

Aseguramiento de la Calidad del Software

Avance 1 de Proyecto

Primer Semestre 2017

Instituto Tecnológico de Costa Rica,
Escuela de Ingeniería en Computación
Esteban Arias - 200546435
Daniel Estrada Hernandez - 2014081245
Edward Rodríguez - 2014082498
Jose Ruiz Jara - 2014096391

Sunday 26th March, 2017

1 SyRS ISO-9126

1.1 Introducción

El avance tecnológico aplicado al análisis de imágenes biomédicas y reconocimiento de patrones de las mismas ha despertado gran interés en el campo de la biomedicina, ingeniería en software y ha constituido una gran evolución y ambiente de investigación en los sistemas de información orientado a los profesionales de la salud.

A continuación se detalla una serie de características del sistema basado en el estándar ISO/IEC/IEEE 29148-2011

1.1.1 Propósito del sistema

El sistema descrito en este documento tiene como objetivo principal reconocer y analizar patrones de señales en una imagen de tejidos celulares. Para lograr dicha finalidad es necesario realizar todo un proceso de segmentación de la imagen y extraer características de cada objeto obtenido y al final tomar una o varias decisiones específicas.

1.1.2 Alcance del sistema

Se va a construir una aplicación web que permitirá al usuario cargar imágenes de tejidos celulares generadas por un microscopio electrónico basado en fluorescencia.

Por lo general este tipo de imágenes presentan ruido. Este ruido es un equivalente a cambios aleatorios en la señal debido a causas externas.; ejemplo de ello son las variaciones térmicas, electromagnéticas, etc.

Existen varios tipos de ruido, sin embargo el tipo más frecuente es el Gaussiano. Para mitigar este ruido el sistema deberá realizar un proceso de transformación de la señal de entrada conocida como filtrado.

Este proceso de filtrado consiste en la eliminación de ciertas características de la señal utilizando una función para tal efecto.

La aplicación contará con una interfaz gráfica mediante la cual el usuario es capaz de cargar un listado de imágenes alojadas en una ruta específica y definida por este.

Posteriormente el usuario va a ecualizar el histograma de cada una de las imágenes para mejorar el contraste.

Una vez ecualizado el histograma el usuario utilizará dos tipos de filtros: El Gaussiano y el filtro bilateral; cuando se hayan aplicado los filtros el usuario podrá navegar, mostrar y descargar las imágenes resultantes.

Es importante destacar que se deben configurar los filtros de navegación así como definir los tamaños de la ventana y los parámetros de desviaciones estándar.

Se debe permitir la navegación entre las imágenes y los resultados de las mismas así como visualizar el tiempo transcurrido durante el procesamiento de las imágenes.

Utilizar métricas de exactitud en los resultados calculando la relación pico señal a ruido. Para ello se deberá utilizar una imagen sin ruido y otra con ruido artificial y de esta forma poder determinar la mejora en la calidad de la imagen.

1.1.3 Resumen del sistema

Diseño de una aplicación web que permitirá al usuario cargar una secuencia de imágenes generadas por un microscopio electrónico basado en fluorescencia.

Las imágenes cargadas son analizadas y procesadas por el usuario utilizando dos tipos de filtros: Filtro Gaussiano y Filtro bilateral.

1.1.3.1 Contexto del sistema

Los sistemas de captura y procesamiento de imágenes orientadas al campo de la biomedicina tales como radiografías, resonancias magnéticas y otro tipo de sistemas informáticos que contribuyen al sector salud cada vez son más comunes en nuestra vida cotidiana y han expandido a gran escala el interés por esta área tecnológica. Lo descrito anteriormente es la causa-efecto del sistema en cuestión.

1.1.3.2 Funciones del sistema

- Configurar directorio de entrada de secuencia de imágenes: usuario define la ruta de entrada de las imágenes.
- Ecualizar histograma de imagen: utilizado para mejorar el contraste de la imagen
- Filtrar imágenes utilizando filtro Gaussiano y filtro Bilateral: utilizados para seleccionar los objetos y características de interés de la imagen.
- Mostrar imágenes resultantes: Visualización de los resultados posteriores al filtrado.
- Descargar imágenes resultantes: el usuario podrá hacer descarga de las imágenes procesadas.
- Navegar entre imágenes y resultados: visualización de imágenes cargadas e imágenes procesadas.
- Visualizar el tiempo de ejecución del procesamiento de las imágenes: El usuario podrá ver el progreso de la ejecución de un lote de imágenes.
- Calcular pico señal a ruido: utilizando una imagen sin ruido y otra con ruido artificial para realizar una comparación y visualización de la mejora en la calidad de la imagen.

1.1.3.3 Características del usuario

Usuario especialista en quimiosensibilidad. Conocimientos Básicos de computación

1.1.4 Referencias

Laboratorio de Quimiosensibilidad tumoral, de la Universidad de Costa Rica.

Doctores Steve Quirós y Rodrigo Mora.(Proyectos de investigación relacionados con la terapia personalizada para pacientes con cáncer).

Su premisa consiste en desarrollar tratamientos quimioterapéuticos personalizados a cada paciente, de forma que los químicos utilizados sean optimizados. Para ello es necesario realizar experimentos invivo para estudiar la reacción de a nivel celular en el tejido cancerígeno a un tratamiento específico.

1.1.5 Requerimientos del Sistema

1.1.5.1 Requerimientos funcionales

El usuario debe poder cargar una serie de imágenes al programa.

El programa permite cargar entre una o más imágenes para posteriormente procesarlas.

El programa permite al usuario poder escoger entre distintos algoritmos para el procesamiento de imágenes.

El programa debe permitir la descarga de las imágenes procesadas.

El programa debe permitir el ingreso de parámetros.

El programa permite la visualización de los resultados del procesamiento de las imágenes.

1.1.5.2 Requerimientos de usabilidad

La interfaz debe ser sencilla.

Métrica: El entendimiento de la interfaz debe ser comprendido con una efectividad del 80%.

El programa debe ser fácil de usar.

Métrica: La cantidad de clicks que debe hacer el usuario para poner en marcha el programa debe ser menor a 10.

El programa debe permitir el acceso al usuario desde cualquier navegador.

Métrica: Se probará el funcionamiento del sistema desde distintos navegadores, buscando una efectividad del 90%

El manual de usuario debe ser claro para el usuario

Métrica: El tiempo que el usuario invierte en leer y comprender el manual de usuario debe ser menor a treinta minutos.

Las funciones del programa deben ser fácilmente entendibles.

Métrica: El tiempo que el usuario se detiene por que no comprende una funcion debe ser menor a dos minutos.

1.1.5.3 Requerimientos de desempeño

El programa debe ser capaz de procesar 1 imagen en 6 segundos.

Métrica: La duración del procesamiento de grupos de imágenes debe cumplir con el promedio de menos de 6 segundos por imagen en al menos el 90% de los casos.

El programa debe ser capaz de procesar un máximo de 3MB por tamaño de imagen.

Métrica: Se realizan pruebas donde la aceptación y correcto procesamiento de imágenes de hasta 3 MB es acertada en el 95% de los casos.

El sistema debe hacer uso adecuado de los recursos.

Métrica: Se ejecutarán pruebas que intenten emular el uso máximo de memoria que pueda hacer el software, para ello se analizara la memoria ram disponible antes y durante cada prueba. Con el fin de buscar un máximo aceptable y definir limites de entrada que no sobrecarguen el sistema.

1.1.5.4 Interface del sistema

1.1.5.5 Operaciones del sistema

Configurar una ruta de entrada donde se cargan las imagenes.

Ecualizar el histograma de una imagen.

Procesamiento de imagenes mediante el filtro Gaussiano y filtro Bilateral.

Mostrar imagenes resultantes.

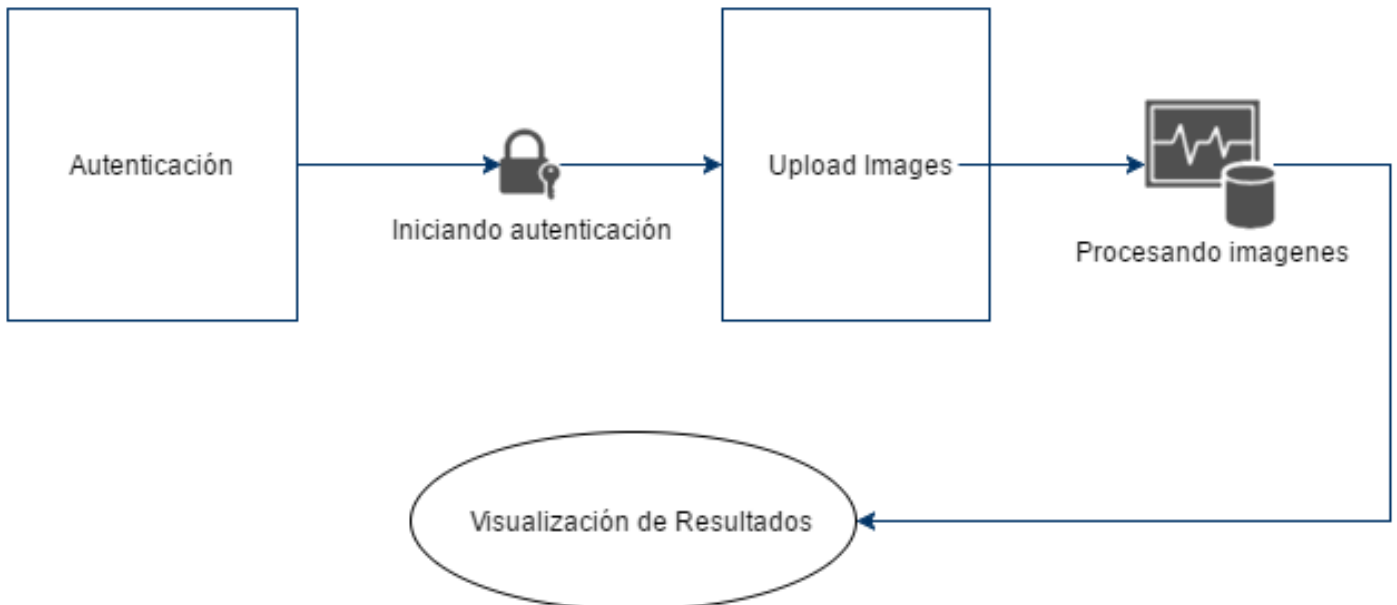
Descargar imagenes resultantes.

Navegar entre imagenes e imagenes resultantes.

Mostrar el tiempo de ejecución durante el procesamiento de imágenes.

Calcular pico señal a ruido.

1.1.5.6 Modos y estados del sistema



1.1.5.7 Características físicas

Aplicación web en java.

Servidor Apache.

1.1.5.8 Condiciones ambientales

El sistema se alojará en un servidor Apache utilizando una máquina física local.

1.1.5.9 Sistema de seguridad

El acceso al sistema se hará bajo una conexión segura utilizando un certificado de seguridad (SSL) y el método de acceso propiamente a la aplicación mediante una autenticación de usuario.

1.1.5.10 Gestión de la información

Una base de datos que permita tener un control de los usuarios de la aplicación así como el almacenamiento de las imágenes subidas y los resultados obtenidos del análisis y procesamiento.

2 Estandar de Codificación

Para el desarrollo del proyecto se establece la utilización del estándar de Google Java Style, el cual se aplicara tanto para java como JavaScript, las cuales son las tecnologías utilizadas.

Si bien Google cuenta también con un estilo para JavaScript, se opta por seguir las reglas que aplicasen que se encuentren en el estándar de java para facilitar la implementación de un código fácil de leer.

Este estándar define una serie de sugerencias a seguir y provee un archivo .xml que facilitara la revisión del cumplimiento del mismo en Eclipse, este archivo se puede encontrar en <https://github.com/google/styleguide>

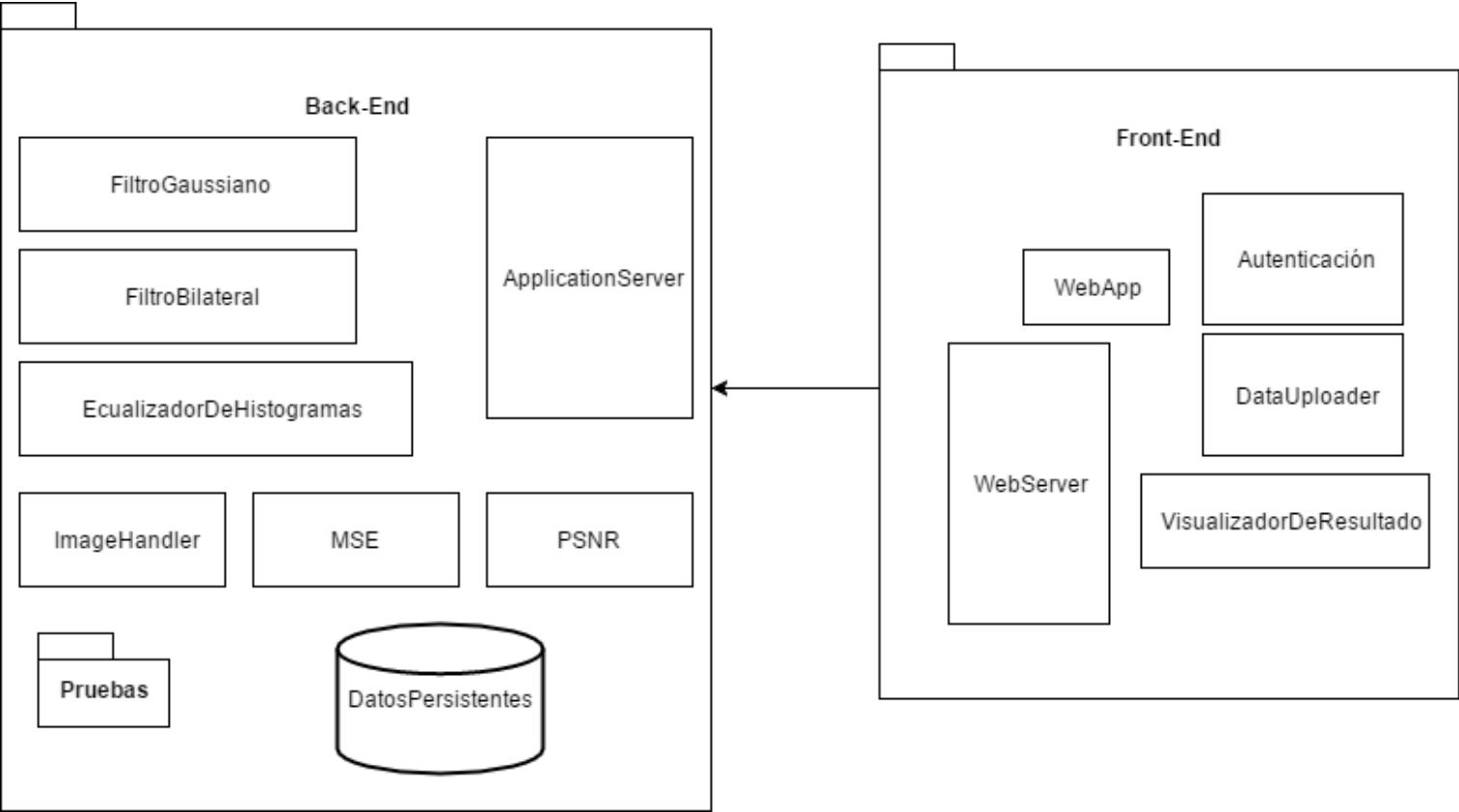
La guía completa del estándar se puede leer en <https://google.github.io/styleguide/javaguide.html>

A continuación definimos algunos de los puntos definidos en el estándar de Google para codificación en Java, y algunos apuntes extra que Google no termina de definir.

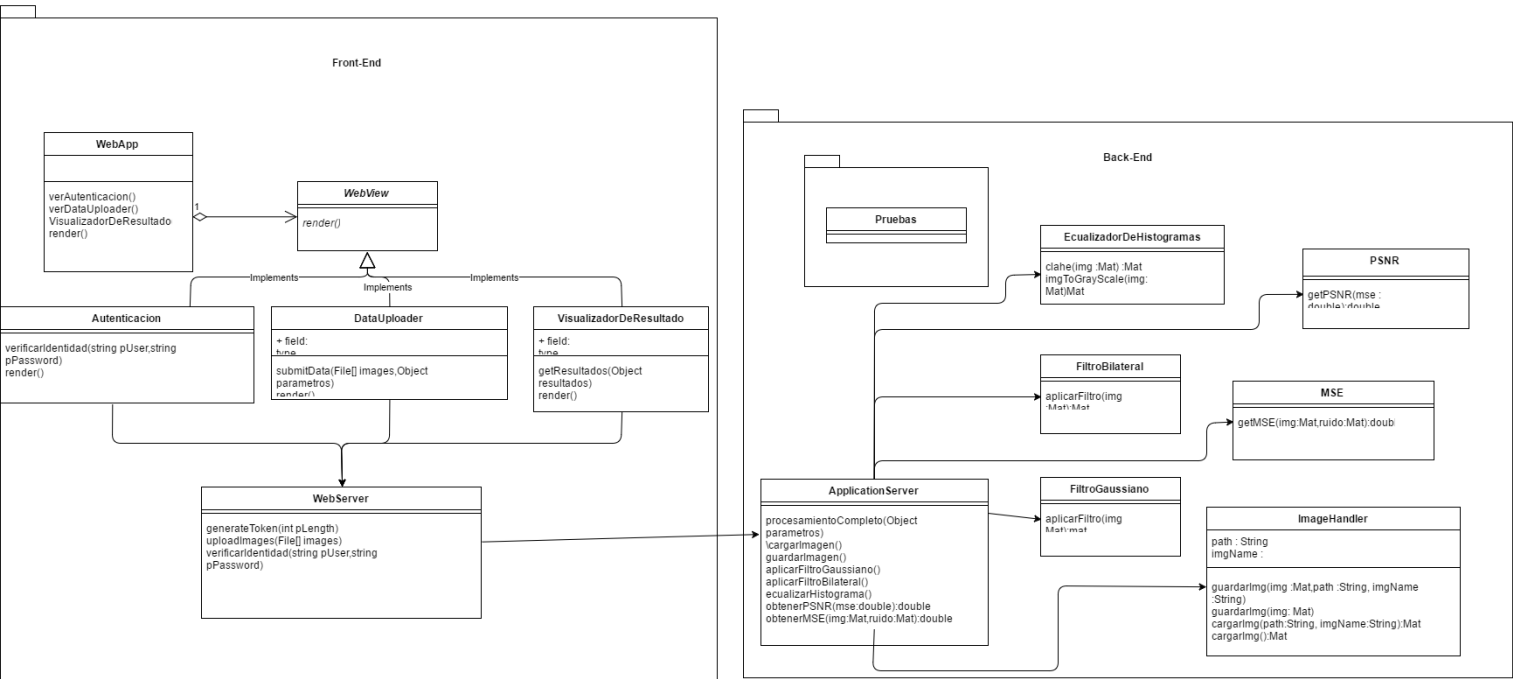
1. Se seguirá el estándar javadoc para la documentación de cada función.
2. Se utilizaran +4 espacios para la indentación
3. Se buscara evitar las líneas con más de 100 caracteres, si se debe fraccionar una línea se intentaran seguir los siguientes principios: Fraccionar luego de una coma. Fraccionar luego de un operador.
4. Se establece una declaración por línea.
5. Cada línea debe contener a lo sumo una sentencia
6. Los nombres de las clases deben ser simples y descriptivos, y seguir el formato Upper Camel Case (cuando son compuestos tendrán la primera letra de cada palabra en mayúsculas).
7. No utilizar asignaciones embebidas.
8. No abusar del uso de comentarios en el código.
9. Los comentarios deben ser concisos.
10. Evitar los nombres de variables con más de 15 caracteres.
11. Evitar nombres que difieran solamente en una letra.
12. Se evitara el uso del carácter “_” en el nombre de variables.
13. Los nombres de las clases de “test” comenzaran con el nombre la clase a evaluar y terminan en “test”.
14. Variables locales y parámetros se escriben usando lowerCamelCase

3 Diagramas del Sistema

3.1 Diagrama de componentes del sistema



3.2 Diagrama de clases del sistema



En la parte de Front-End se utiliza el patrón de diseño "Strategy"[1], para poder cambiar la funcionalidad de cada vista en tiempo de ejecución.

4 Actividades de aseguramiento de calidad

La siguiente tabla contiene las actividades con las cuales se va a dar el aseguramiento de la calidad. La columna Categoría presenta tres distintos valores: Planeamiento del ACS PP. Aseguramiento del producto AP. Aseguramiento del proceso APR.

Actividad	Tarea	Responsable	Planificación (sprint o iteración)	Categoría
Definición del método para tener un control versiones.	Definir cual herramienta se debe usar para el control de versiones.	Daniel Estrada.	Sprint 1	PP
Definición de los estándares de codificación.	Definir el estándar de codificación a seguir en el proyecto.	Daniel Estrada.	Sprint 1	PP
Definición de modelos o arquitectura aplicar al proyecto.	1. Desarrollar los diagramas generales que definirán al proyecto, creación de la arquitectura del software, para esta tarea también se debe incluir la definición de una herramienta para la realización del modelado. 2. Definir un patrón estructural para complementar la arquitectura del proyecto.	Jose Ruiz	Sprint 1	PP
Revisión y validación de requerimientos del sistema.	Se debe corroborar según las etapas definidas en la planificación, que se mantengan y se cumplan los requerimientos del programa, para esto se presentan muestras del programa al cliente, y posteriormente se validan o se definen los cambios por escrito que se deben realizar.	Esteban Arias	Sprint 3	AP
Verificación del uso del sistema por medio de pruebas funcionales y no funcionales.	Realizar las pruebas necesarias para verificar el cumplimiento de tareas funcional y no funcional del programa.	Esteban Arias	Sprint 3	AP
Definición de las métricas.	Se deben crear métricas para permitir el ACS.	Daniel Estrada	Sprint 1	AP
Construcción de los tests unitarios, de integración y aceptación.	Desarrollar tests unitarios, de integración y aceptación.	Daniel Estrada	Sprint 1 y Sprint 3.	PP
Revisión de los tests unitarios, de integración y aceptación.	Revisar el funcionamiento correcto de los tests.	Jose Ruiz	Sprint 4	AP
Evaluar la utilización de la arquitectura de software definida.	Ver y revisar el cumplimiento de los modelos definidos como arquitectura del software.	Edward Rodríguez.	Sprint 3.	APR
Verificar el seguimiento correcto en los estándares de codificación.	Hacer una revisión de los estándares definidos como codificación y verificar que su cumple su uso.	Edward Rodríguez.	Sprint 3 y Sprint 4	AP
Problemas presentados.	Se debe crear un informe detallado sobre problemas dados y soluciones presentadas en el sprint en cuestion.	Esteban Arias.	Al final de cada sprint.	APR
Evaluar el ciclo de vida	Evaluar que se haya seguido un ciclo de vida para el proceso de ACS.	Esteban Arias	Sprint final	APR
Evaluar las métricas.	Se debe de evaluar el producto haciendo uso de de las métricas que se definieron.	Jose Ruiz	Sprint 3	AP
Reevaluación de las métricas.	De acuerdo a la evaluación generada en el sprint 3 sobre las métricas se debe actualizar la definición de estas, en caso de que se amerite.	Edward Rodríguez.	Sprint 3	AP
Evaluar los procesos.	Se deben de evaluar los procesos utilizados para ACS y verificar que se siga el rumbo previamente establecido.	Edward Rodríguez.	Sprint 3	AP

5 Referencias

[1] *Strategy .NET Design Pattern in C# and VB - dofactory.com*. (2017). Dofactory.com. Accedido el 26 de Marzo de 2017, desde <http://www.dofactory.com/net/strategy-design-pattern>