

Aseguramiento de la Calidad del Software

Artefactos de Análisis

Javier Contreras Muñoz, Carnet 2016093730

Randall Delgado Miranda, Carnet 2016238520

Sebastián Porras Fallas, Carnet 2015071017

November 15, 2018

1. Documento de Especificación de Requerimientos

1.1. Introducción

En la industria medicinal, se ha avanzado a pasos agigantados de la mano con el desarrollo tecnológico, a tal punto que se busca constantemente maneras de facilitar este trabajo y hacerlo más confiable, donde llegan a depender muchas vidas humanas. Un ejemplo de ello es la captura y procesamiento de imágenes biomédicas, que se ha utilizado para la identificación de células, al realizar diagnósticos automáticos a partir de imágenes y señales biomédicas.

Un sistema de software capaz de brindar una terapia personalizada para pacientes con alguna enfermedad que signifique un daño de las células (por ejemplo, cáncer) resulta sumamente provechoso, ya que se puede identificar de manera

más precisa el medicamento o químico necesario para cierto caso específico, tomando en cuenta que la única forma de realizar esto es mediante un análisis de reacciones celulares.

1.2. Propósito del sistema

La intención de desarrollo de este sistema es, principalmente, a partir de ciertas imágenes biomédicas de células capturadas por un microscopio especial, realizar un conteo de las células presentes en cada captura, y, de ser necesario, segmentar aquellas que podrían ser confundidas por una sola, y tomarlas por células separadas. Mediante el uso de métricas que identifiquen la precisión de dichas segmentaciones, se espera que los usuarios del sistema tengan la suficiente certeza de que los resultados propuestos son correctos, y dar una atención y tratamiento más especializados a cada paciente con base en esto, lo cual otorgará una precisión más alta que mediante el uso de otros métodos más mecánicos.

1.3. Alcance del sistema

El enfoque de este sistema es en el área biomédica, tomando en consideración el ambiente a donde pertenecen los clientes. Debido al propósito de dicho sistema mencionado en la sección anterior, se diseña orientándose al uso del personal del Laboratorio, y así facilitar el trabajo realizado sobre la identificación de células. No se tiene como descartada la utilización del sistema en otros ámbitos o con otro tipo de cliente, sin embargo, tampoco está dentro de lo contemplado actualmente.

1.4. Vista global del sistema

1.4.1. Contexto del sistema

Los doctores del Laboratorio de Quimiosensibilidad Tumoral de la UCR han trabajado durante cierto tiempo en la investigación de métodos para el tratamiento personalizado de pacientes con cáncer, donde generalmente el mayor problema se da a nivel celular. Sus estudios se han basado sobre la hipótesis de que, por cada paciente, hay un comportamiento diferente de sus células, así como sus reacciones ante ciertos factores de ambiente.

Tomando en cuenta que esta actividad constante se debe capturar en el momento justo de la reacción, por ejemplo, en la aplicación de un tratamiento quimioterapéutico, ellos se proponen a identificar los diferentes cambios que se dan por paciente, y así dar una atención más enfocada hacia cierto objetivo. El laboratorio cuenta con un microscopio capaz de tomar fotografías o capturas, por ende, su propósito es utilizar las imágenes capturadas de las células en momentos específicos y realizar un análisis sobre ellas, mediante la facilidad que proporciona un sistema de software.

1.4.2. Funciones del sistema

El sistema, en su fase final, deberá ser capaz de:

- Cargar una secuencia de imágenes de células capturadas que se encuentren almacenadas en la dirección que provea el usuario.
- Realizar un conteo de dichas células en las imágenes seleccionadas (por cada imagen).
- Generar un informe con el centroide y área de cada elemento o célula identificada, en un archivo de valores separados por comas (CSV).

- Permitir en la interfaz gráfica la navegación entre imágenes y resultados, así como la selección de un grupo de imágenes para procesar (todo el grupo a la vez, además de poder hacerlo una por una).
- Mostrar al usuario el tiempo de ejecución por cada imagen y/o conjunto de imágenes que se procese.
- Cargar un archivo de ground truth con los blobs de los objetos marcados, para analizar la precisión de la segmentación mediante una métrica indicada.

1.4.3. Características del usuario

Los usuarios son doctores pertenecientes al personal del Laboratorio de Quimiosensibilidad Tumoral de la Universidad de Costa Rica, enfocados en el área biomédica. Ellos cuentan con equipo para la captura de imágenes a nivel celular, específicamente microscopios electrónicos de fluorescencia capaces de tomar fotografías, y esperan, mediante las imágenes tomadas y el sistema desarrollado, realizar un análisis sobre las células con las que estén trabajando.

1.5. Requerimientos funcionales

Se espera que la aplicación como tal cumpla las funcionalidades mencionadas en el punto 3.2, de manera más general:

- Cargar imágenes (una o por lote) de una dirección dada por el usuario.
- Contar y segmentar las células de las imágenes.
- Identificar los centroides y áreas por cada célula identificada y guardar un informe a partir de ello.
- Permitir ver las imágenes y sus resultados de manera sencilla y cercana.

- Llevar los tiempos de ejecución por cada conteo realizado.
- Permitir el uso de diferentes métricas para medir la precisión de las segmentaciones por imagen.

El enfoque es más global debido a su carácter funcional, tomando en cuenta atributos de calidad como la necesidad de una alta precisión o exactitud de los procesos y análisis, la adecuación de las funcionalidades respecto al contexto donde se desarrollará el sistema, y un enfoque no tan alto de desarrollar un sistema seguro.

1.6. Requerimientos de usabilidad

Al ser un sistema desarrollado para un cliente que podría no estar tan familiarizado con algún tipo de tecnología, se busca que dicho software resulte de fácil aprendizaje y entendimiento, y sea lo más intuitivo posible, con un fácil ingreso de entradas y una interfaz de complejidad baja. De esta manera, los usuarios no deberán invertir mucho tiempo entendiendo sus partes, y podrán realizar sus labores de manera mucho más rápida, evitando una alta curva de aprendizaje.

1.7. Requerimientos de rendimiento

Como el sistema involucra un análisis de grandes cantidades de información, pudiendo llegar a miles de imágenes, y un uso constante de métricas, el cliente espera un software bastante eficiente. Se dará un gran enfoque al mayor aprovechamiento de recursos (es decir, un uso racional que no implique un consumo excesivo o una disminución en el rendimiento del ordenador), y a ser posible, mantener los tiempos de respuesta al mínimo posible. Además, se visualiza un sistema en el que se necesiten pocas y sencillas entradas para obtener una gran

cantidad de datos de salida, con el fin de facilitar las labores realizadas por el personal del laboratorio.

1.8. Interfaces del sistema

Al ser un sistema Web, deberá haber una interfaz presente entre el software y los protocolos HTTP, para su disponibilidad en la red. Además, y en conjunto con lo anterior, éste debe de poder ser visualizado en un navegador para su ejecución por parte de los usuarios. Deberá tener una interacción con los directorios de la computadora donde se ejecute, para el manejo de imágenes, archivos de reporte y de métricas.

1.9. Operaciones del sistema

1.9.1. Requerimientos de interacción humano-sistema

Por la naturaleza del sistema, la interacción entre los usuarios y el sistema es bastante segura, ya que sólo es necesario para éste indicar la función que desea ejecutar, ya sea el análisis de imágenes, generación de informes o aplicación de métricas. Por ende, por parte del usuario sólo se requiere un conocimiento básico de la utilización del software, y que se asegure de estar en un lugar con conexión a Internet.

1.9.2. Mantenibilidad

Como el sistema se entregará de forma completa, y dado el tipo de software que se está desarrollando, el mantenimiento es un factor importante, incluso si no es de constante frecuencia. En caso de presentarse algún error inesperado dentro del sistema, el cliente podrá contactar a los desarrolladores para dar la atención requerida. Por otro lado, si se necesita una funcionalidad adicional, se

podrá también implementar siempre y cuando no implique un problema para otra función ya agregada.

Se debe considerar que dicho sistema deberá ser de fácil probabilidad (capacidad de ser probado) para asegurarse de que las funciones se implementaron de manera correcta, tener una alta estabilidad con el fin de no causar inconvenientes o retrasos a los usuarios, e idealmente permitir el análisis del software como tal, para diagnosticar problemas o carencias dentro de su funcionamiento, como lo podría ser el rendimiento o la utilización de los recursos.

1.9.3. Confiabilidad

Para el personal del laboratorio, resulta de gran importancia que el sistema resulte de gran confianza a la hora de utilizarlo, es decir, tomar en cuenta que se requiere un sistema con poca o ninguna presencia de fallas, y en caso de que inevitablemente se presente alguna, sea de manera transparente para el usuario, que no interfiera con la ejecución del programa y sea capaz de recuperarse y volver a su estado original. Si el sistema desarrollado cumple estas características, su uso resultará más agradable para la población meta y no retrasará sus operaciones.

1.10. Modos y estados del sistema

Básicamente, el sistema puede tener los siguientes estados:

- Se tiene un estado inicial, donde se encuentra una pantalla principal en la cual el sistema está a la espera de que el usuario indique las imágenes a cargar. El sistema en este momento está inactivo.
- El sistema ya ha cargado una imagen, y puede permanecer cargando imágenes hasta que el usuario indique que desea continuar a segmentar.

- El sistema se encuentra segmentando las imágenes, por ende, no se procederá a otro estado hasta que se finalice la segmentación (o se vea interrumpido el proceso).
- El sistema finalizó la segmentación y muestra los resultados obtenidos de dicho proceso, además, por debajo ha generado un archivo de informe a partir de esos resultados. El usuario puede aplicar las métricas de precisión en este estado, o volver al estado inicial.
- El sistema recibió una indicación de aplicar una métrica, por ende, se encuentra en proceso de calcular qué tan precisa fue la segmentación realizada en el lote de imágenes.
- El sistema aplicó la métrica que el usuario indicó, y en la pantalla muestra la precisión entre el resultado que se obtuvo de segmentar las imágenes, y las imágenes originales. El usuario puede decidir aplicar otra métrica, o volver a la pantalla principal.

1.11. Requerimientos físicos

A pesar de que el producto a entregar se desarrolla en un ambiente virtual, se recomienda una computadora de un procesador medianamente poderoso, debido a que un proceso de segmentación de células requiere recursos computacionales de buena calidad.

1.12. Requerimientos de adaptabilidad

El sistema está diseñado de forma que perdure en los equipos del laboratorio, y está planeado para utilizar cualquier extensión de imagen. Además, su estructura permite que sea posible (si es necesario) añadir más métricas que podrían ser relevantes para medir la precisión de la segmentación. También puede llegar a generar informes en algún otro formato si así se deseara.

1.13. Condiciones del ambiente

El ambiente general donde el sistema se ejecutará es en un laboratorio, por lo cual no se perciben problemas o condiciones que se puedan encontrar de parte del ambiente, ya sea a nivel físico o empresarial. Al ser parte del ámbito científico, se podrían presentar cambios o información adicional sobre el tratamiento de las células, y podría ser necesario cambiar alguna función del sistema.

1.14. Seguridad del sistema

- Puesto que el desarrollo del proyecto es Web, el sistema debe evitar el acceso a la información generada durante el procesamiento de datos.
- El sistema, al ser de un ámbito tan específico, no tiene grandes problemas con la seguridad, puesto que no se maneja información sensible, sin embargo es necesario invertir recursos en este aspecto.

1.15. Manejo de la información

- El sistema ha de ser capaz de procesar imágenes individuales, así como lotes numerosos de imágenes.
- El sistema deberá exportar la información obtenida en un archivo de formato CSV.
- El sistema debe de procesar información con formato y extensiones válidas de imagen.
- El sistema deberá manejar archivos ground truth para el uso de las métricas.

1.16. Regulaciones y lineamientos

Aunque no se reflejan políticas establecidas que puedan interferir en la correcta ejecución del sistema, se tiene presente que los avances científicos con el paso de los años pueden cambiar el contexto biomédico sobre el cual se trabaja. Por ende, se da por entendido que la información manejada sobre las células y el equipo con que se trabaja está sujeto a cambios.

1.17. Sustento del ciclo de vida del sistema

La naturaleza de este tipo de sistema es ir evolucionando conforme se identifican sus requerimientos, y avanzar en diferentes etapas cada funcionalidad por orden de prioridad. Se mantiene además un contacto cercano con el cliente, en caso de algún cambio en la especificación del sistema requerido. Habiendo tomado en cuenta las etapas de análisis y diseño del software, se procede a desarrollarlo como tal, realizar diferentes pruebas sobre éste, y por último hacer su entrega al cliente para su respectivo mantenimiento en un futuro.

1.18. Entrega del sistema

Por el tipo de sistema que se está desarrollando, la entrega al cliente se realizará mediante medios digitales de manera presencial, con toda la estructura de archivos necesarios para la ejecución del programa, de manera que también tengan acceso a la capa del servidor.

1.19. Verificación

La funcionalidad del sistema se comprueba en correctitud por medio de un conjunto de pruebas unitarias, que evalúen diferentes casos posibles que pueden

ocurrir durante la ejecución de una función específica. El sistema, evidentemente, deberá ser de carácter probable desde un inicio, para una ejecución sencilla de cada una de las características.

1.20. Suposiciones y dependencias

- Se asume que el sistema será utilizado únicamente para un contexto biomédico.
- Se asume que los recursos pertenecientes a los usuarios tienen una capacidad intermedia de procesamiento.
- Se asume que el Sistema Operativo usado por los usuarios es Windows.
- Se asume que los usuarios tienen cierto grado de conocimiento informático.

1.21. Prioridades de las necesidades del sistema

- Cargar imágenes (una o por lote) de una dirección dada por el usuario: Alta, ya que si el sistema no es capaz de cargar imágenes, que es la entrada principal, no podrá realizar ningún otro proceso; por ende, se podría decir que es la cabeza del sistema.
- Contar y segmentar las células de las imágenes: Alta, debido a que es el principal propósito según lo indicado por los clientes (la razón de ser del software). Incluso si carga las imágenes, debe ser capaz de saber qué hacer con ellas.
- Identificar los centroides y áreas por cada célula identificada y guardar un informe a partir de ello: Media, ya que los clientes desean un reporte de los análisis ejecutados, sin embargo, hay requerimientos de mayor prioridad, y el hecho de poder realizar su trabajo sin ello lo prioriza menos.

- Permitir ver las imágenes y sus resultados de manera sencilla y cercana: Media, ya que le sería de mucha utilidad a los miembros del laboratorio que los resultados generados de la segmentación sean legibles y estén bien ordenados, incluso si no es indispensable.
- Llevar los tiempos de ejecución por cada conteo realizado: Baja, ya que su no presencia no interfiere en el funcionamiento correcto del sistema; aunque esto no significa que no deba incluirse en el sistema, su costo es menor al no agregarse.
- Permitir el uso de diferentes métricas para medir la precisión de las segmentaciones por imagen: Alta, ya que, además de que el cliente dio un énfasis especial en la presencia de estas métricas, es necesario tener a la mano una serie de herramientas que midan qué tan preciso es el sistema que se va a utilizar.
- Permitir un fácil aprendizaje y entendimiento del software, así como intuitividad de la interfaz: Media, ya que, aunque es de suma importancia que el cliente pueda usar el sistema de manera fácil e intuitiva, se espera que dicho usuario ya tenga cierto conocimiento previo que le facilite la utilización del software.
- Desarrollar un sistema de bajos tiempos de respuesta y máximo aprovechamiento de recursos: Alta, por la naturaleza del sistema de generar resultados de manera eficaz y eficiente, es decir, utilice de manera balanceada los recursos dispuestos por la computadora.
- Crear un software en el cual se puedan realizar pruebas de funcionalidad y ejecución, así como flexibilidad para mantenerlo: Media, ya que los clientes podrían requerir agregar una funcionalidad nueva conforme pasen los años, o incluso modificar su metodología actual para darle un diferente enfoque. Sin embargo, esto no gana una prioridad alta ya que esto no ocurrirá de manera muy frecuente.

- Implementar un sistema con poca o ninguna presencia de fallas, y un manejo transparente de ellas: Alta, debido a que se necesita generar confianza en los usuarios para que usen el software sin problema alguno. De esta manera, se evitará a toda costa que el sistema falle, y en caso de alguna falla inevitable o fuera del alcance, se maneje de forma que no interrumpa en gran medida (o en ninguna) las actividades de los usuarios.
- Desarrollar un software pensado para un ordenador de potencia media en procesador: Baja, ya que el trabajo de los desarrolladores involucra implementar los algoritmos de la forma que asegure más rendimiento posible. El ambiente donde se desarrollará dicho software queda fuera del alcance, por ende, cumplir los requerimientos de rendimiento resuelven también los físicos.
- Crear un sistema adaptable, con facilidad para agregar o eliminar métricas, y manejo de diferentes extensiones: Media, ya que la variación de métricas puede ser un factor de presencia constante y se adaptará de acuerdo con lo que los investigadores deseen. El reconocimiento de extensiones de archivos es menos probable a cambiar, sin embargo, se debe tener la facilidad para modificarlas si fuera necesario.
- Implementar un producto de alta seguridad, que evite el acceso durante el procesamiento de imágenes y conexión de terceros: Baja, ya que los permisos de acceso siempre resultan relevantes para cualquier tipo de software. No obstante, para este caso, la sensibilidad de la información es baja y no es tan estricta la inversión de recursos en este campo, pero no significa que se deba dejar de lado su implementación.

2. Métricas utilizadas

Métricas de Aseguramiento de Calidad

Característica	Métrica	Nivel requerido
Precisión	Cantidad de resultados correctos respecto al umbral establecido	95 % de resultados correctos
Seguridad	Cantidad de riesgos identificados y su severidad	0-3 riesgos, de severidad baja/media
Conveniencia	El porcentaje de funciones directamente relacionadas con la funcionalidad principal del sistema	90 % de las funciones
Madurez	Cantidad de fallas detectadas en un plazo de ejecución determinado	0-3 fallas detectadas en una hora
Tolerancia a fallos	Porcentaje de recuperaciones exitosas del sistema tras n cantidad de fallos	95 % de fallas recuperadas exitosamente
Recuperabilidad	Porcentaje de datos recuperados con éxito luego de una falla del sistema	90 % de recuperación de datos
Entendibilidad	Cantidad de letras promedio que tienen las etiquetas de las funciones	10-15 letras aproximadamente
Aprendibilidad	Cantidad de horas que le toma al doctor recorrer el software por completo	1,5 horas máximo
Operabilidad	Cantidad y tipo de funciones ejecutadas por el usuario en cierto tiempo	10 funcionalidades realizadas en una hora, de tipo variado

Utilización de recursos	Cantidad de memoria RAM que utiliza el programa	100 MB de RAM como máximo
Comportamiento del tiempo	Cantidad de imágenes procesadas por tiempo definido	50-60 imágenes procesadas en un minuto
Eficiencia	Cantidad de datos de entrada necesarios contra sus salidas respectivas	5 entradas para 10 datos de salida por imagen
Probabilidad	Total de pruebas exitosas luego de modificar el software	95 % de pruebas exitosas
Analizabilidad	Cantidad de líneas comentadas en el código	10 líneas por función implementada
Estabilidad	Tiempo de continuidad del software en ejecución sin interrupciones	2 horas aproximadamente sin presentar fallos
Adaptabilidad	Cantidad de entornos en los que el sistema puede ejecutar de forma completa	Entre 5 y 10 entornos diferentes para ejecutarse
Coexistencia	Cantidad de horas para implementar un sistema externo al principal	2-3 horas de implementación
Instalabilidad	Tamaño del descriptor de despliegue (archivos de configuración) en líneas/elementos	200 líneas para archivos de texto, 10 elementos/atributos para archivos XML

Cuadro 1: Métricas para los Atributos de Calidad

3. Diagrama de Componentes del Sistema

