

Aseguramiento de la Calidad del Software:

Avance 1, proyecto semestral

Yin Cheng Liang
2015018200

Jose Antonio Salas
2015013633

Emanuel Alvarado
2015183027

18 de Septiembre de 2018

Git: <https://github.com/joseant12/CellSegmentationProject>

1 Introduction

En este presente primer avance del proyecto semestral sobre el análisis automatizado de imágenes biomédicas se mostrara el documento de requerimientos del sistema(SyRS), basados en el estandar ISO/IEC/IEEE 29148-2011. Por otro lado, se documentara sobre los estándares de codificación y documentación que se utilizará a lo largo del proyecto en base a una investigación. Además se incluirá algunos diagramas que detallen la primer iteración de la arquitectura del sistema y también se muestran las actividades de aseguramiento de calidad a realizar en cada sprint. Por último se presentará las pruebas unitarias hechas para la funcionalidad de la carga de una secuencia de imágenes almacenada en la dirección provista por el usuario.

2 Documento de especificación de los requerimientos del sistema(SyRS)

2.1 Propósito del sistema

El desarrollo de este sistema tiene el propósito de agilizar y facilitar el proceso de investigación en la industria biomédica por medio del análisis automatizado de imágenes biomédicas obtenidas con la ayuda de microscopios electrónicos de fluorescencia. El análisis manual de miles de imágenes es una tarea difícil y agotador para un experto, por eso con la ayuda del sistema se podrá ahorrar una cantidad importante de tiempo y esfuerzo para utilizarla para realizar otras actividades. Además de poder generar con el sistema resultados eficientes a gran escala.

2.2 Alcance del sistema

El sistema de análisis automatizado de células tendrá la funcionalidad de procesamiento de imágenes biomédicas que consiste en la segmentación y conteo de las mismas dado una serie de imágenes provistas por los doctores. Al finalizar el proceso se generan varios archivos con el informe del análisis detallando la cantidad de células y la posición de cada una en cada cuadro.

El sistema no será para todo público, sino solamente para los doctores y personal autorizado del Laboratorio de Quimiosensibilidad tumoral de la Universidad de Costa Rica.

Los beneficios al utilizar este sistema es la mejora en el tiempo de respuesta del análisis, disminuyendo el esfuerzo por parte de los expertos y así agilizando el proceso de atención a los pacientes.

2.3 Resumen del sistema

2.3.1 Contexto del sistema

El sistema funcionara en una aplicación web en donde se tendrá acceso a una o varias direcciones provistas por el usuario en donde se encuentran localizadas las imágenes biomédicas a analizar.

El usuario dará una dirección, a partir del cual se cargan las imágenes obtenidas usando microscopios electrónicos de fluorescencia a la aplicación. Luego se procederá al proceso de análisis de las imágenes y al finalizar el usuario podrá visualizar el informe generado de los resultados del estudio en pantalla y en archivos aparte.

El diagrama de la arquitectura del sistema se muestra en la Figura 2.

2.3.2 Funciones del sistema

El sistema tendrá la capacidad de cargar una serie de imágenes biomédicas en escala de grises adquiridas mediante un microscopio electrónico y procesarlas de manera que facilite su análisis para luego hacer la detección y el conteo de objetos importantes mostradas en la imagen. En la etapa final del proceso, además de mostrar los resultados obtenidos del análisis en pantalla, se generarán archivos de imágenes y archivos de informes en formato .csv.

2.3.3 Características del usuario

Los usuarios finales, los cuales utilizaran la aplicación serán microbiólogos con conocimientos generales de herramientas informáticas y que forman parte del equipo del Laboratorio de Quimiosensibilidad Tumoral de la Universidad de Costa Rica.

2.4 Requerimientos funcionales

2.4.1 Cargar secuencia de imágenes

Dado una dirección provista por el usuario se podrá cargar un conjunto de imágenes biomédicas tomadas con microscopios electrónicos de fluorescencia.

2.4.2 Detección y conteo de objetos

Después de haberse cargado las imágenes, con esta funcionalidad se lograra detectar y contar los objetos de interés de las imágenes como son las células, y mostrarlas en la interfaz gráfica para su visualización.

2.4.3 Generar informe

Al finalizar el proceso de análisis, los resultados obtenidos se guardarán en archivos en formato .csv en donde se indicará la cantidad de células, la posición de cada una en cada cuadro, entre otras características de los objetos encontrados.

2.4.4 Visualizar proceso

Se mostrará los detalles por cada imagen o por lote de imágenes en pantalla del proceso de análisis en cuando esta finalice. Como por ejemplo visualizar el tiempo de ejecución.

2.4.5 Cálculo de métricas

A partir de una imagen se utilizara una métrica, por ejemplo la métrica de Dice, para cuantificar la precisión de la segmentación, evaluando la diferencia de señales binarias y luego generar una imagen ground truth con los blobs de los objetos de interés manualmente marcados.

2.5 Requerimientos de usabilidad

El sistema debe de tener facilidad de aprendizaje y usabilidad para los usuarios ya que como es una aplicación de análisis de imágenes biomédicas, esto requiere un tiempo estimado en su ejecución, y mientras menor sea el tiempo invertido en aprender cómo funciona y usarlo mejor. Y en cuanto a la interfaz gráfica, debe de ser simple y cómodo para el usuario, de manera que los microbiólogos puedan entender el sistema y encontrar las funcionalidades necesarias para su trabajo.

2.6 Requerimientos del rendimiento

La apertura de la aplicación y la carga de todas las funcionalidades en la interfaz gráfica no debería de durar mucho ya que es un sistema simple es esta parte. La carga de imágenes junto con su procesamiento y la generación de informes

podría durar su tiempo, ya que dependiendo de la cantidad de imágenes a analizar podría disminuir o aumentar el tiempo de ejecución pero lo ideal es que se trate de durar lo menos posible y así agilizar el proceso de investigación.

2.7 Interfaces del sistema

El sistema se base en la arquitectura Cliente-Servidor por lo que dispondrá tanto de una interface para cliente y para el servidor. Se dispondrá de una interfaz gráfica en la cual el usuario pueda realizar todas las funcionalidades que el sistema va a disponer para los usuarios finales.

2.8 Operaciones del sistema

2.8.1 Requerimientos del integración del sistema humano

El sistema debe ser altamente tolerante a fallos provocados por la interacción con el usuario, con lo cual se espera que el sistema al tener problemas por la mala manipulación del mismo sea capaz en la mayor parte de poder recuperarse y así poder seguir funcionando correctamente, esto para poder procesar la mayor cantidad de imágenes posibles.

El sistema al procesar imágenes puede darse el caso en que el usuario ingrese imágenes incorrectas o que del todo no se puedan procesar, pudiendo provocar errores en el sistema.

2.8.2 Mantenibilidad

Al esperarse poder procesar una secuencia de imágenes que puede ser en algunos casos de una cantidad considerable, puede darse los casos que el sistema se sature o bloquee lo cual implicaría una revisión y mejora en el código fuente para optimizar el tiempo de procesamiento. Se espera que el tiempo de mantenibilidad no sobrepase más de 5 horas.

2.8.3 Confiabilidad

El usuario final del sistema son médicos y especialista en la interpretación de imágenes médicas, los cuales darán al sistemas las imágenes correctas para su posterior procesamiento, al ser dirigido para estos usuario se logra un nivel altamente positivo de confiabilidad.

2.9 Modos y estados del sistema

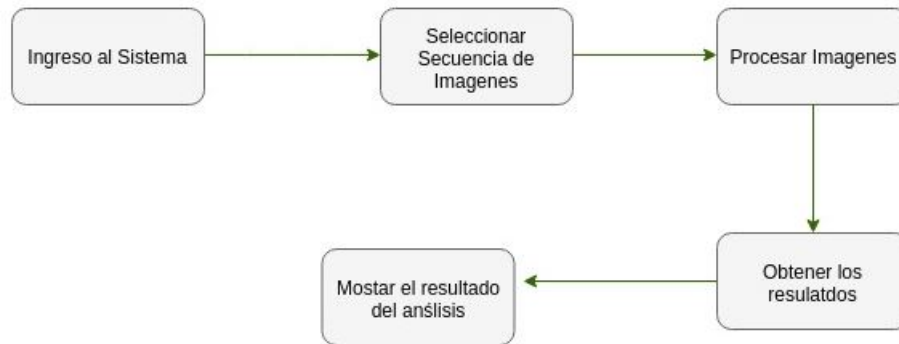


Figure 1: Modos y estados del sistema

2.10 Características físicas

2.10.1 Requerimientos físicos

El sistema va ser instalado en equipos propios de los especialistas médicos, en el laboratorio de trabajo de ellos mismo, aparte del computador solo se va a necesitar un monitor donde se pueda realizar las tareas del sistema.y muestre resultados.

2.10.2 Requerimientos de adaptabilidad

En caso de que el volumen de procesamiento de imágenes crezca rápidamente se debe de necesitar de mayor poder de procesamiento lo que implicaría en un futuro, la adquisición de más módulos de memoria ram y procesador. Para que el sistema cumpla con sus tareas en un tiempo considerablemente rápido.

2.11 Condiciones ambientales

Al ser un sistema de procesamiento de imágenes es recomendable que el equipo esté en un lugar fresco, para que la alta temperatura no afecte el funcionamiento del equipo al poder llegar a ser sobrecargado con mucho proceso.

2.12 Seguridad el sistema

El sistema será utilizado por especialista con un gran conocimiento en imágenes médicas para el tratamiento de enfermedades. Con respecto a las imágenes que fueron ya procesadas, serán almacenadas en una base de datos para mantener una copia de seguridad de las mismas, la base de datos estará protegida por su respectivo usuario y contraseña para prevenir posible mala manipulación de las misma.

2.13 Administración de la información

El sistema de análisis de segmentación de células recibe como entrada tres atributos para crear una colección de imágenes. Estos campos corresponden a un título de colección, una descripción de no más de 256 caracteres y por último una o más imágenes. Estas imágenes deben coincidir con las características de los ejemplos. Se recomienda que las imágenes sean formato png, aunque cabe aclarar que el sistema soporta imágenes formato jpg. La salida de datos consta de un archivo en formato csv donde mostrará el centroide y el área de cada objeto encontrado por el sistema.

2.14 Regulaciones y políticas

En el apartado de políticas del sistema, se encuentra la confidencialidad con los datos suministrados por el cliente. Mediante la funcionalidad de registro/login los clientes acceden a su cuenta privada y suben su colección de imágenes a evaluar, para luego recolectar los resultados correspondientes, sin que ningún agente externo interfiera en el proceso. El sistema no se hace cargo de irregularidades en los resultados propuestos, por ello se insta al usuario a verificar la correctitud de los datos.

2.15 Sostenibilidad del ciclo de vida del sistema

El sistema será sometido a pruebas rigurosas para asegurar una calidad óptima en pro de los usuarios. Para ello se tiene control de una serie de procedimientos a efectuarse luego de cada sprint en el transcurso de desarrollo de la aplicación. También se tiene programado realizar mediciones al software con respecto a una serie de métricas previamente establecidas, con el objetivo de tener procesos de control de calidad exitosos. El sistema cuenta con una documentación robusta que garantiza su mantenibilidad. Se espera capacitar al usuario por medio de guías y documentación con tal de satisfacer sus necesidades.

2.16 Empaquetado, manejo, envío y transporte

El sistema consta con atributos de calidad como instalabilidad y adaptabilidad, ya que consta de archivos digitales que pueden ser instalados en cualquier computador con soporte de Django. El sistema está contenido en un repositorio en línea lo cual lo hace fácilmente accesible.

2.17 Verificación

La verificación del sistema está a cargo de agentes tanto internos como externos. Se utilizará herramientas como Sonarqube, Reqview, PyUnit, Doxygen, análisis de salidas, fl score y entrevista con clientes. Para mayor información al respecto, se recomienda ver el documento de "Atributos de calidad y métricas" (correspondiente a la tarea 1) donde se expone detalladamente el uso de cada una de estas herramientas.

2.18 Suposiciones y dependencias

En cuanto a las suposiciones y dependencias se anuncia lo siguiente:

1. El usuario tiene un conocimiento mínimo del campo biológico correspondiente a las funciones del sistema.
2. El usuario tiene el conocimiento mínimo en cuanto al uso de una computadora y acceso a la web.
3. Las imágenes suministradas al sistema son únicamente de células con el formato correcto.
4. El usuario está consciente que las imágenes suministradas al sistema pueden ser manipuladas como este desee.

3 Estándares de codificación y documentación

El estándar de documentación que nuestro grupo de trabajo escogió para la elaboración de este proyecto se trata de javadocs, ya que, a pesar de no estar trabajando sobre el lenguaje de programación Java, es el estándar en el cual más conocimiento y experiencia tenemos con otros proyectos. Además, la herramienta de generación de documentación elegida, Doxygen, cuenta con un módulo de python que acepta y reconoce javadocs de forma nativa, lo cual impulsó aún más esta escogencia.

En cuanto a la codificación, el estándar empleado en el proyecto es PEP8. Esta guía de codificación fue escogida debido a su gran popularidad y uso en la comunidad de desarrolladores de Python, convirtiéndola en líder dentro de su terreno. Otro motivo en consideración es que el propio creador de Python recomienda su uso.

4 Diagramas UML y de componentes

Durante la etapa de diseño del proyecto en cuestión, se decidió utilizar el patrón de diseño adaptador para la comunicación entre los módulos de imágenes y el modelo de reconocimiento de células. Este patrón fue escogido debido a la facilidad que brinda de ajustar módulos independientes entre sí (tal es el caso de la web app con keras) y de sustituir estos módulos sin hacer mayor cambio al código del programa (por ejemplo, si en un futuro se decide utilizar otro modelo de reconocimiento de patrones).

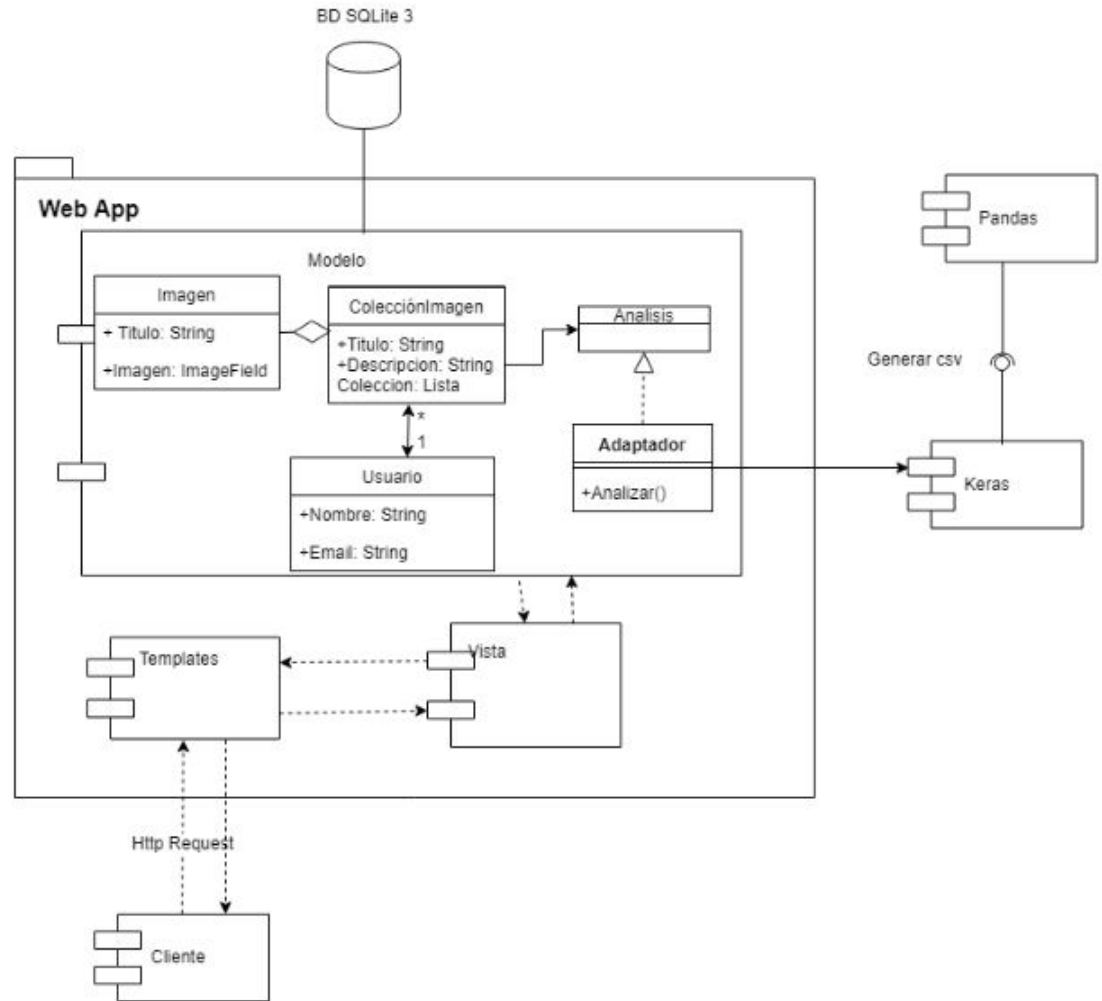


Figure 2: Diagrama de componentes y de clases

5 Actividades de aseguramiento de calidad

A continuación se mostraran el conjunto de actividades de aseguramiento de calidad a realizar junto con sus responsable y su respectivo sprint, basados en las actividades definidas en el estándar IEEE-730-2002.

Actividad	Sprint o Iteración	Responsable(s)
Definición de requerimientos del sistema	Sprint 2	Yin Cheng Liang Jose Antonio Salas Emanuel Alvarado
Definición de estándares de codificación	Sprint 2	Jose Antonio Salas
Definición de estándares de documentación	Sprint 2	Jose Antonio Salas
Definición de métricas	Sprint 1	Yin Cheng Liang Jose Antonio Salas Emanuel Alvarado
Construcción de pruebas unitarias	Sprint 1	Yin Cheng Liang Jose Antonio Salas Emanuel Alvarado
Evaluación de métricas	Sprint 4	Yin Cheng Liang
Evaluación de problemas	Al finalizar cada Sprint	Yin Cheng Liang
Evaluación del clima organizacional	Antes de comenzar cada Sprint	Yin Cheng Liang Jose Antonio Salas Emanuel Alvarado
Evaluación del diseño arquitectónico	Sprint 3	Emanuel Alvarado
Definir condiciones de aceptación	Sprint 3	Emanuel Alvarado

6 Pruebas Unitarias

1. Test Format: Prueba que se encarga de evaluar si el formulario es capaz de manejar imágenes en distintos formatos y convertirlos a png al momento de guardar en la base de datos.
2. Test Add Photo: Prueba encargada de evaluar el correcto funcionamiento de la base de datos. Se crea un objeto de tipo colección con una imagen asociada a este, luego se procede a hacer un save y se evalúa si ambos objetos se guardaron satisfactoriamente en la base de datos.
3. Test Size: Prueba encargada de evaluar si la base de datos respeta las dimensiones de la imagen original al guardar una instancia en el sistema. Para ello se carga una imagen de prueba llamada black.png con dimensiones de 275x183 pixeles y se compara con la original.
4. Test Form: Prueba encargada de evaluar si el formulario para la creación de una colección de imágenes se procesa adecuadamente. Es importante destacar que por medio de este formulario es que el cliente se comunica con el servidor para subir sus imágenes, por lo tanto se prueba se los campos de entrada de datos del template validan el formulario.

References

- [1] Sellares, T. (2015) The Adapt Pattern. Universitat de Girona. Recuperado el 12 de septiembre de 2018 de <http://ima.udg.edu/sellares/EINF->

ES1/AdapterToni.pdf