

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computación
Aseguramiento de la Calidad del Software
M. Sc. Saúl Ramírez Calderón

Avance 2 - Proyecto Semestral

Pablo Navarro
2014003728

Ariel Rodríguez
2014053647

Kevin Ramírez
2014117104

27 abril, 2017

I Semestre 2017

Índice

1. Documento de Requerimientos Basado en el Estándar ISO/IE-C/IEEE 29148-2011	4
1.1. Propósito del sistema	4
1.2. Alcance del sistema	4
1.3. Resumen del sistema	4
1.3.1. Contexto del sistema	4
1.3.2. Funciones del sistema	5
1.3.3. Características del usuario	5
1.4. Requerimientos funcionales	6
1.5. Requerimientos de usabilidad	7
1.6. Requerimientos de rendimiento	7
1.7. Interfaces del sistema	7
1.8. Operaciones del sistema	8
1.8.1. Requerimientos de integración del sistema humano . .	8
1.8.2. Mantenibilidad	8
1.8.3. Confiabilidad	8
1.9. Modos del sistema y estados	9
1.10. Características físicas	9
1.10.1. Requerimientos físicos	9
1.10.2. Requerimientos de adaptabilidad	9
1.11. Condiciones del ambiente	9
1.12. Seguridad del sistema	10
1.13. Administración de la información	10
1.14. Políticas y regulaciones	10
1.15. Sostenibilidad del ciclo de vida del software	11
1.16. Empaquetado, manejo, envío y transporte	11
1.17. Verificación	11
1.18. Asunciones y dependencias	11
2. Validación del diseño	12
3. Validación de la implementación	13
4. Detalles de herramientas de estándares de codificación	14
5. Obtención de la versión actual del sistema	14

6. Métricas	18
7. Especificación de pruebas unitarias	19

1. Documento de Requerimientos Basado en el Estándar ISO/IEC/IEEE 29148-2011

1.1. Propósito del sistema

La necesidad de crear procesos más eficientes en la medicina se ha vuelto una prioridad debido a las incrementales necesidades de la sociedad, para esto es necesario tener sistemas que mejoren y/o agilicen tareas repetitivas en busca de algún resultado importante. El Laboratorio de Quimiosensibilidad Tumoral de la Universidad de Costa Rica, donde los doctores Steve Quirós y Rodrigo Mora desarrollan proyectos de investigación relacionados con la terapia personalizada para pacientes con cáncer, requieren de alguna herramienta que les facilite tareas como el conteo de células en una foto y hasta el seguimiento de la descendencia de algunas células. Esto mediante el uso de fotografías capturadas por sistemas propios del laboratorio.

1.2. Alcance del sistema

El sistema de análisis de muestras celulares, será capaz de:

1. Segmentación a partir de imágenes capturas de tejido canceroso, mediante el microscopio basado en fluorescencia.
2. Distinción del fondo y los objetos de interés, en este caso las células.
3. Etiquetar los objetos mediante un nombre.
4. El sistema puede ser accedido desde cualquier computador del laboratorio con un navegador web y acceso al servidor principal.
5. Será capaz de seleccionar archivos (fotos de las células), seleccionar entre distintos procesamiento sobre las imágenes y obtener los resultados de distintas maneras, ya sea como una imagen dentro de la página web o un archivo guardado en el computador.

1.3. Resumen del sistema

1.3.1. Contexto del sistema

El sistema está destinado a servir a biólogos con conocimientos básicos en computación. El sistema se basa en una aplicación web, con acceso a un

servidor encargado de realizar los procesos sobre las imágenes y dar los resultados. Los biólogos no solamente estarán encargados de incluir las imágenes en el sistema y para un mejor entendimiento, ellos son los encargados de escoger las muestras de células adecuadas, capturar fotos del comportamiento por un lapso de alrededor de 72 horas cada 10 minutos y analizar dicho lote de fotos en el sistema.

Los métodos para el análisis de imágenes son muy variados por ejemplo la microscopía de fluorescencia y la citometría de flujo, según sea el método, se obtienen distintos datos. En el caso del sistema y para nuestro interés, el microscopio de fluorescencia es el encargado de obtener las imágenes a procesar por el sistema. Luego de haber procesado las imágenes, el biólogo encargado será capaz de visualizar en mejor manera los objetos de interés, cada uno con una respectiva etiqueta y crear estadísticas sobre los datos obtenidos.

1.3.2. Funciones del sistema

Carga de lotes de imágenes o imágenes en particular para su futuro análisis mediante los distintos algoritmos que se carguen en el servidor. Para esto es necesario tener previamente el lote creado a partir de uno de los distintos métodos de obtención de muestras, por ejemplo, el microscopio de fluorescencia. El resultado del análisis será un conjunto de imágenes tratadas por los procesos del sistema, junto con un archivo tablas para identificar ciertos atributos y características de la muestra.

1.3.3. Características del usuario

Los usuarios principales del sistema son los biólogos del laboratorio, con poco conocimiento en computación, pero basto en el área de su interés. Debido a esta limitante es necesario que la aplicación que se vaya a desarrollar sea de fácil acceso y de rápido entendimiento para cualquier colaborador del proyecto.

Para el mantenimiento del programa es necesario de desarrolladores con conocimiento básico en las tareas que se requieren. Para esto es necesario de una pequeña explicación de los procesos necesarios para el análisis de resultados sobre muestras de células y tejidos.

1.4. Requerimientos funcionales

La aplicación debe proveer una interfaz gráfica de usuario amigable, que permita (por orden de prioridad):

1. Cargar una secuencia de imágenes almacenada en la dirección provista por el usuario.
2. Segmentar las imágenes en las regiones de fondo y no fondo.
 - a) Los resultados deben ser almacenados en archivos.
3. Calcular métrica de precisión de la segmentación o clasificación:
 - a) Para el caso de implementarse la segmentación, usar una métrica de segmentación como el índice de Dice o de Jaccard.

$$D = \frac{2|S \cap G|}{|S + G|}$$

donde S es la región automáticamente segmentada, G es la región manualmente segmentada, y el valor absoluto se refiere a la cantidad de pixeles en tales regiones.

4. Detectar y contar los objetos de interés en las imágenes (células), de forma que sea fácil de visualizar desde la interfaz gráfica.
 - a) Opcionalmente colorear y etiquetar cada objeto.
5. Genere un informe que indique por cada objeto encontrado, su centroide y su área, en formato .csv.
6. Permitir la navegación entre imágenes y resultados.
 - a) Debe permitirse el procesamiento de un lote de imágenes en una sola acción.
7. Visualizar el tiempo de ejecución por imagen y por el lote de imágenes.
8. Todas las funcionalidades deben ser accesibles desde la interfaz web del sistema.

1.5. Requerimientos de usabilidad

Debido a que se trata de un sistema dirigido a biólogos no expertos en computación, se debe hacer lo más simple posible, además que está dirigido para personal colaborador en el proyecto. Por esta causa se debe tener un alto nivel de entendibilidad y aprendibilidad en el caso de nuevos integrantes, esto con el fin de reducir el tiempo en tareas innecesarias como aprender a usar la herramienta, ajustando parámetros, buscando funciones y enfocarse en lo que realmente es necesario.

Se requiere de una interfaz amigable con el usuario, para hacer más efectivo el uso del software. La cantidad de tiempo requerida en usar una función del software, por ejemplo, la subida de imágenes no debe tomar más de 3 clicks, desde el momento en que se abre la aplicación hasta seleccionar las imágenes.

El acceso a los procesos sobre las imágenes debe ser completamente visible al usuario y debe por sí mismo dar a entender dónde están los procesos, esto es, diseñar una interfaz con la posibilidad inmediata luego de subir las imágenes, poder tratarlas.

Las direcciones donde se almacenan las imágenes deben ser fácil y completamente accesibles por el usuario, debido a que son resultados críticos en la investigación. Una ruta no debe contener caracteres poco comunes y especialmente debe estar compuesta por nombre significativos.

1.6. Requerimientos de rendimiento

La cantidad de imágenes a analizar por lote por el sistema puede superar la cantidad de miles, esto quiere decir que se requiere de minimizar el tiempo lo más posible para obtener un resultado concreto y aun factible. El análisis de una imagen debe tomar como máximo 1 minuto y 30 segundos.

El acceso a la aplicación debe tomar no más de 10 segundos entre que se intenta ingresa al navegador web y se obtiene el sitio.

La generación de los archivos .csv, deben ser inmediatamente después de haber analizado la imagen o lote de imágenes.

1.7. Interfaces del sistema

Entre las interfaces más importantes del sistema se encuentra la interfaz gráfica, el principal medio de comunicación con el usuario. Esta interfaz debe

ser simple, completa y accesible por cualquier colaborador del proyecto.

Debido a que es una aplicación web, es necesario de un cliente y algún servidor, con sus respectivos métodos de comunicación.

1.8. Operaciones del sistema

1.8.1. Requerimientos de integración del sistema humano

El sistema requiere de poca atención durante sus operaciones, además de las acciones que requiere para subir, procesar y descargar las imágenes, solo están las labores de monitoreo y puesta a punto por parte del equipo de mantenimiento del sistema.

1.8.2. Mantenibilidad

Debido a la necesidad para la obtención de mejores resultados en los procesos, es necesario la modificación de estos, por ello el software requerirá de una constante modificación o adición de algoritmos, esto según sea la necesidad del usuario. Debido a que son algoritmos independientes y a la modularidad del software, se prevé que los mantenimientos para la adición de software o modificación no tarde más de 3 días

1.8.3. Confiabilidad

El uso constante no es una prioridad en este caso, debido a esto el tiempo funcional del software puede variar entre 90 y 98 % del total, para cumplir con la cantidad de horas se puede optar por un servidor en la nube. Al ser un servidor y de acuerdo a las horas de trabajo del equipo de investigación del laboratorio, el mantenimiento y otras actividades para cumplir con los requisitos de confiabilidad están pensadas para suceder durante horas no-laborales.

Las estaciones (computadoras) accesibles por los investigadores y colaboradores, deben poseer conexión a internet para poder tener acceso al servidor. Estas estaciones deben recibir un mantenimiento regular, no muy intensivo y poseer as últimas actualizaciones del sistema operativo y el navegador web.

El almacenamiento en la computadora dónde los usuarios operen debe tener al menos 10GB disponible.

1.9. Modos del sistema y estados

El sistema va a operar en un solo modo de procesamiento de imágenes, donde estas van a entrar, ser seleccionadas y puestas a ser analizadas por los distintos algoritmos que sean codificados en el sistema en el lado del servidor.

1.10. Características físicas

1.10.1. Requerimientos físicos

El lugar donde se encuentren los computadores, no requiere de ninguna construcción especial, pero es necesario que cumpla con los requisitos descritos en la 1.11 de este documento.

1.10.2. Requerimientos de adaptabilidad

El software demanda de mucho poder computacional, debido al análisis constante y exhaustivo de imágenes, para solventar la constante demanda de recursos es necesario un sistema escalable, tanto vertical como horizontalmente, esto quiere decir que las tareas por cumplir no solo necesitan de un servidor muy potente, sino de varios trabajando en conjunto. La solución para esto es el uso de un sistema almacenado en la nube.

El tráfico de datos puede variar increíblemente, especialmente por la cantidad de imágenes y para obtener una respuesta lo más rápido posible, es necesario de un ancho de banda de al menos 100Mb/s.

1.11. Condiciones del ambiente

El lugar donde se encuentren las computadoras accesibles por el usuario debe permanecer seco y limpio (libre de polvo o cualquier otra sustancia que pueda flotar en el aire), para evitar daños en el hardware y así garantizar el correcto funcionamiento del software además de las computadoras. Como es posible el uso de químicos en los procesos realizados por los investigadores, es necesario asegurarse que la operación del equipo se haga bajo las siguientes condiciones:

1. Nunca usar las computadoras teniendo las manos húmedas, por ningún tipo de líquido o sustancia.

2. Evitar el uso de los computadores en lugares donde se trabaje con sustancias que puedan desprender vapor o humo, esto puede causar reacciones no deseadas en el comportamiento del equipo.
3. Mantener los equipos lejos de cualquier sustancia, fuera del lugar donde no se hagan labores de computación de imágenes o resultados.

1.12. Seguridad del sistema

El sistema no maneja ningún tipo de información sensible, como por ejemplo identificación de usuarios, manejo de tarjetas de débito o crédito, información personal o ubicación de labores. Sin embargo, la información que se comparte entre el servidor y las estaciones en el laboratorio, debe ser única y exclusivamente leída y manipulada por los expertos en el desarrollo del experimento. Esta es una razón para asegurar la exactitud de la información y la protección del servidor de cualquier ataque externo.

La actualización regular e inmediata del sistema operativo es vital para mantener el nivel más alto de seguridad ofrecido por fabricante o distribuidor del software. Además de las actualizaciones que realice el equipo de trabajo al producto.

1.13. Administración de la información

La información que se recibe en el sistema son imágenes capturadas por uno de los sistemas externos a el producto bajo desarrollo, estas imágenes requieren ser almacenadas en alguna ruta significativa en el disco duro del computador para luego ser seleccionadas por el usuario y ser procesadas por el software instalado en el servidor. Luego de ser procesadas por los algoritmos, estas imágenes son enviadas devuelta a las computadoras en el laboratorio, antes de ser borradas para evitar el llenado del espacio en el servidor y posible malfuncionamiento del sistema.

1.14. Políticas y regulaciones

Existe un riesgo de exposición a químicos y sustancias que pueden colapsar el hardware en el que opera el software, también existe la posibilidad de dejarlo completamente inoperable. Por esta razón es completa y estrictamente necesario tener lugares especiales donde se ubiquen los computadores, las

condiciones de estos ambientes se describen en la sección 1.11 del presente documento, cualquier incumplimiento a estas reglas o normas de seguridad pueden provocar daños en otros equipos realmente caros o heridas al personal.

1.15. Sostenibilidad del ciclo de vida del software

Asegurar el buen funcionamiento del software durante su vida útil requiere de un constante análisis en el estado del servidor, por ejemplo, en el tiempo que toma analizar X cantidad de imágenes, para esto existen herramientas que monitorean la cantidad de recursos que están siendo utilizados por los procesos. La necesidad de mantener el sistema en su más alto rendimiento es fundamental para que los investigadores puedan centrarse en las tareas de más relevancia que dependen de los resultados del producto en desarrollo.

Al ser tareas de alto rendimiento, es posible que sea necesario la escalabilidad del hardware, en este caso es necesario verificar el estado y la frecuencia de cada uno de los CPUs, la memoria, el espacio en el disco duro y el estado de entrada/salida de la red que comunica cada uno de los servidores con los clientes.

1.16. Empaquetado, manejo, envío y transporte

El producto de software se pretende mantener en servidores en la nube donde puedan ser accedidos mediante un navegador web, es decir, no se requiere de instalación, almacenamiento o empaquetado.

1.17. Verificación

La aplicación se considerará funcional y aceptable en el momento que el usuario así lo considere, esto según los requerimientos establecidos en la sección 1.4.

1.18. Asunciones y dependencias

Se prevee que las imágenes sean almacenadas en el disco duro desde dónde se acceda a la sistema, además de esto cualquier imagen debe cumplir con un procesamiento especificado en el documento requerido no presente en este proyecto.

Las imágenes resultado no van a ser almacenadas en el servidor y por eso se borrarán luego de cierto tiempo.

El rendimiento de la infraestructura dentro del laboratorio (computadoras que acceden al sistema) depende del uso que se le de, al no ser estaciones completamente dedicadas a las tareas aquí especificadas pueden perder rendimiento. La puesta a punto depende del mantenimiento que el personal del laboratorio brinde.

2. Validación del diseño

Requerimientos del sistema:

1. Cargar una secuencia de imágenes almacenada en la dirección provista por el usuario.
2. Segmentar las imágenes en las regiones de fondo y no fondo.
 - a) Los resultados deben ser almacenados en archivos.
3. Calcular métrica de precisión de la segmentación o clasificación:
 - a) Para el caso de implementarse la segmentación, usar una métrica de segmentación como el índice de Dice o de Jaccard.

$$D = \frac{2|S \cap G|}{|S + G|}$$

donde S es la región automáticamente segmentada, G es la región manualmente segmentada, y el valor absoluto se refiere a la cantidad de píxeles en tales regiones.

4. Detectar y contar los objetos de interés en las imágenes (células), de forma que sea fácil de visualizar desde la interfaz gráfica.
 - a) Opcionalmente colorear y etiquetar cada objeto.
5. Genere un informe que indique por cada objeto encontrado, su centroide y su área, en formato .csv.
6. Permitir la navegación entre imágenes y resultados.

- a) Debe permitirse el procesamiento de un lote de imágenes en una sola acción.
7. Visualizar el tiempo de ejecución por imagen y por el lote de imágenes.
 8. Todas las funcionalidades deben ser accesibles desde la interfaz web del sistema.

Número de requerimiento	Diagrama de clases (back-end)	Diagrama de clases (front-end)	Diagrama de componentes	Diagrama de secuencia
1	X		X	X
2	X			
3	X			
4	X			
5	X	X		
6		X		X
7	X	X	X	
8	X	X	X	X

Cuadro 1: Validación de diseño

3. Validación de la implementación

Requerimientos del sistema:

1. Diagrama de clases (back-end)
2. Diagrama de clases (front-end)
3. Diagrama de componentes
4. Diagrama de secuencia

Número de requerimiento	Código fuente	OpenCV	IDE	Plugin estandar codificación	JUnit	Hardware	Servlet
1	X	X	X	X	X		X
2	X		X	X			X
3		X			X	X	X
4		X					

Cuadro 2: Validación de implementación

4. Detalles de herramientas de estándares de codificación

Se utilizó CodePro Analytix (un plugin de Google para Eclipse) que revisa que tanto se apegue al estandar. Esto se realiza dando click derecho al archivo (dentro de Eclipse), codepro tools y audit code; esto genera banderas en la línea de código donde no se siga el estandar, las banderas pueden ser de errores graves o medios dependiendo del color rojo o amarillo.

5. Obtención de la versión actual del sistema

Para obtener la última versión del software se deben seguir los siguientes pasos:

1. Se da click derecho sobre el proyecto y se sitúa el puntero sobre Team y seguido Share Project

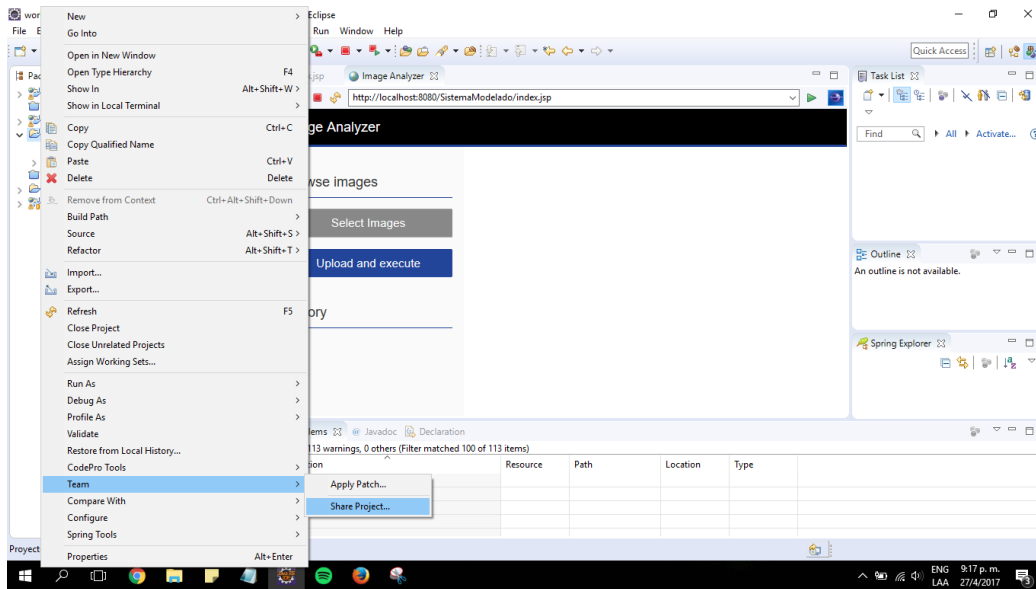


Figura 1: Team

2. Se selecciona Pull

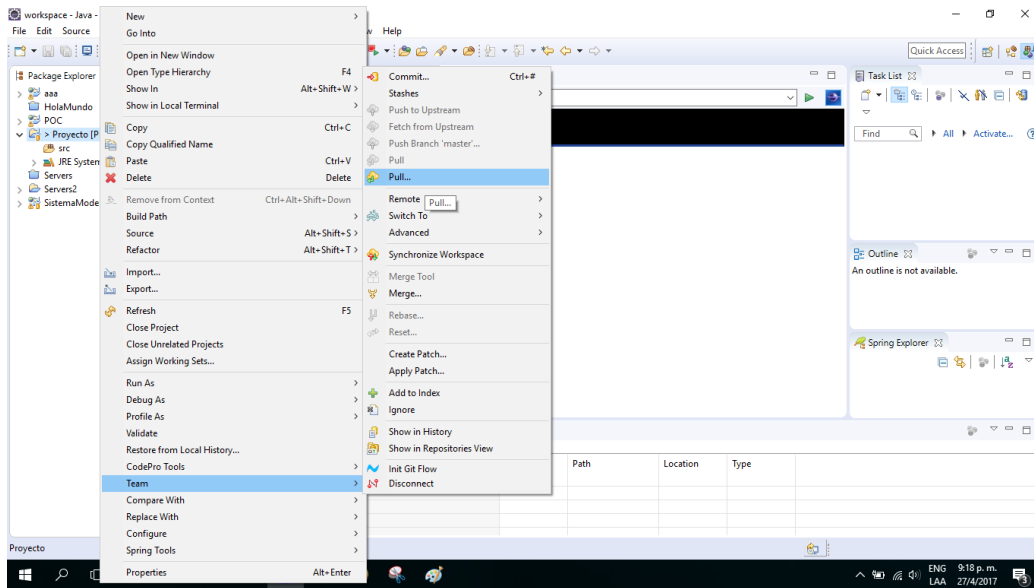


Figura 2: Pull

3. Se selecciona la ubicación donde se va a descargar el repositorio

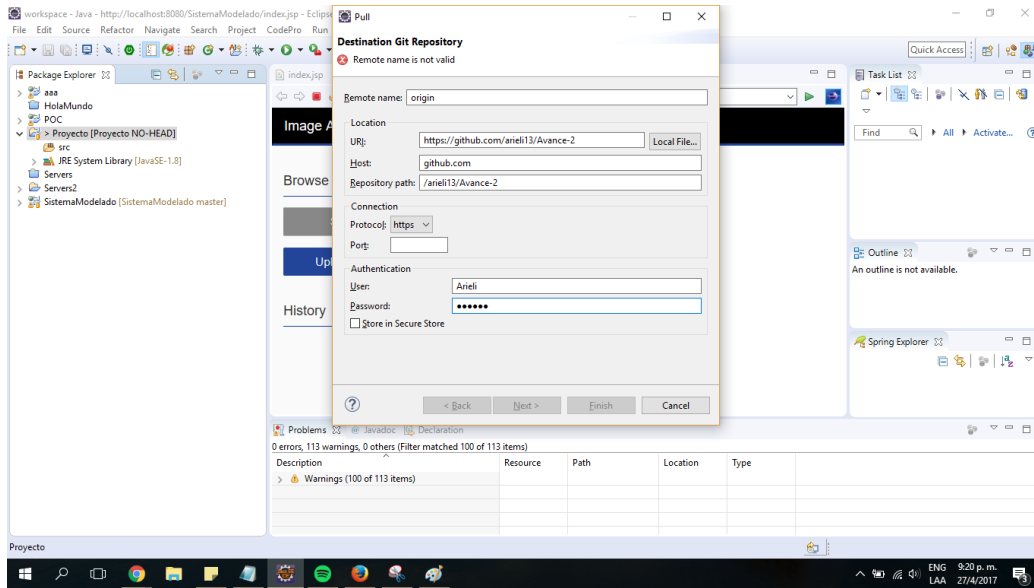


Figura 3: Interfaz de la aplicación

4. Abre la ventana para seleccionar el proyecto y se añade el link del repositorio donde se encuentra ubicado el proyecto, además se debe ingresar con un usuario y contraseña

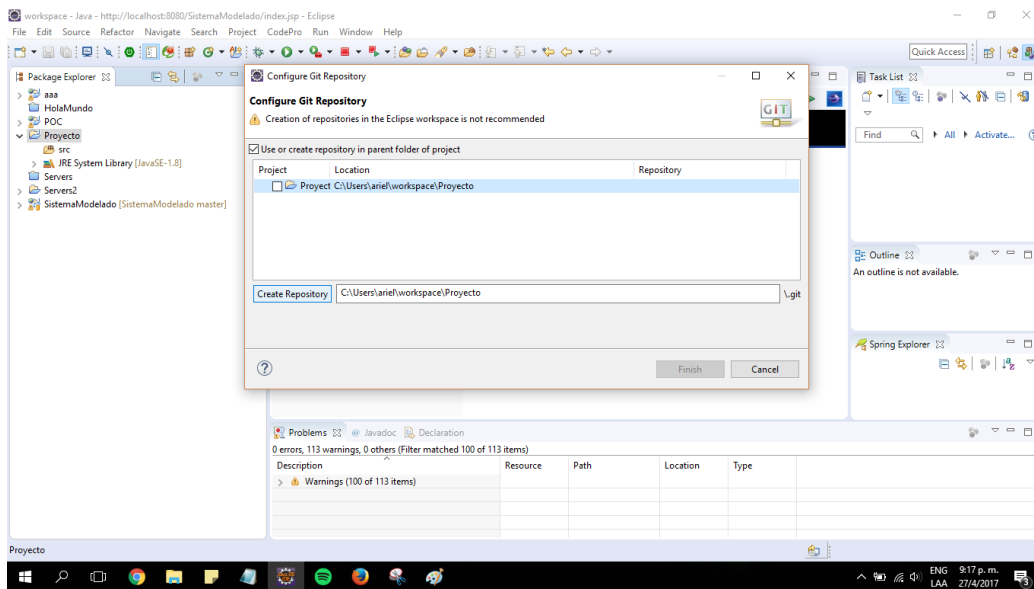


Figura 4: Ingreso al repositorio

6. Métricas

1. **Facilidad de uso:** El sistema de es relativamente fácil de usar y el entrenamiento del usuario final para su uso frecuente no requiere de muchi tiempo. Esto se debe a que se muestra una interfaz simple de usar, debidamente identificados, uno para subir las imágenes, otro para comenzar el proceso y un grupo de botones que indican los procesos anteriores y sus resultados.

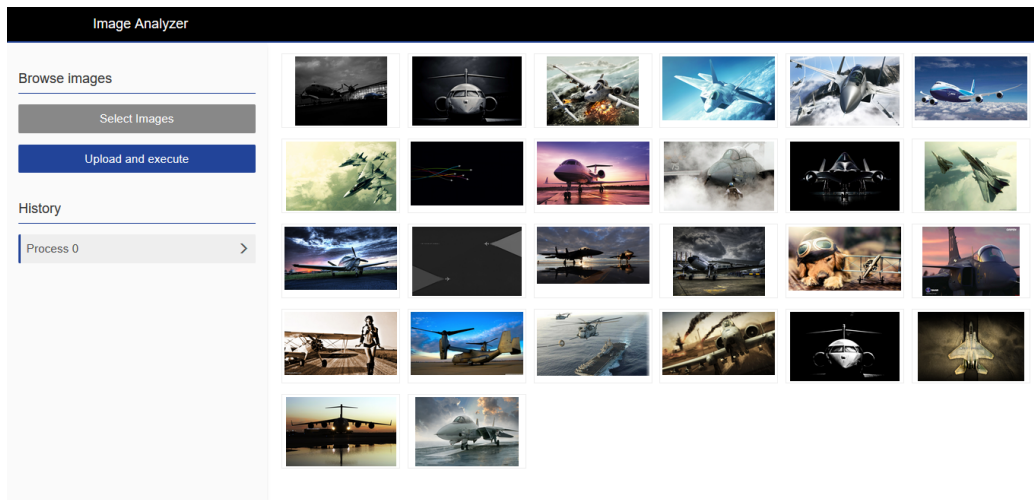


Figura 5: Interfaz de la aplicación

2. **Velocidad de respuesta del servidor:** La velocidad de respuesta del servidor viene dado por la carga de imágenes que se haga en determinado tiempo y el proceso que toma el análisis y procesado de las mismas, así como del hardware provisto. Con respecto al hardware, se cuenta con el servidor en sí, las computadoras cliente y la red para la subida de imágenes.

7. Especificación de pruebas unitarias

1. **Coeficiente de Dice:** Se segmentó una imagen manualmente y otra la segmentó el algoritmo que ha sido implementado, posteriormente, se pasan por la función que retorna el Coeficiente de Dice y se acepta el número si este era mayor a 0.7
2. **Kittler:** La imagen provista como ejemplo fue pasada por el algoritmo de Kittler y el umbral óptimo es comparado con el brindado por el profesor.
3. **Conteo de células:** Se cuenta la cantidad de células manualmente de una imagen y se compara ese número con el resultado del algoritmo con esa misma imagen.

4. **Conteo de células 2:** Se cuenta la cantidad de células manualmente de una imagen (distinta a la anterior) y se compara ese número con el resultado del algoritmo con esa misma imagen.