation/>

Se recomienda el uso de MySQL por haberse implementado la aplicación para dicha base de datos, lo que hará que no sea necesario modificar nada. Pero, si desea utilizar otra base de datos relacional, deberá modificarse el fichero webmvc-config.xml, en el que encontramos la propiedad "hibernate.dialect", donde deberemos indicar contra que base de datos vamos a conectarnos.

```
<!-- Hibernate Session Factory -->
<bean id="mainSessionFactory"
    class="org.springframework.orm.hibernate4.LocalSessionFactoryBean">
    <property name="dataSource" ref="mainDataSource" />
    <property name="packagesToScan">
        <array>
            <value>com.c3.analyzer.*</value>
        </array>
    </property>
    <property name="hibernateProperties">
        <props>
            <prop key="hibernate.dialect">org.hibernate.dialect.MySQLDialect</prop>
            <prop key="hibernate.show_sql">true</prop>
        </props>
    </property>
</bean>
```

Además, deberemos modificar el fichero library.xml, en el que encontramos las propiedades de conexión así como usuario, contraseña y esquemas de ambas bases de datos

```
<bean id="mainDataSource" class="org.apache.commons.dbcp.BasicDataSource"
    destroy-method="close">
    <property name="driverClassName" value="com.mysql.jdbc.Driver" />
    <property name="url"
        value="jdbc:mysql://localhost:3306/gestiondnaanalyzer" />
    <property name="username" value="root" />
    <property name="password" value="root" />
</bean>
<bean id="mongo" class="org.springframework.data.mongodb.core.MongoFactoryBean">
    <property name="host" value="localhost" />
</bean>
<!-- MongoTemplate for connecting and querying the documents in the database -->
<bean id="mongoTemplate" class="org.springframework.data.mongodb.core.MongoTemplate">
    <constructor-arg name="mongo" ref="mongo" />
    <constructor-arg name="databaseName" value="analyzer" />
</bean>

<!-- Use this post processor to translate any MongoExceptions thrown in @Repository annotated classes -->
<bean class="org.springframework.dao.annotation.PersistenceExceptionTranslationPostProcessor" />
```

Creación de las BBDD

La única base de datos que necesita de una creación de esquemas es la base de datos relacional, para lo que deberá ejecutarse el fichero CREATE_DATABASE.sql que se incluye en el código del proyecto.

7.2 Ficheros de configuración de seguridad

La aplicación utiliza spring-security para controlar los accesos a la aplicación. En caso de querer modificar el tipo de acceso, crear nuevos roles de usuario o modificar el lugar o las tablas a las que se accede para controlar si el usuario es válido, deberá modificarse el fichero spring-security.xml, que tiene la siguiente información:

```
<security:global-method-security secured-annotations="enabled" />

<security:http auto-config="true">
  <!-- Restrict URLs based on role -->
  <security:intercept-url pattern="/login*" access="IS_AUTHENTICATED_ANONYMOUSLY" />
  <security:intercept-url pattern="/index*" access="IS_AUTHENTICATED_ANONYMOUSLY" />
  <security:intercept-url pattern="/action/login*" access="IS_AUTHENTICATED_ANONYMOUSLY" />
  <security:intercept-url pattern="/action/loginError*" access="IS_AUTHENTICATED_ANONYMOUSLY" />
  <security:intercept-url pattern="/logout*" access="IS_AUTHENTICATED_ANONYMOUSLY" />
  <security:intercept-url pattern="/recursos/css/**" access="IS_AUTHENTICATED_ANONYMOUSLY" />

  <security:intercept-url pattern="/action/**" access="ROLE_USER, ROLE_ADMIN" />
  <security:intercept-url pattern="/recursos/js/**" access="ROLE_USER, ROLE_ADMIN" />

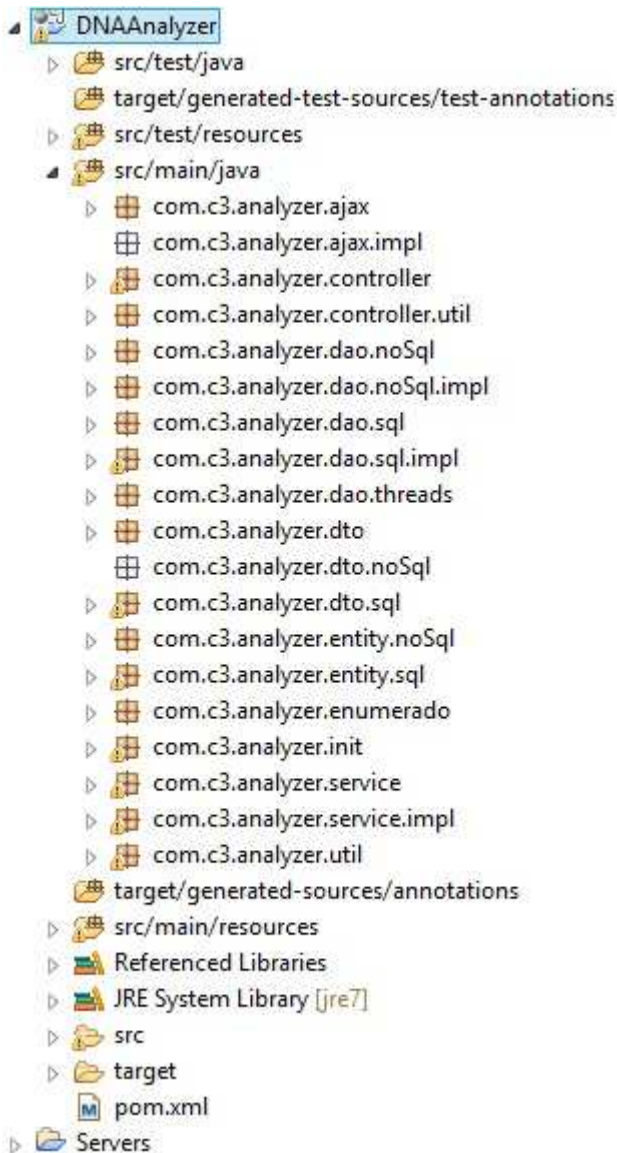
  <!-- Override default login and logout pages -->
  <security:form-login login-page="/action/login"
    default-target-url="/action/customLogin"
    always-use-default-target="true"
    authentication-failure-url="/action/loginError" />

  <security:logout
    invalidate-session="true"
    logout-success-url="/" />
</security:http>
<authentication-manager>
  <authentication-provider>
    <jdbc-user-service data-source-ref="mainDataSource"
      users-by-username-query="
        select id_login, password, activo from persona where id_login = ?"
      authorities-by-username-query="
        select id_login, role as authorities from roles where id_login = ?" />
    </authentication-provider>
  </authentication-manager>
</beans:beans>
```



7.3 Estructura del proyecto JAVA

El proyecto tiene la siguiente distribución de paquetería



A continuación vamos a describir la paquetería, definiendo que clases se incluyen en cada paquete. Se indicarán igualmente algunas pautas sobre qué deben contener las clases de los distintos paquetes para que el seguimiento de la propia arquitectura sea más sencillo y para facilitar igualmente cualquier modificación o mejora en el proyecto.

**com.c3.analyzer.entity**

Dentro de este paquete encontramos una nueva división, necesaria por el acceso a dos bases de datos muy distintas desde la aplicación.

- .sql – Dentro de éste paquete encontraremos las clases con el mapeo de las tablas de la Base de Datos SQL
- .noSql – Dentro de éste paquete encontraremos las clases con el mapeo de las colecciones de la Base de Datos no SQL

Todas las clases que se encuentren dentro de ésta paquetería tendrán el sufijo Entity añadido al nombre descriptivo de las mismas.

Para cada columna o entrada de la tabla o colección de la base de datos se definirá una variable del tipo correspondiente, de forma que pueda ser utilizada por la aplicación.

com.c3.analyzer.dao

Al igual que en el caso de la paquetería entity, encontramos este paquete dividido en .sql y .noSql. Para cada entity definido tendremos un interfaz con la definición de los métodos disponibles de acceso para esa tabla o colección en la base de datos, siguiendo la siguiente nomenclatura:

- .sql – En el caso de una interfaz para un entity que accederá a la Base de Datos SQL – IXxxxxSQLDAO, correspondiéndose las “xxxxxx” con el nombre identificativo de la tabla sobre la que opera.
- .noSql – En el caso de que sea la interfaz para un entity que accede a una colección de la Base de Datos no SQL será – IXxxxxNoSQLDAO, correspondiéndose las “xxxxxx” con el nombre identificativo de la colección sobre la que opera.

Cada interfaz será implementada por una clase en la que se desarrollarán los métodos definidos, siendo la nomenclatura de dichas implementaciones similar a la definida para las interfaces:

- En el caso -sql – XxxxxSQLDAOImpl
- En el caso -noSQL – XxxxxNoSQLDAOImpl

Por normativa definida por la arquitectura del proyecto, las clases de la paquetería DAO sólo podrán ser utilizadas dentro de la paquetería Services.

Una clase de la paquetería DAO siempre recibirá entities y devolverá entities. En ningún caso devolverá un dto ni lo recibirá.

**com.c3.analyzer.dto**

Se trata del mapeo de los entities definidos en la paquetería `com.c3.analyzer.entity` para el uso en el resto de las clases definidas en la aplicación. Serán las clases devueltas por las clases de la paquetería `.service` definida más adelante, para su uso en la parte del controlador. Podemos decir que los `dto`'s contienen la información que va a ser necesaria en las pantallas de la aplicación y que tienen correspondencia con la información contenida en los entities que tienen el mapeo de Base de Datos.

Para seguir con la nomenclatura, todos aquellos `dtos` que se correspondan exactamente con un entity, no se les añadirá nada al nombre de la clase, simplemente será `Xxxxx`, de forma que sea identificativo a su contenido y al `dto` al que se corresponde.

En caso de que se trate de una clase creada específicamente para su uso en alguna de las pantallas de la aplicación, el nombre será `XxxxxDto`.

com.c3.analyzer.service

En el paquete `.service` encontramos las clases que contienen los servicios encargados del tratamiento de la información de la aplicación, así como los principales algoritmos de la misma.

Para cada service definido tendremos un interfaz en el que encontraremos los servicios que podrán ser utilizados en los controladores de la aplicación. Al menos tendremos tantos services como `daos`, ya que para cada `dao` tendremos uno que hará que podamos utilizar la funcionalidad ofrecida hacia la base de datos.

La nomenclatura seguida para los interfaces será `IXxxxxService`, coincidiendo las 'xxxxx' con el nombre del `dao` que recubre o siendo éstas un nombre identificativo del contenido del servicio.

Cada interfaz de servicios tendrá una implementación, cuya nomenclatura será `XxxxxServiceImpl`, definiéndose las 'xxxxx' del mismo modo que en el interfaz.

Los servicios utilizarán los `daos` definidos para acceder a la Base de Datos y en caso de tener que devolver algún objeto completo deberá ser de los contenidos en la paquetería `dto` y nunca un entity. Los servicios serán los encargados de convertir los `dtos` a entities para enviarlos a las funciones de los `daos`. Del mismo modo, recibirán los entities de los propios `daos` para transformarlos en `dtos`.

com.c3.analyzer.controller

En la paquetería `controller` incluiremos las clases encargadas del control del flujo de las distintas páginas de la aplicación. La nomenclatura de las mismas será `XxxxxController`, siendo las `xxxxx` definidas de forma que identifiquen el contenido del controlador.



7.4 Internacionalización

En el fichero webmvc-config.xml encontramos los parámetros de configuración para la internacionalización:

Encontramos en el paquete src/main/resources, dentro de la carpeta i18n los ficheros con los textos y mensaje de la aplicación. Ahora mismo, encontramos dos ficheros:

text_en.properties: Contiene todos los textos en inglés, es el utilizado por defecto

text_es_ES.properties: Contiene todos los textos en español.

Dentro de estos ficheros encontramos todos los textos que aparecen en las pantallas de la aplicación y los mensajes de error o de aviso que se muestran desde el servidor.

Además, debemos tener en cuenta que hay ciertos mensajes que se muestran directamente desde la parte del cliente, como por ejemplo ciertas validaciones previas al envío de los formularios o las confirmaciones de ciertas acciones. Estos mensajes los encontramos dentro de la carpeta src/main/webapp/recursos/jd/bundle. Ahora mismo, al igual que ocurre en el caso anterior, sólo hay dos ficheros:

Messages_en.properties: Contiene los mensajes en inglés, es el utilizado por defecto.

Messages_es_ES.properties: Contiene los mensajes en español.

En caso de querer añadir un nuevo idioma, deberíamos crear un nuevo fichero text y otro Messages, en el que se incluirán todos los mensajes ya existentes traducidos al idioma correspondiente. El sufijo del fichero será el correspondiente al del idioma deseado.

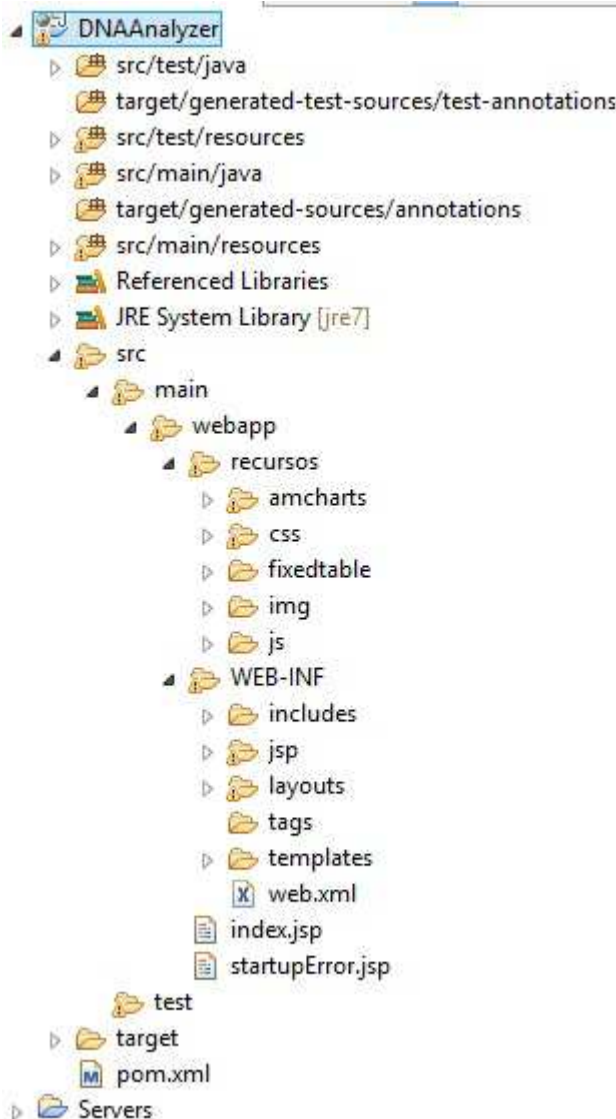
```
<bean id="localeResolver"
      class="org.springframework.web.servlet.i18n.CookieLocaleResolver"/>

<mvc:interceptors>
  <bean class="org.springframework.web.servlet.i18n.LocaleChangeInterceptor"/>
</mvc:interceptors>
<bean id="multipartResolver"
      class="org.springframework.web.multipart.commons.CommonsMultipartResolver" />
```



7.5 Estructura de la parte de la vista

Encontramos la siguiente distribución de elementos para la parte de la vista del proyecto:



A continuación vamos a describir donde encontramos las distintas partes y que ficheros deberán tenerse en cuenta:

Recursos

Amcharts

Aquí encontramos los ficheros necesarios para aplicar las utilidades de creación de histogramas en las pantallas de resumen y resultados del proyecto. No es necesario modificar nada en los ficheros que encontramos en ésta carpeta.

CSS

Se trata de la carpeta que contendrá los ficheros .css con los estilos aplicados en la aplicación.

Fixedtable

Contiene los ficheros para aplicar las utilidades de las tablas utilizadas en la parte de la administración de la aplicación

Img

Contiene las imágenes e iconos utilizados en toda la aplicación. Cualquier imagen necesaria, deberá incluirse en esta carpeta.

JS

Contiene todos los ficheros .js necesarios en la aplicación.

Existe una carpeta especial llamada JQUERY en la que se incluirán todos los ficheros del framework JQuery utilizados en la aplicación.

El resto de carpetas se distribuirá de forma idéntica a la definida en la carpeta JSP (descrita más adelante). Existirá un fichero .js para cada jsp, que contendrá las funciones específicas de dicha JSP.

WEB-INF

JSP

Además de los ficheros .jsp que contienen la información de las pantallas de la aplicación, se incluye en ésta carpeta el fichero Tiles.xml, en el que se define la configuración de los tiles utilizados.

Layouts

En esta carpeta se incluyen los layouts o disposición de elementos, diseñados para las pantallas de la aplicación.

En nuestro caso se han definido tres layouts:

- StandardLogin – Layout específico para el login
- Standard – Layout estándar de la aplicación
- StandardMenu – Layout similar al estándar, pero que incluye un menú lateral a la izquierda.

Templates

Aquí encontraremos los elementos comunes utilizados en los layouts para la presentación de las páginas. Inicialmente se ha definido:

- Header – Cabecera común utilizada en todas las pantallas de la aplicación
- Footer – Diseño del pie común de todas las pantallas de la aplicación.
- Lateral – Diseño del menú lateral utilizado en el layout StandardMenu
- SimpleHeader – Una cabecera más sencilla utilizada en el caso del login.



8. Pruebas Funcionales del proyecto

Se utiliza para las pruebas funcionales del proyecto Selenium, utilizando el *plugin* para el navegador Firefox.

Inicialmente, todos los casos de prueba creados son de resultado positivo, es decir, muestran que el funcionamiento normal de la aplicación es correcto y que no obtenemos un error en los casos bases de las pruebas. Podrían completarse los casos de prueba añadiendo casos de error esperado, validando así los casos en que la aplicación debe mostrar determinados mensajes de error.

Se han diferenciado las pruebas para la parte de administración del proyecto y para la parte de análisis puro, definiendo por lo tanto los siguientes casos de prueba:

Administración

CrearUsuario: Se crearán dos usuarios, el primero de ellos con nombre de usuario “UPrueba” que será utilizado en los casos de prueba de estudios y de usuarios. El segundo de ellos, con nombre de usuario “desactivar”, será utilizado en el caso de prueba de desactivación de usuarios.

CambiarPasswordAdmin: En este caso de prueba el usuario con rol de administrador será el que cambie la contraseña del usuario.

CambiarPasswordUsuario: En este caso de prueba, es el propio usuario el que modifica su contraseña.

EditarUsuario: El usuario con rol de administrador, modifica los datos de un usuario.

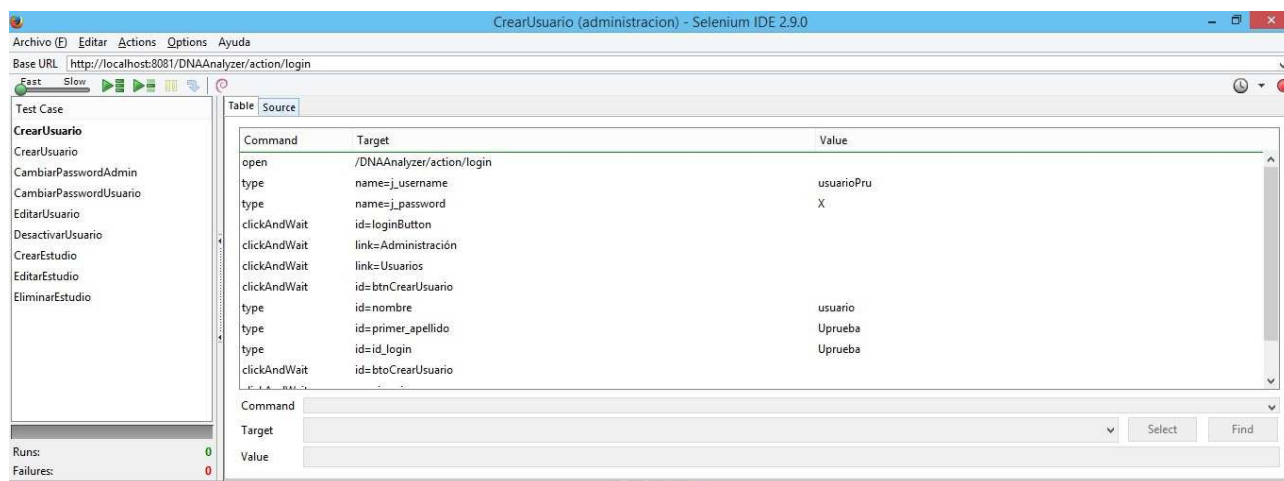
DesactivarUsuario: El usuario con rol de administrador, desactiva al usuario.

CrearEstudio: El usuario con rol de administrador, crea un nuevo estudio, en el que es incluido él mismo como usuario.

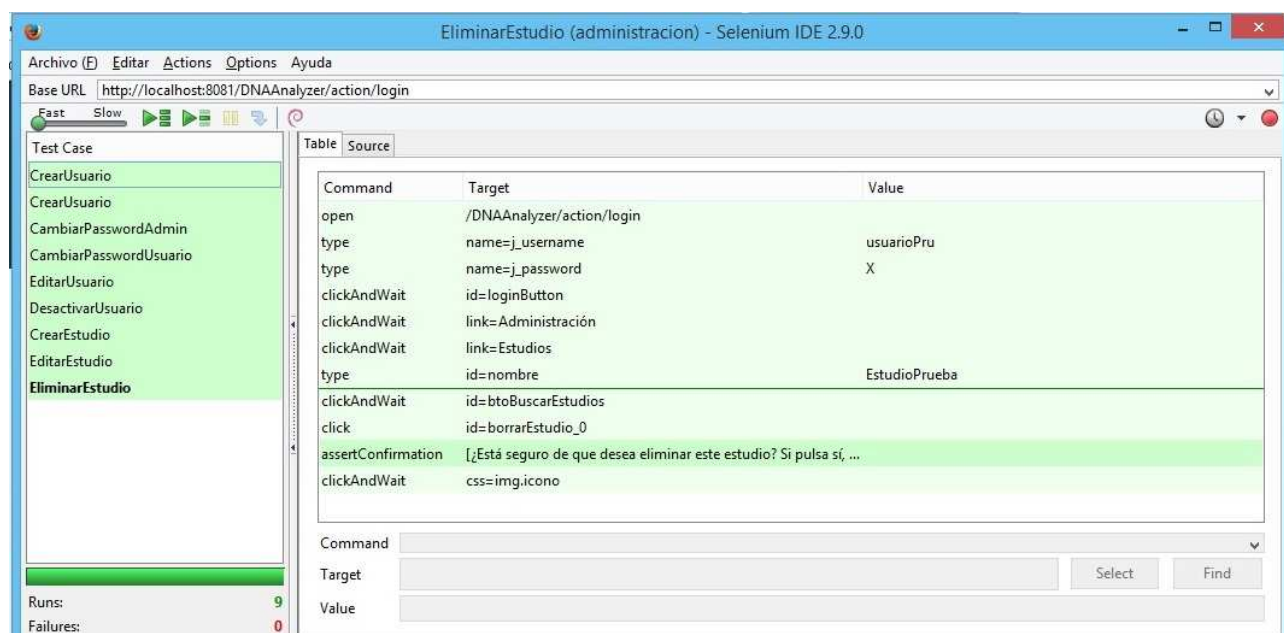
EditarEstudio: El usuario con rol de administrador, edita el estudio creado, modificando sus datos. Además, añade un nuevo usuario al estudio y se elimina otro.

EliminarEstudio: El usuario con rol de administrador, elimina el estudio creado.

Esta es la definición de los casos de prueba del proyecto para administración:



Y el resultado de su ejecución, en la que vemos que todo ha sido correcto



Análisis

CrearEstudioAnálisis: Se crea un nuevo estudio al que se añade el usuario con nombre de usuario UsuarioPru, que será el encargado de llevar a cabo los análisis.

AnálisisEMBLBANK-nuevo: Se carga una secuencia de análisis buscándola en EMBLBank y se realiza un análisis.

AnálisisEMBLBANK-duplicado: Se indica que se quiere buscar la misma secuencia que en el análisis anterior en EMBLBank y se indica que no queremos volver a guardarla, si no utilizar la



ya guardada para realizar el análisis.

AnalisisEMBLBANK-reutilizado: Se busca en la página de resultados del estudio la secuencia buscada en EMBLBank y se realiza un nuevo análisis sobre ella.

AnalisisEMBLBANK-dupNuevo: Se indica que se quiera buscar la misma secuencia que en el primer análisis en EMBLBank, pero indicamos un nuevo nombre para que se guarde nuevamente y realizar el análisis sobre esta.

AnalisisFichero-nuevo: Se indica que quiere hacerse una carga desde secuencia desde fichero, se selecciona uno del propio ordenador y se ejecuta el análisis.

AnalisisFichero-duplicado: Se indica que quiere realizarse la carga de secuencia desde el mismo fichero que en el caso anterior y se indica que no queremos volver a guardar la secuencia, si no utilizar la ya guardada para el análisis.

AnalisisFichero-reutilizado: Se busca en la página de resultados del estudio la secuencia añadida desde fichero y se realiza un nuevo análisis sobre ella.

AnalisisFichero-dupNuevo: Se indica que se quiere cargar la secuencia del mismo fichero utilizado en el primer análisis de fichero, pero indicamos un nuevo nombre para que se guarde nuevamente y ejecutar el análisis sobre esta.

AnalisisTextarea-nuevo: Se indica que quiere utilizarse la secuencia indicada en pantalla, con el nombre y la descripción dados y se ejecuta el análisis.

AnalisisTextarea-duplicado: Se indica que quiere utilizarse la secuencia indicada en pantalla, con el nombre igual al del análisis anterior y se indica que no queremos volver a guardarlo, si no que se utilizará para el análisis la secuencia con ese nombre ya guardada.

AnalisisTextarea-reutilizado: Se busca en la página de resultados del estudio la secuencia añadida desde pantalla y se realiza un nuevo análisis

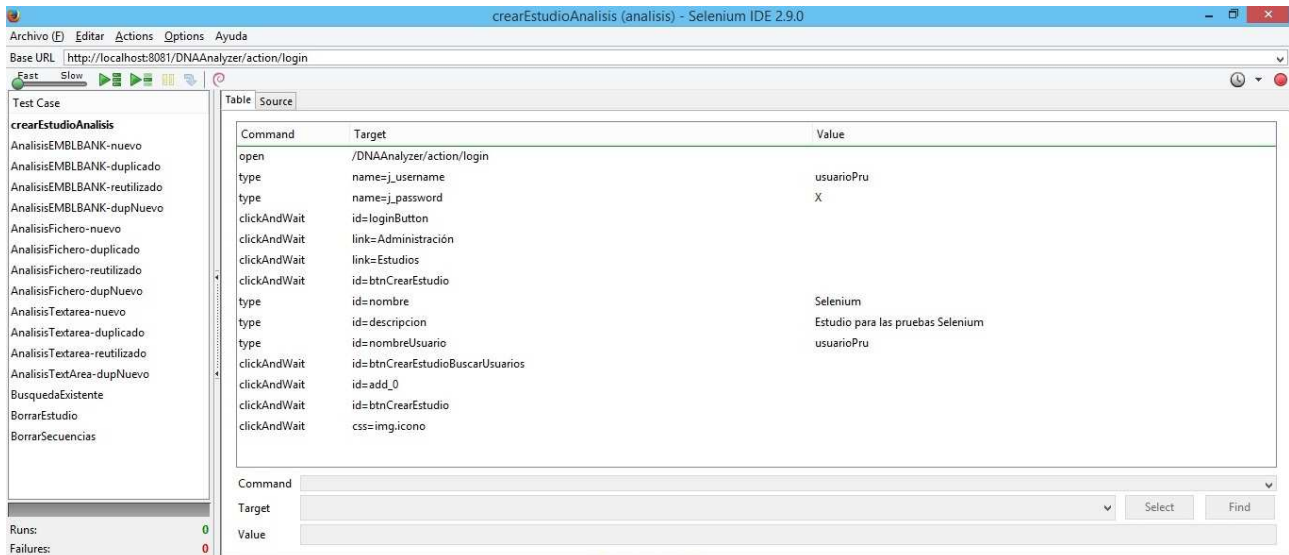
AnalisisTextarea-dupNuevo: Se indica que quiere utilizarse la secuencia indicada en pantalla con el nombre y la descripción dados, siendo el nombre el mismo que en primer análisis de Textarea. Se indicará un nuevo nombre, ya que queremos que guarde nuevamente la secuencia y que sea esta la utilizada en el análisis.

BusquedaExistente: Se indicará que quiere realizarse la búsqueda sobre una secuencia que ya existe en el sistema.

BorrarEstudio: Borrará el estudio creado para las pruebas así como todos los análisis ejecutados en él.

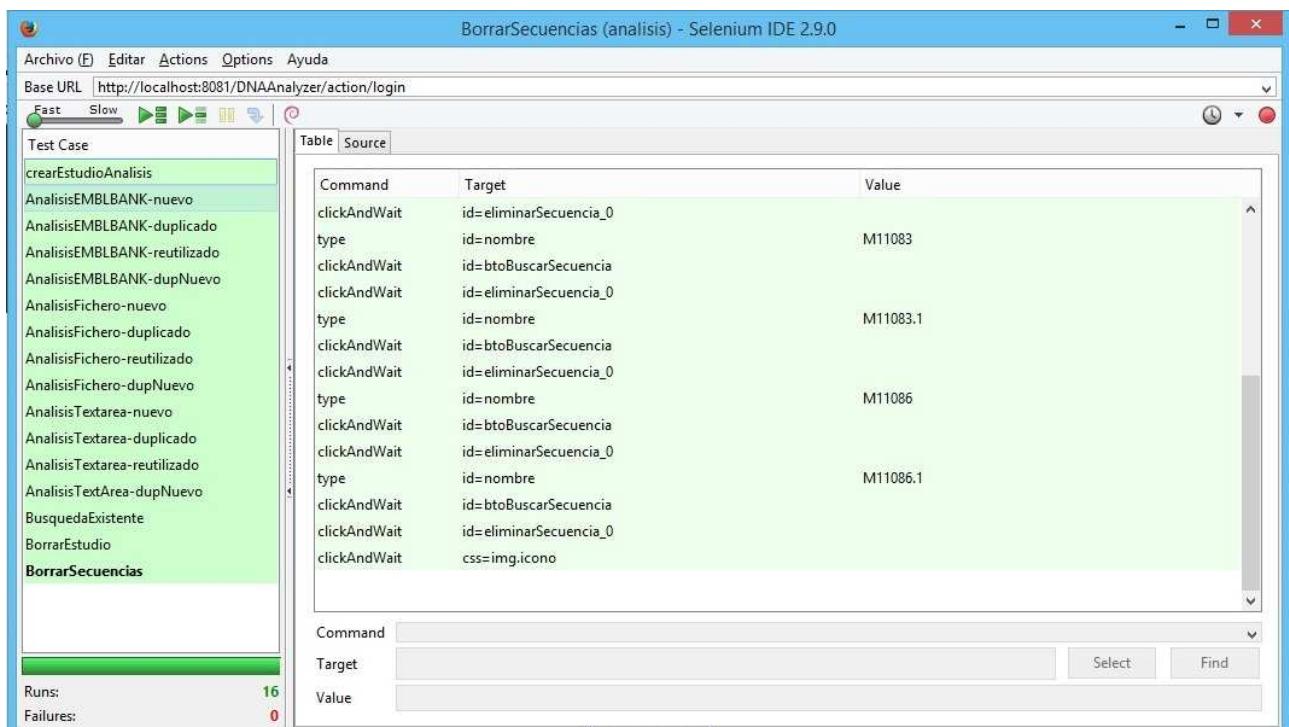
BorrarSecuencias: Se borrarán una a una, todas las secuencias utilizadas en estas pruebas.

Esta es la secuencia de pruebas ejecutadas para el análisis:



Command	Target	Value
open	/DNAAnalyzer/action/login	
type	name=j_username	usuarioPru
type	name=j_password	X
clickAndWait	id=loginButton	
clickAndWait	link=Administración	
clickAndWait	link=Estudios	
clickAndWait	id=btnCrearEstudio	
type	id=nombre	Selenium
type	id=descripcion	Estudio para las pruebas Selenium
type	id=nombreUsuario	usuarioPru
clickAndWait	id=btnCrearEstudioBuscarUsuarios	
clickAndWait	id=add_0	
clickAndWait	id=btnCrearEstudio	
clickAndWait	css=img.icono	

Y el resultado de su ejecución, en el que vemos que todo ha sido correcto



Command	Target	Value
clickAndWait	id=eliminarSecuencia_0	
type	id=nombre	M11083
clickAndWait	id=btoBuscarSecuencia	
clickAndWait	id=eliminarSecuencia_0	
type	id=nombre	M11083.1
clickAndWait	id=btoBuscarSecuencia	
clickAndWait	id=eliminarSecuencia_0	
type	id=nombre	M11086
clickAndWait	id=btoBuscarSecuencia	
clickAndWait	id=eliminarSecuencia_0	
type	id=nombre	M11086.1
clickAndWait	id=btoBuscarSecuencia	
clickAndWait	id=eliminarSecuencia_0	
clickAndWait	css=img.icono	

9. Resultados

Uno de los objetivos fundamentales del presente proyecto ha sido la construcción de una aplicación sencilla y amigable, que permita realizar los análisis de las secuencias de forma ágil y cómoda.

Para demostrar la eficacia a nivel de tiempos de análisis de la aplicación, vamos a ejecutar dos tipos de pruebas:

- Pruebas de carga sobre distintas longitudes de secuencia, con ventanas de análisis pequeñas, desplazamientos cortos y secuencias de búsqueda cortas.
- Pruebas de tiempos en análisis sobre secuencias consideradas muy largas, en el que se utilizarán distintos tamaños de ventana y desplazamiento y distintos tamaños de secuencias.

Por lo tanto, vamos a poder definir el aguante de la aplicación y el tiempo real que un usuario puede tardar en realizar un análisis de distintos tipos de secuencias, para demostrar la eficacia de la aplicación y que el rendimiento de la misma es bueno.

Para ello, vamos a realizar un análisis de rendimiento. Se trata de una investigación sobre el comportamiento de la aplicación, usando la información que se obtiene desde el análisis dinámico. En este tipo de análisis ejecutamos la aplicación y observamos su comportamiento. Para que este estudio resulte efectivo, el programa debe ejecutar los suficientes casos de prueba como para producir algún tipo de comportamiento interesante.

El objetivo principal de este análisis es confirmar cuales son nuestros puntos fuertes y buscar los puntos débiles, para proponer una estrategia de mejora sobre ellos, ya sea a nivel de velocidad de respuesta o de consumo de recursos.

9.1 Pruebas de Carga

Se realizarán pruebas de carga sobre secuencias medias y grandes siguiendo los siguientes parámetros.

Las secuencias utilizadas en las pruebas son:

- AB027413, con 5710 nucleótidos.
- ADMH02000252 con 70611 nucleótidos

En ambos casos, el tamaño de ventana será 5, 10 o 15 y el desplazamiento de 1 a 4, con secuencias de búsqueda de 2 o 3 nucleótidos, incluyendo además el carácter comodín.

Las pruebas se realizarán también sobre dos tipos de carga de secuencias: descarga desde EMBL-BANK y por fichero.

Además, se incluirá el estudio de secuencias muy cortas, de unos 250 caracteres que hayan sido introducidas por pantalla.

La información contenida en las siguientes tablas tiene el siguiente significado:

- #Muestras: cantidad de Thread utilizados para la URL.
- Media: tiempo promedio en milisegundos para un conjunto de resultados.
- Mediana: valor en tiempo del percentil 50.
- Línea de 90%: máximo tiempo utilizado por el 90% de la muestra, al resto de la misma le llevo más tiempo.
- Línea de 95%: máximo tiempo utilizado por el 95% de la muestra, al resto de la misma le llevo más tiempo.
- Línea de 99%: máximo tiempo utilizado por el 99% de la muestra, al resto de la misma le llevo más tiempo.
- Min: tiempo mínimo de la muestra de una determinada URL en milisegundos
- Max: tiempo máximo de la muestra de una determinada URL en milisegundos
- %Error: porcentaje de requerimientos con errores.
- Rendimiento: rendimiento medido en los requerimiento por segundo / minuto / hora.
- KB/sec: rendimiento medido en Kbytes por segundo.

9.1.1 Caso Base

Se trata de un caso base de uso intensivo de la aplicación, en el que tendremos 5 usuarios concurrentes, que accederán a la aplicación en una rampa de un segundo y que cada uno ejecutará 10 peticiones.

EMBLBANK

Secuencia Media

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	50	57	36	141	142	247	19	247	0,00%	6,9/sec	69,4
Seleccionar ...	50	24	13	50	57	149	8	149	0,00%	7,0/sec	73,4
CargarSecu...	50	208	159	308	418	810	137	810	0,00%	6,9/sec	80,6
Renombrar...	50	173	168	219	220	312	141	312	0,00%	7,6/sec	58,5
Analisis	50	200	73	286	1185	1326	41	1326	0,00%	7,8/sec	236,6
logout	50	17	11	16	55	133	8	133	0,00%	9,3/sec	27,4
Total	300	113	63	197	247	1185	8	1326	0,00%	38,7/sec	472,8

Secuencia Grande

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	50	71	32	216	232	330	19	330	0,00%	2,7/sec	26,7
Seleccionar ...	50	26	11	86	102	132	7	132	0,00%	2,7/sec	27,7
CargarSecu...	50	627	603	816	876	1794	302	1794	0,00%	2,6/sec	26,6
Renombrar...	50	607	638	733	754	839	314	839	0,00%	2,7/sec	26,7
Analisis	50	401	329	684	717	780	115	780	0,00%	2,8/sec	86,8
logout	50	30	11	69	144	231	7	231	0,00%	2,8/sec	8,8
Total	300	294	179	697	754	876	7	1794	0,00%	15,1/sec	176,1

En el caso de la carga de secuencias desde EMBLBank, podemos apreciar que para secuencias más cortas, el tiempo se nos va en CargarSecuencia y en Renombrar, que son los puntos en los que se accede a EMBLBank, en el primero para consultar la descripción de la nueva secuencia y en el segundo para traernos toda la información de la misma. Para saber si ya tenemos una secuencia con ese nombre en el sistema tardamos 137 segundos en el caso de las secuencias cortas y 302 en el de las largas.

Es posible, que este tiempo se deba en parte a la latencia por el acceso por servicio rest a la Base de Datos de EMBLBank y que no podemos controlar su capacidad y ni el tiempo que tarda en servirnos.

La media en el tiempo de análisis se duplica en el caso de la secuencia grande (401) con respecto a la secuencia media (200), aunque se trata de un buen dato, ya que el aumento no es proporcional al tamaño de la secuencia, ya que la secuencia grande tiene más del doble de nucleótidos que la mediana.

Además, el mejor tiempo de análisis para la secuencia media es de 41, siendo para la grande 115. Si resulta significativo que en el caso de la secuencia media, el máximo tiempo empleado en el análisis es 1326 milisegundos, siendo en la secuencia grande tan solo de 780.

FICHERO

Secuencia Media

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	50	60	35	127	132	160	17	160	0,00%	10,3/sec	104,6
Seleccionar ...	50	52	31	123	125	134	7	134	0,00%	10,4/sec	108,7
CargarFiche...	50	43	30	95	99	122	7	122	0,00%	10,4/sec	60,9
Upload	50	40	23	100	104	128	13	128	0,00%	10,4/sec	75,8
CargarSecu...	50	48	23	119	123	127	11	127	0,00%	10,6/sec	81,1
Análisis	50	121	109	184	188	256	38	256	0,00%	10,5/sec	317,2
logout	50	39	19	103	109	127	7	127	0,00%	10,6/sec	31,3
Total	350	57	34	124	149	188	7	256	0,00%	70,7/sec	753,3

Secuencia Grande

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	50	36	25	82	85	120	16	120	0,00%	6,8/sec	68,9
Seleccionar ...	50	16	12	21	43	93	7	93	0,00%	6,8/sec	71,4
CargarFiche...	50	17	13	23	37	107	7	107	0,00%	6,8/sec	40,0
Upload	50	40	28	86	94	104	17	104	0,00%	6,8/sec	44,5
CargarSecu...	50	23	15	60	70	77	8	77	0,00%	6,9/sec	52,8
Análisis	50	374	328	640	676	800	154	800	0,00%	6,5/sec	203,0
logout	50	14	10	32	36	41	6	41	0,00%	6,7/sec	19,8
Total	350	74	20	246	420	676	6	800	0,00%	44,7/sec	477,9

En el caso de la carga de secuencias desde fichero, el tiempo empleado para la comprobación y la carga de la secuencia es muy pequeño, ya que como mínimo tardará 11 milisegundos y como máximo 127.

Al igual que en el caso anterior, para la secuencia grande se duplica el tiempo, aunque para la carga desde fichero el tiempo es menor que en la carga desde EMBLBank. Puede deberse a que al estar más liberado el sistema por no requerir recursos para el acceso rest, los emplee para el análisis.

Aún así los tiempos de análisis tanto para la secuencia media como para la secuencia larga son mejores en la carga por fichero, reduciéndose aproximadamente la media en unos 100 milisegundos en ambos casos.

El tiempo mínimo en un análisis para la secuencia media es de 38 mientras que para la grande es de 154, muy similares a los valores obtenidos en el análisis por carga EMBLBank.

TEXTAREA

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	50	71	82	111	124	159	14	159	0,00%	14,7/sec	148,8
Seleccionar ...	50	42	24	92	107	125	7	125	0,00%	14,8/sec	154,4
CargarSecu...	50	44	27	91	102	138	11	138	0,00%	14,8/sec	111,7
Analisis	50	76	81	119	141	157	20	157	0,00%	14,8/sec	243,6
logout	50	30	15	75	106	128	6	128	0,00%	14,9/sec	44,0
Total	250	53	37	109	123	144	6	159	0,00%	72,2/sec	685,5

En el caso de las secuencias pequeñas, vemos que los tiempos empleados tanto en la carga como en el análisis son mínimos.

9.1.2 Forzar Usuario

Se trata de un caso en el que serán más los usuarios concurrentes que entren a la aplicación y realizarán un mayor número de peticiones:

Entrarán 50 usuario con una rampa de 2 segundos y cada uno de ellos ejecutará 15 peticiones.

EMBLBank

Secuencia Media

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	750	1475	1384	1849	2276	3297	953	3598	0,00%	8,5/sec	85,7
Seleccionar ...	750	394	397	464	472	509	188	567	0,00%	8,8/sec	91,7
CargarSecu...	750	956	935	1138	1203	1356	670	1395	0,00%	8,7/sec	101,7
Renombrar...	750	1577	1572	1854	1909	1993	935	2124	0,00%	8,7/sec	67,1
Analisis	750	882	787	1158	1842	2583	268	2710	0,00%	8,8/sec	263,3
logout	750	644	654	796	834	933	25	1007	0,00%	9,1/sec	26,8
Total	4500	988	895	1652	1806	2337	25	3598	0,00%	49,3/sec	597,5

Secuencia Grande

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	750	1616	1555	2183	2456	2963	865	3390	0,00%	2,4/sec	24,3
Seleccionar ...	750	413	403	531	574	669	153	844	0,00%	2,4/sec	25,1
CargarSecu...	750	8680	8592	9325	9592	9943	7516	10483	0,00%	2,3/sec	18,1
Renombrar...	750	8865	8798	9826	10119	10493	3660	10775	0,00%	2,4/sec	18,4
Analisis	750	1233	1138	1732	2119	2775	612	3688	0,00%	2,4/sec	76,9
logout	750	757	758	994	1098	1348	7	1487	0,00%	2,4/sec	7,2
Total	4500	3594	1326	8982	9363	10085	7	10775	0,00%	13,8/sec	161,7

Al aumentar el número de usuarios concurrentes, los tiempos aumentan, pero sobretodo se ven afectados aquellos puntos en los que se accede de forma remota a la Base de Datos de EMBLBank, ya que vemos que el tiempo medio de acceso aumenta a 8680. El tiempo de análisis también aumenta, siendo la media para las secuencias de tamaño medio de 882 milisegundos y para las largas de 1233.

FICHERO

Secuencia Media

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	750	1108	1184	1438	1517	1713	51	2065	0,00%	10,9/sec	108,9
Seleccionar ...	750	332	356	403	434	487	21	519	0,00%	10,9/sec	114,2
CargarFiche...	750	340	358	403	451	506	16	636	0,00%	10,9/sec	64,0
Upload	750	490	491	640	724	867	21	1167	0,00%	10,9/sec	79,6
CargarSecu...	750	492	486	646	719	918	20	1161	0,00%	11,0/sec	84,3
Analisis	750	685	703	844	903	1445	60	1962	0,00%	11,0/sec	325,1
logout	750	532	587	650	670	753	6	825	0,00%	11,2/sec	33,1
Total	5250	568	493	1083	1248	1502	6	2065	0,00%	75,0/sec	792,3

Secuencia Grande

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	750	1536	1481	2098	2426	3762	19	4587	0,00%	8,3/sec	83,6
Seleccionar ...	750	422	382	564	660	1245	9	1621	0,00%	8,3/sec	86,1
CargarFiche...	750	445	391	634	700	1280	8	1502	0,00%	8,3/sec	48,3
Upload	750	838	659	1259	1704	4186	21	5505	0,00%	8,3/sec	53,7
CargarSecu...	750	542	512	755	849	1328	10	2681	0,00%	8,3/sec	52,0
Analisis	750	1278	1183	2041	2400	4318	14	5987	0,00%	8,2/sec	196,6
logout	750	782	767	1088	1182	1483	5	1865	0,00%	8,3/sec	24,4
Total	5250	835	633	1593	1859	3122	5	5987	0,00%	57,4/sec	542,0

Como en el caso de carga de secuencia desde EMBLBANK, los tiempos aumentan para la carga desde fichero, aunque de una forma mucho más comedida que en el caso anterior.

Al igual que en el caso base y a diferencia de la carga desde EMBLBANK, los tiempos son más altos en análisis, y no en la carga de la secuencia, lo que hace que la máquina se encuentre más liberada y no haya problemas de rendimiento.

Podemos ver igualmente, que el tiempo medio empleado en el análisis de secuencias de tamaño medio se reduce en la carga por fichero en unos 200 milisegundos, mientras que para el caso de las secuencias de gran tamaño se mantiene similar en los dos tipos de carga.

TEXTAREA

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	750	1295	1279	1691	1841	2121	16	2357	0,00%	13,7/sec	138,9
Seleccionar ...	750	354	377	418	426	452	7	493	0,00%	13,7/sec	143,2
CargarSecu...	750	570	529	783	905	1146	16	1271	0,00%	13,7/sec	103,8
Analisis	750	708	669	911	1042	1303	34	1459	0,00%	13,7/sec	204,2
logout	750	552	549	667	682	699	6	746	0,00%	13,7/sec	40,5
Total	3750	696	604	1286	1474	1845	6	2357	0,00%	68,4/sec	628,8

Al forzar el número de usuarios, los tiempos aumentan ligeramente, pero para el caso de secuencias cortas, siguen siendo muy buenos.

9.1.3 Forzar Peticiones

Se trata del caso en el que se realizan peticiones de forma masiva a la aplicación, elevando tanto el número de usuario del caso base como el de peticiones:

Entrarán 25 usuarios con una rampa de 2 segundos y cada uno de ellos ejecutará 75 peticiones.

EMBLBANK

Secuencia Media

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1875	457	460	569	583	624	23	1735	0,00%	9,2/sec	93,6
Seleccionar ...	1875	178	215	239	246	261	9	491	0,00%	9,2/sec	96,6
CargarSecu...	1875	675	687	808	833	882	146	2256	0,00%	9,2/sec	71,2
Renombrar...	1875	674	689	804	827	879	136	1872	0,00%	9,3/sec	71,5
Analisis	1875	397	355	477	493	656	45	2319	0,00%	9,3/sec	275,2
logout	1875	246	229	348	359	381	6	588	0,00%	9,4/sec	27,7
Total	11250	438	443	716	790	847	6	2319	0,00%	55,3/sec	631,5

Secuencia Grande

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1875	516	542	583	595	672	14	1620	0,00%	2,8/sec	28,8
Seleccionar ...	1875	193	215	234	237	251	6	381	0,00%	2,8/sec	29,7
CargarSecu...	1875	3549	3563	3725	3777	3861	330	4074	0,00%	2,8/sec	22,0
Renombrar...	1875	3555	3580	3736	3798	3925	219	4046	0,00%	2,8/sec	22,0
Analisis	1875	632	592	851	917	1101	296	1466	0,00%	2,9/sec	90,0
logout	1875	254	231	342	353	370	6	490	0,00%	2,9/sec	8,4
Total	11250	1450	555	3644	3702	3821	6	4074	0,00%	17,0/sec	200,0

Comparando con el caso de forzar usuarios, observamos que el aumento del número de peticiones no degrada los tiempos ni en análisis ni en carga de datos, ya que incluso son mejores que en el estudio anterior.

El tiempo medio empleado en el análisis para las secuencias medias es de 397 y para las largas de 632, lo que prácticamente es la mitad del tiempo empleado en el caso de forzar usuarios. Es prácticamente el doble del tiempo empleado en el análisis para el caso base.

FICHERO

Secuencia Media

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1875	508	517	669	688	715	16	828	0,00%	11,1/sec	112,7
Seleccionar ...	1875	215	238	274	287	352	7	404	0,00%	11,1/sec	116,2
CargarFiche...	1875	213	239	274	286	353	10	416	0,00%	11,1/sec	65,1
Upload	1875	270	257	384	405	444	19	656	0,00%	11,1/sec	72,5
CargarSecu...	1875	235	244	282	288	370	10	452	0,00%	11,1/sec	85,7
Analisis	1875	412	397	534	556	649	44	813	0,00%	11,1/sec	327,5
logout	1875	286	266	400	420	450	5	549	0,00%	11,1/sec	32,9
Total	13125	306	261	509	574	679	5	828	0,00%	77,7/sec	811,5

Secuencia Grande

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1875	572	588	752	798	988	13	1518	0,00%	8,9/sec	90,7
Seleccionar ...	1875	207	225	306	363	432	6	615	0,00%	8,9/sec	93,5
CargarFiche...	1875	222	242	327	363	475	7	732	0,00%	8,9/sec	52,4
Upload	1875	335	287	449	497	1602	18	2758	0,00%	8,9/sec	58,4
CargarSecu...	1875	249	251	356	391	477	7	613	0,00%	8,9/sec	64,1
Analisis	1875	721	661	1078	1269	2507	8	3129	0,00%	8,9/sec	253,8
logout	1875	306	288	440	473	533	6	732	0,00%	8,9/sec	26,4
Total	13125	373	283	684	800	1250	6	3129	0,00%	62,5/sec	638,2

Como en el caso anterior, los tiempos mejoran para la carga por fichero en el caso de forzar peticiones con respecto a forzar usuarios.

El tiempo de análisis es muy similar al de carga por EMBL BANK. En este caso, si los comparamos con los tiempos del caso base, que eran 121 para secuencias medias y 374 para secuencias grandes, podemos ver que la media es más alta, dándonos 412 para secuencias de tamaño medio y 721 para secuencias largas.

TEXTAREA

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1875	515	523	645	688	726	14	820	0,00%	14,4/sec	146,0
Seleccionar ...	1875	216	229	271	284	349	9	412	0,00%	14,4/sec	150,4
CargarSecu...	1875	281	255	377	406	444	12	630	0,00%	14,4/sec	109,4
Análisis	1875	378	365	490	522	574	23	683	0,00%	14,4/sec	213,4
logout	1875	293	274	395	413	448	5	508	0,00%	14,4/sec	42,5
Total	9375	337	333	538	590	688	5	820	0,00%	71,9/sec	661,3

En cualquier caso, para el uso de secuencias cortas, los tiempos siguen siendo muy buenos, notándose igualmente que si forzamos peticiones los tiempos de análisis se reducen, a algo más de la mitad, aunque quedan algo lejos del caso base.

9.1.4 Resumen de resultados

Tiempos Carga en MS

Caso Base

	Media	Grande
EMBL BANK	173	607
FICHERO	48	23

Forzar Usuarios

	Media	Grande
EMBL BANK	1577	8865
FICHERO	492	542

Forzar Peticiones

	Media	Grande
EMBL BANK	674	3555
FICHERO	235	249

Tiempos Análisis en MS

Caso Base

	Media	Grande
EMBL BANK	200	401
FICHERO	121	374

Forzar Usuarios

	Media	Grande
EMBL BANK	882	1233
FICHERO	685	1278

Forzar Peticiones

	Media	Grande
EMBL BANK	397	632
FICHERO	412	721

9.2 Pruebas sobre Análisis

Tras las pruebas de carga, vamos a realizar un análisis de una secuencia considerada muy larga: 33282659 nucleótidos, para evaluar el tiempo que tarda en realizarse un análisis con distintos parámetros.

Para todos los casos vamos a utilizar el mismo tipo de carga de secuencia, por fichero, para que de esta forma, afecte lo menos posible a los tiempos en análisis.

Ventana pequeña (10) con desplazamiento corto (3) y secuencia corta (ATG)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	933	933	933	933	933	933	933	0,00%	1,1/sec	7,1
Seleccionar...	1	46	46	46	46	46	46	46	0,00%	21,7/sec	226,9
CargarFiche...	1	27	27	27	27	27	27	27	0,00%	37,0/sec	216,3
Upload	1	1517	1517	1517	1517	1517	1517	1517	0,00%	39,6/min	4,8
CargarSecu...	1	367	367	367	367	367	367	367	0,00%	2,7/sec	21,1
Análisis	1	39389	39389	39389	39389	39389	39389	39389	0,00%	1,5/min	,8
logout	1	53	53	53	53	53	53	53	0,00%	18,9/sec	55,8
Total	7	6047	367	1517	39389	39389	27	39389	0,00%	9,9/min	1,7

El tiempo total del análisis es de 39389 milisegundos, que en segundos es de 39,39.

Ventana pequeña (10), con desplazamiento corto (3) y secuencia larga (ATGCACGTA)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	66	66	66	66	66	66	66	0,00%	15,2/sec	99,8
Seleccionar...	1	17	17	17	17	17	17	17	0,00%	58,8/sec	613,9
CargarFiche...	1	16	16	16	16	16	16	16	0,00%	62,5/sec	365,0
Upload	1	1347	1347	1347	1347	1347	1347	1347	0,00%	44,5/min	5,4
CargarSecu...	1	286	286	286	286	286	286	286	0,00%	3,5/sec	27,0
Análisis	1	9980	9980	9980	9980	9980	9980	9980	0,00%	6,0/min	1,3
logout	1	34	34	34	34	34	34	34	0,00%	29,4/sec	86,9
Total	7	1678	66	1347	9980	9980	16	9980	0,00%	35,7/min	4,6

El tiempo total del análisis es de 9980 milisegundos, que en segundos es de 9,99.

Ventana pequeña (10), con desplazamiento corto (3) y secuencia con comodines (A?T?G)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	61	61	61	61	61	61	61	0,00%	16,4/sec	108,0
Seleccionar...	1	17	17	17	17	17	17	17	0,00%	58,8/sec	613,9
CargarFiche...	1	16	16	16	16	16	16	16	0,00%	62,5/sec	365,0
Upload	1	1588	1588	1588	1588	1588	1588	1588	0,00%	37,8/min	4,6
CargarSecu...	1	283	283	283	283	283	283	283	0,00%	3,5/sec	27,3
Análisis	1	28906	28906	28906	28906	28906	28906	28906	0,00%	2,1/min	1,1
logout	1	47	47	47	47	47	47	47	0,00%	21,3/sec	62,9
Total	7	4416	61	1588	28906	28906	16	28906	0,00%	13,6/min	2,3

El tiempo total del análisis es de 28906 milisegundos, que en segundos es de 28,91.

Por lo tanto, tras ver el análisis con ventana pequeña y desplazamiento corto de los distintos tipos de secuencia, afirmamos que es más rápido el análisis cuando se busca una secuencia larga de nucleótidos. Respecto a secuencias de búsqueda cortas, podemos afirmar que el uso de los comodines influye de manera positiva, ya que el tiempo de búsqueda se reduce en algo más de 10 segundos.

Ventana media (100), con desplazamiento medio (10) y secuencia corta (ATG)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	949	949	949	949	949	949	949	0,00%	1,1/sec	6,9
Seleccionar ...	1	52	52	52	52	52	52	52	0,00%	19,2/sec	200,7
CargarFiche...	1	31	31	31	31	31	31	31	0,00%	32,3/sec	188,4
Upload	1	1559	1559	1559	1559	1559	1559	1559	0,00%	38,5/min	4,7
CargarSecu...	1	308	308	308	308	308	308	308	0,00%	3,2/sec	25,1
Análisis	1	34038	34038	34038	34038	34038	34038	34038	0,00%	1,8/min	1,2
logout	1	68	68	68	68	68	68	68	0,00%	14,7/sec	43,5
Total	7	5286	308	1559	34038	34038	31	34038	0,00%	11,3/min	2,2

El tiempo total del análisis es de 34038 milisegundos, que en segundos es de 34,04.

Ventana media (100), con desplazamiento medio (10) y secuencia larga (ATGCACGTA)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	64	64	64	64	64	64	64	0,00%	15,6/sec	103,0
Seleccionar ...	1	17	17	17	17	17	17	17	0,00%	58,8/sec	613,9
CargarFiche...	1	16	16	16	16	16	16	16	0,00%	62,5/sec	365,0
Upload	1	1443	1443	1443	1443	1443	1443	1443	0,00%	41,5/min	5,1
CargarSecu...	1	367	367	367	367	367	367	367	0,00%	2,7/sec	21,1
Análisis	1	5225	5225	5225	5225	5225	5225	5225	0,00%	11,5/min	5,3
logout	1	39	39	39	39	39	39	39	0,00%	25,6/sec	75,8
Total	7	1024	64	1443	5225	5225	16	5225	0,00%	58,5/min	9,5

El tiempo total del análisis es de 5225 milisegundos, que en segundos es de 5,23.

Ventana media (100), con desplazamiento medio (10) y secuencia con comodines (A?T?G)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	62	62	62	62	62	62	62	0,00%	16,1/sec	106,3
Seleccionar ...	1	17	17	17	17	17	17	17	0,00%	58,8/sec	613,9
CargarFiche...	1	16	16	16	16	16	16	16	0,00%	62,5/sec	365,0
Upload	1	1434	1434	1434	1434	1434	1434	1434	0,00%	41,8/min	5,1
CargarSecu...	1	322	322	322	322	322	322	322	0,00%	3,1/sec	24,0
Análisis	1	33746	33746	33746	33746	33746	33746	33746	0,00%	1,8/min	1,2
logout	1	38	38	38	38	38	38	38	0,00%	26,3/sec	77,8
Total	7	5090	62	1434	33746	33746	16	33746	0,00%	11,8/min	2,3

El tiempo total del análisis es de 33746 milisegundos, que en segundos es de 33,75.

Al igual que en el caso anterior, vemos que el uso de una secuencia de búsqueda larga hace que el tiempo de análisis se reduzca con respecto a los otros dos casos. Para el caso del tamaño de ventana medio, no hay tanta diferencia en el uso de comodines en la secuencia de búsqueda, la diferencia de tiempo no llega a un segundo.

Además, hay una ligera mejora en todos los tiempos, que se ven reducidos en el uso de una ventana más grande y un desplazamiento algo mayor.

Ventana grande (200), con desplazamiento medio (10) y secuencia corta (ATG)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	932	932	932	932	932	932	932	0,00%	1,1/sec	7,1
Seleccionar ...	1	46	46	46	46	46	46	46	0,00%	21,7/sec	226,9
CargarFiche...	1	27	27	27	27	27	27	27	0,00%	37,0/sec	216,3
Upload	1	1612	1612	1612	1612	1612	1612	1612	0,00%	37,2/min	4,5
CargarSecu...	1	302	302	302	302	302	302	302	0,00%	3,3/sec	25,6
Analisis	1	66965	66965	66965	66965	66965	66965	66965	100,00%	53,8/hour	,1
logout	1	65	65	65	65	65	65	65	0,00%	15,4/sec	45,5
Total	7	9992	302	1612	66965	66965	27	66965	14,29%	6,0/min	,7

El tiempo total del análisis es de 66965 milisegundos, que en segundos es de 66,97.

Ventana grande (200), con desplazamiento medio (10) y secuencia larga (ATGCACGTA)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	61	61	61	61	61	61	61	0,00%	16,4/sec	108,0
Seleccionar ...	1	17	17	17	17	17	17	17	0,00%	58,8/sec	613,9
CargarFiche...	1	18	18	18	18	18	18	18	0,00%	55,6/sec	324,4
Upload	1	1380	1380	1380	1380	1380	1380	1380	0,00%	43,5/min	5,3
CargarSecu...	1	323	323	323	323	323	323	323	0,00%	3,1/sec	23,9
Analisis	1	5689	5689	5689	5689	5689	5689	5689	0,00%	10,5/min	8,8
logout	1	36	36	36	36	36	36	36	0,00%	27,8/sec	82,1
Total	7	1074	61	1380	5689	5689	17	5689	0,00%	55,7/min	12,1

El tiempo total del análisis es de 5689 milisegundos, que en segundos es de 5.70.

Ventana grande (200), con desplazamiento medio (10) y secuencia con comodines (A?T?G)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	61	61	61	61	61	61	61	0,00%	16,4/sec	108,0
Seleccionar ...	1	19	19	19	19	19	19	19	0,00%	52,6/sec	549,2
CargarFiche...	1	17	17	17	17	17	17	17	0,00%	58,8/sec	343,5
Upload	1	1363	1363	1363	1363	1363	1363	1363	0,00%	44,0/min	5,4
CargarSecu...	1	283	283	283	283	283	283	283	0,00%	3,5/sec	27,3
Analisis	1	69836	69836	69836	69836	69836	69836	69836	100,00%	51,5/hour	,1
logout	1	44	44	44	44	44	44	44	0,00%	22,7/sec	67,2
Total	7	10231	61	1363	69836	69836	17	69836	14,29%	5,9/min	,7

El tiempo total del análisis es de 69836 milisegundos, que en segundos es de 69.84.

Como en los casos anteriores, la búsqueda más rápida es para la secuencia de búsqueda larga. En este caso, si vemos que el uso de comodines perjudica la búsqueda, ya que tarda casi dos segundos más.

Con respecto al uso de una ventana de tamaño medio vemos que no hay casi variación en el tiempo de búsqueda de secuencias largas. Por el contrario, el tiempo de búsqueda de secuencias cortas o con comodín se ve realmente afectado, llegando a duplicarse en ambos casos.



Ventana grande (200), con desplazamiento largo (100) y secuencia corta (ATG)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	549	549	549	549	549	549	549	0,00%	1,8/sec	12,0
Seleccionar ...	1	53	53	53	53	53	53	53	0,00%	18,9/sec	196,9
CargarFiche...	1	32	32	32	32	32	32	32	0,00%	31,2/sec	182,5
Upload	1	1470	1470	1470	1470	1470	1470	1470	0,00%	40,8/min	5,0
CargarSecu...	1	277	277	277	277	277	277	277	0,00%	3,6/sec	27,9
Analisis	1	6962	6962	6962	6962	6962	6962	6962	0,00%	8,6/min	7,2
logout	1	35	35	35	35	35	35	35	0,00%	28,6/sec	84,4
Total	7	1339	277	1470	6962	6962	32	6962	0,00%	44,7/min	9,7

El tiempo total del análisis es de 6962 milisegundos, que en segundos es de 6.96.

Ventana grande (200), con desplazamiento largo (100) y secuencia larga (ATGCACGTA)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	56	56	56	56	56	56	56	0,00%	17,9/sec	117,7
Seleccionar ...	1	16	16	16	16	16	16	16	0,00%	62,5/sec	652,2
CargarFiche...	1	14	14	14	14	14	14	14	0,00%	71,4/sec	417,1
Upload	1	1273	1273	1273	1273	1273	1273	1273	0,00%	47,1/min	5,7
CargarSecu...	1	260	260	260	260	260	260	260	0,00%	3,8/sec	29,7
Analisis	1	3111	3111	3111	3111	3111	3111	3111	0,00%	19,3/min	10,7
logout	1	42	42	42	42	42	42	42	0,00%	23,8/sec	70,4
Total	7	681	56	1273	3111	3111	14	3111	0,00%	1,5/sec	15,5

El tiempo total del análisis es de 3111 milisegundos, que en segundos es de 3.11.

Ventana grande (200), con desplazamiento largo (100) y secuencia con comodines. (A?T?G)

Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimiento	Kb/sec
Login	1	56	56	56	56	56	56	56	0,00%	17,9/sec	117,7
Seleccionar ...	1	14	14	14	14	14	14	14	0,00%	71,4/sec	745,4
CargarFiche...	1	15	15	15	15	15	15	15	0,00%	66,7/sec	389,3
Upload	1	1281	1281	1281	1281	1281	1281	1281	0,00%	46,8/min	5,7
CargarSecu...	1	250	250	250	250	250	250	250	0,00%	4,0/sec	30,9
Analisis	1	7538	7538	7538	7538	7538	7538	7538	0,00%	8,0/min	6,6
logout	1	38	38	38	38	38	38	38	0,00%	26,3/sec	77,8
Total	7	1313	56	1281	7538	7538	14	7538	0,00%	45,6/min	9,9

El tiempo total del análisis es de 7538 milisegundos, que en segundos es de 7.54.

Para el caso de las ventanas grandes con desplazamiento largo, vemos que la búsqueda de secuencias largas es la más eficaz. Con respecto al uso de secuencias cortas, el uso de comodines perjudica levemente el tiempo de análisis, ya que aumenta en medio segundo con respecto al análisis que no los usa.

En comparación con el tiempo de análisis del resto de tamaños de ventana y desplazamiento, el uso de ventanas grandes y desplazamientos largos beneficia el análisis de la secuencia, reduciendo los tiempos a la décima parte en relación a la búsqueda en ventanas de tamaño medio y la quinta parte en las secuencias pequeñas.

10. Conclusiones

Tras las pruebas realizadas podemos llegar a la conclusión de que la aplicación es robusta y está preparada para soportar una gran carga de datos, análisis y usuarios concurrentes.

Cabe destacar el buen rendimiento obtenido para secuencias cortas, medias y largas, en el que los tiempos de carga y análisis son muy reducidos, facilitando el trabajo y el uso de la aplicación, reduciendo así los tiempos de espera del usuario.

Aun así, y aunque los tiempos de respuesta no son malos, sería conveniente mejorar aún más el rendimiento de la aplicación para el posible uso de secuencias muy largas, como en el caso de la utilizada para el estudio de los tiempos de análisis con distintos parámetros.

Para la carga de datos, resulta recomendable utilizar secuencias que ya se encuentren cargadas en el sistema o realizar la carga desde un fichero ubicado en el ordenador del usuario. Al cargar las secuencias utilizando el servicio REST de la Base de Datos EMBL-BANK, nos vemos expuestos a sus tiempos de respuesta, que aunque no son malos, son superiores a los del resto de tipos de carga y aumentan para las secuencias más largas.

Por otro lado, en el estudio de tiempos de análisis, vemos que la aplicación funciona mejor con ventanas largas, desplazamientos largos y secuencias de búsqueda largas. Esto es lo esperado, ya que con estos tipos de análisis se reducen tanto la cantidad de muestras obtenidas (ventanas más grandes y mayores desplazamientos, generan menos ventanas que ventanas cortas y desplazamientos cortos), como la cantidad de veces que encontramos la secuencia buscada en cada ventana.

Otro de los puntos a destacar es que el uso del carácter comodín no penaliza especialmente los tiempos de respuesta, lo que permite hacer análisis más variados y menos específicos, algo que puede ser beneficioso para el usuario.

Finalmente y a nivel funcional, se aportan todas las funcionalidades básicas requeridas en la aplicación, dando la posibilidad de agrupar en estudios a los usuarios y guardar de forma ordenada los resultados de los distintos análisis obtenidos.

11. Futuras líneas de trabajos

Podemos separar las mejoras y futuras líneas de trabajo en dos puntos muy diferenciados: las mejoras a nivel funcional y las mejoras a nivel de arquitectura:

Mejoras a nivel funcional

Tras la implementación y el estudio con el usuario del uso de la aplicación, se comprende más a fondo cuáles son sus necesidades reales o que puntos pueden aportarle valor a la aplicación. En ocasiones son situaciones que el propio usuario no ha valorado, no ve posible o inicialmente no considera que puedan serle útiles.

De entre todas las mejoras que hemos visto y que se han tratado con el usuario y ha valorado como positivas para posibles evoluciones, destacamos las siguientes:

- Comprobación de si un análisis sobre una secuencia ya existe dentro del estudio. De ésta forma, evitamos al usuario repetir estudios de forma innecesaria estudios que ya ha realizado.
- Posibilidad de realizar el estudio off-line, de forma que el usuario no tenga que esperar el resultado inmediato. Esto puede ser utilizado para las secuencias más largas, dando la posibilidad al sistema de realizar el análisis en momentos de menos carga (por la noche, por ejemplo). Se enviaría un correo al usuario con acceso directo al resultado del análisis requerido.
- Dar la posibilidad al usuario de elegir el idioma en el que desea trabajar en la aplicación, sin necesidad de modificar el idioma del navegador. Actualmente, además de haciéndolo en el navegador, puede conseguirse una vez dentro de la aplicación añadiendo a la URL ?locale=es o ?locale=en, que lo cambiará al español o al inglés. Pero, resulta evidente que esto no es cómodo ni intuitivo para el usuario.
- Posibilidad de implementar búsquedas de secuencias ya utilizadas en el sistema que cumplan determinadas condiciones, como por ejemplo que contengan determinada secuencia de nucleótidos.
- Posibilidad de implementar búsquedas sobre los resultados de análisis realizados que cumplan determinadas condiciones o que se hayan realizado utilizando parámetros idénticos o similares sobre la misma o distintas secuencias.
- Extender el uso de la aplicación, para aplicarlo al estudio de secuencias de proteínas, en la que existen 20 aminoácidos en lugar de 4 nucleótidos.

Mejoras a nivel de arquitectura

A nivel de arquitectura sobretodo cabe destacar la posibilidad de mejorar los tiempos de análisis, con la implementación de un sistema en multihilo, capaz de fragmentar la secuencia y realizar análisis en paralelo.



12. Bibliografía y Referencias

- [1] – Bases de Datos Biológica : http://es.wikipedia.org/wiki/Base_de_datos_biol%C3%B3gica
- [2] – EMBL-BANK: <http://www.ebi.ac.uk/ena>
- [3] – DNA Data Bank of Japan: <http://www.ddbj.nig.ac.jp/>
- [4] – GENBANK: <http://www.ncbi.nlm.nih.gov/nucleotide>
- [5] – Aplicación SeeDNA: <http://www.ncbi.nlm.nih.gov/pubmed/15862120>
- [6] – K-MER: <https://en.wikipedia.org/wiki/K-mer>
- [7] – Next-generation DNA Sequencing
<http://www.nature.com/nbt/journal/v26/n10/full/nbt1486.html>
- [8] – Selenium: <http://www.seleniumhq.org/>
- [9] – JMETER: <http://jmeter.apache.org/>
- [10] – Profiling: https://es.wikipedia.org/wiki/An%C3%A1lisis_de_rendimiento_de_software
- [11] – Badboy: <http://www.badboy.com.au/>
- [12] - Metodología ágil: https://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software
- [13] – Scrum: <http://www.proyectosagiles.org/que-es-scrum>
- [14] – Raspberry Pi: <https://www.raspberrypi.org/>
- [15] - Spring Framework: <http://projects.spring.io/spring-framework/>
- [16] – Maven: <https://maven.apache.org/>
- [17] – Tiles: <https://tiles.apache.org/>
- [18] – AmCharts: <http://www.amcharts.com/>
- [19] MongoDB: <https://www.mongodb.org/>



13. Anexos

Anexo 1 – Documentación Sprint 1

Anexo 2 – Documentación Sprint 2

Anexo 3 – Documentación Sprint 3

Anexo 4 – Documentación Sprint 4

Anexo 5 – Manual de Usuario