

Instituto Tecnológico de Costa Rica  
Escuela de Ingeniería en Computación  
Curso de Aseguramiento de la Calidad del Software  
Avance No. 1 de Proyecto:  
Documento de Especificación de Requerimientos (SyRS)

Francisco Monge Zúñiga (2013029434)  
Andrés Gutiérrez Salas (201223823)  
Joseph Salazar Acuña (2015100516)

September 19, 2018

## 1 Introducción

Los dispositivos de captura y procesamiento de imágenes biomédicas, como resonancias magnéticas, radiografías y tomografías computarizadas, cobran cada vez mayor protagonismo en los procesos de toma de decisiones para los profesionales de la salud. Más recientemente, la industria biomédica ha desarrollado interesantes sistemas de software que realizan diagnósticos automáticos a partir de imágenes y señales biomédicas.

En nuestro caso específico, se abordará un problema de gran interés: la segmentación de imágenes digitales tomadas de un tejido de glioblastoma, para la detección de células cancerígenas. Proceso que, de realizarse de manera manual, toma demasiado tiempo por parte de los microbiólogos encargados.

En el presente Documento de Especificación de Requerimientos (SyRS) encontrará una serie de funcionalidades, atributos y condiciones que el sistema de software a desarrollar debe cumplir, de cara a resolver satisfactoriamente las necesidades de quienes lo van a utilizar.

## 2 Propósito del sistema

Se desea proveer de un sistema de software a los microbiólogos encargados de realizar un registro de los cambios en las células cancerígenas. Dichas células son observadas en imágenes digitales, que son tomadas de un vídeo generado por un microscopio basado en fluorescencia.

El proceso actual consiste en tomar cada una de las imágenes e identificar, manualmente, cuáles de los elementos presentes en la imagen forman parte de las células cancerígenas y cuáles no. Además, se debe llevar registro de cuantas células existen en cada imagen. Este proceso debe repetirse con cada imagen. En algunos casos puede alcanzarse hasta las 170 000 imágenes por procesar, siendo un número excesivamente alto cuando de un proceso manual se trata.

Por tanto, el sistema a desarrollar tiene como propósito hacer que este proceso de segmentación de imágenes sea más efectivo, reduciendo así los tiempos de análisis del conjunto de células, permitiendo a los microbiólogos tomar decisiones más rápidas y consecuentemente poder brindar el tratamiento específico para cada paciente de una manera más oportuna.

En ese sentido, lo que se desea lograr es:

- a) Desarrollar un sistema de software que preprocese, segmente y detecte células cancerígenas en tejidos de glioblastoma en imágenes digitales, tomadas por un microscopio basado en fluorescencia.
- b) Generar un informe con la cantidad de células cancerígenas y la posición de cada una en cada imagen.

## 2.1 Alcance del sistema

Se identificará el sistema de software a desarrollar con el nombre: CellTracker-V1

Se creará una página utilizando tecnologías web. En dicha página web el usuario será capaz de cargar una o más imágenes (imágenes que muestran la actividad celular de un tejido y que van a ser procesadas) seleccionadas de un directorio elegido por el usuario mismo de manera dinámica. Los formatos de imagen que deben admitirse son: .jpg y .png, siendo estos los formatos de imagen más utilizados a nivel global.

Una vez que las imágenes fueron cargadas al sistema, el usuario podrá solicitar al sistema procesar imágenes una a una o bien el conjunto completo.

Para cada imagen el sistema debe primero realizar una segmentación, es decir, elegir por medio de algoritmos computacionales, cuáles píxeles de la imagen corresponden a una célula y cuáles no. Debe entonces producir una imagen binaria, en la que solo habrá dos colores: blanco para los píxeles que corresponden a una célula cancerígena y negro para los que no. Esto facilitará la visualización de las células y los pasos siguientes que el sistema debe implementar.

Una vez que la imagen ha sido completamente segmentada, el sistema deberá contar cuántas células cancerígenas se encuentran presentes en la imagen y mostrar dicho número por pantalla. Opcionalmente se puede colorear y etiquetar cada una de ellas (asignarles un código que las identifique).

Por último, se debe generar un informe en formato de archivo .csv que incluya, para cada célula encontrada, su centroide y su área.

## 3 Visión general

### 3.1 Contexto

CellTracker-V1 será instalado en el Laboratorio de Quimiosensibilidad tumoral, de la Universidad de Costa Rica. En dicho laboratorio se desarrollan proyectos de investigación relacionados con la terapia personalizada para pacientes con cáncer.

De manera más específica, el proyecto se alinea con la tarea que realizan una serie de microbiólogos, de tomar un lote de imágenes médicas para rastrear el comportamiento de las células cancerígenas que se muestran en las mismas. Lo que desean es optimizar la aplicación del tratamiento médico para casos específicos, haciendo que este proceso de identificación sea preciso y más eficiente que con las técnicas actuales.

Por otra parte, la infraestructura computacional del laboratorio es modesta.

Por último, el laboratorio no cuenta actualmente con ninguna herramienta digital que facilite, de manera total o parcial, las necesidades de los usuarios, por lo que el desarrollo de CellTracker-V1 deberá partir desde cero.

### 3.2 Funcionalidades del sistema

Se muestran a continuación una serie de funcionalidades, condiciones y restricciones que el sistema debe comprender:

- Iniciar sesión: cada uno de los usuarios deberá autenticarse para acceder al sistema.
- Pantalla principal: contará con la opción de cargar las imágenes, procesar las mismas, observar el resultado del procesamiento y generar el informe.
- Carga de imágenes: se debe proveer una opción en la que el usuario pueda elegir un lote de imágenes que van a ser procesadas desde un directorio cualquiera en la computadora. Que soporte formatos de imagen .jpg y .png.
- Procesar imágenes: deberá contener un botón que le permita al usuario iniciar el proceso de segmentación de las imágenes.
- Segmentación: el sistema debe generar una imagen binaria en la que sea más sencillo identificar cuáles píxeles corresponde a una célula cancerígena y cuáles no.
- Resultado de segmentación: el usuario podrá ser capaz de visualizar las imágenes ya segmentadas.
- Conteo: el sistema debe mostrar en pantalla cuántas células se encuentran presentes en cada imagen.
- Etiquetado: el sistema deberá etiquetar cada una de las células presentes en una imagen.
- Informe: se debe tener una opción que le permita al usuario generar un informe, en formato de archivo .csv con el resultado del procesamiento. Es decir, para cada imagen, las células encontradas, centroide de cada una y su área.
- Cerrar sesión: el usuario deberá cerrar la sesión una vez que deje de utilizar el sistema.
- Se debe asegurar un procesamiento eficiente, pues se cuenta con lotes de imágenes bastante grande. Se requiere mejorar los tiempos de procesamiento respecto de la técnica manual.
- Se debe asegurar que el sistema no consuma demasiados recursos, pues la infraestructura computacional sobre la que será ejecutado es modesta.
- Se debe contar con una interfaz amigable con el usuario y que permita el acceso a todas las funcionalidades de una manera sencilla.

### 3.3 Características del usuario

Los usuarios finales del sistema serán los doctores microbiólogos del Laboratorio de Quimiosensibilidad tumoral, de la Universidad de Costa Rica. Dichos doctores poseen conocimientos básicos en computación y serán los encargados de cargar las imágenes, solicitar el procesamiento y generar informes. No se conoce el número de usuarios de este tipo que utilizará el sistema.

En caso de ser necesario, el mantenimiento del sistema será ejecutado a lo interno por el mismo equipo que lo desarrolló.

## 4 Requerimientos Funcionales

Preprocesar, segmentar y detectar células en tejido de glioblastoma en imágenes digitales brindando reportes sobre los resultados.

- REQ-1: Cargar una secuencia de imágenes almacenada en la dirección provista por el usuario.
- REQ-2: Segmentar cada una de las imágenes provistas y mostrar el resultado.
- REQ-3: Contabilizar los objetos de interés en las imágenes (células).
- REQ-4: Generar un informe que indique por cada objeto encontrado, su centroide y su área, en formato.csv.
- REQ-5: Iniciar y cerrar sesión de usuario.

## 5 Requerimientos de Usabilidad

El sistema debe proveer una entendibilidad alta, esto quiere decir que debe ser sencillo para el usuario entender cuál es la función específica que realiza cada opción (botones, selector de imágenes, visualizador de resultados). Con esto se desea evitar confusiones en el uso del software, que lleven a los usuarios a tardar más tiempo del esperado en poner en marcha el sistema.

También, el sistema debe garantizar una alta operabilidad. Esto significa que el usuario debe tener control total de las opciones para las que el sistema fue construido. Además, de ser capaz de operarlo de una manera correcta y sencilla, por ejemplo eliminando el uso de muchos botones o pestañas innecesarias.

Por último, se espera que el software sea atractivo de utilizar, que el usuario tenga facilidad de encontrar las funciones que requiere de manera ordenada, por ejemplo, siguiendo la secuencia lógica en que aplican el procedimiento de manera manual. Procurar que todas las opciones sean altamente visibles.

## 6 Requerimientos de Desempeño

Se debe asegurar un sistema eficiente. En este contexto significa que el sistema debe disminuir considerablemente los tiempos de procesamiento respecto de la técnica manual. En otras palabras, el tiempo de procesamiento de cada una de las imágenes tiene que ser considerablemente menor que si un microbiólogo lo hiciera manualmente. Suponiendo entonces que si el tiempo tomado por cada imagen es menor, el tiempo total por lote de imágenes también será menor.

Se debe procurar un uso de recursos modesto, puesto que el equipo de hardware que soportará el sistema también es modesto. En ese sentido debe considerarse que el equipo no puede sucumbir cuando el sistema es ejecutado. Debe permitir que otras aplicaciones se sigan utilizando. Por tanto, debe tenerse un balance entre rapidez de procesamiento y uso de recursos.

## 7 Interfaces del sistema

A nivel de interfaces, el sistema no interactúa directamente con otros sistemas de software. Simplemente debe garantizar que el selector de imágenes pueda acceder al sistema de archivos del sistema operativo. Y por otra parte, debe ser capaz de generar un archivo .csv, por lo que es necesario que cumpla con los estándares de escritura de los mismos.

La interfaz con el usuario debe proveer todo lo necesario para cumplir con las especificaciones que este documento detalla. En resumen, la interfaz de usuario debe ser amigable, esto es que sea visualmente atractiva, ordenada, que le permita al usuario acceder a las funciones de una manera sencilla y rápida. Que muestre en cada sección lo esperado en dicha sección. No debe contener secciones, paneles, botones o pestañas innecesarias, pues se desea maximizar la usabilidad y operabilidad del software.

Se debe usar un estándar similar al que utilizan otras páginas web, de manera que el usuario se sienta familiarizado de alguna forma con lo que ve, mejorando así la curva de aprendizaje del software.

## 8 Funcionamiento del sistema

### 8.1 Requerimientos de Integración

En este sistema no existen requerimientos de integración, ya que ninguno de los usuarios que utilizarán el software va a realizar una tarea crítica, donde tenga alguna responsabilidad directa en su ejecución. Esto se traduce en que la posibilidad de que el factor humano incurra en eventos riesgosos es muy baja, por no decir que nula.

## 8.2 Mantenibilidad

Se debe asegurar un sistema con alta cambiabilidad. Puesto que, aunque no se vislumbra que el sistema llegue a tener muchos cambios, se debe tomar en cuenta que existen ciertos parámetros o formas (algoritmos) en que se realizan ciertas operaciones que podrían cambiar. Incluso, podría no cambiar la forma en que hacen las operaciones, sino únicamente parámetros o valores de ajuste, en busca de conseguir resultados diferentes. Así que el sistema debe ser flexible ante posibles cambios.

Se debe asegurar un sistema con alta estabilidad. El sistema debe ser capaz de introducir cambios sin que esto suponga que alguna funcionalidad se vea afectada o presente algún fallo que dé resultados incongruentes. Además, por el tipo de sistema que es, se espera que sea tan estable que se modifique casi que solo para agregar funcionalidades o para optimizar las ya existentes y no para la corrección de errores.

Se debe asegurar un sistema con alta testeabilidad. El software debe poseer un alto grado de testing en cada módulo de funcionamiento, es decir, cada tarea debe ser probada para un funcionamiento correcto. Ya que por el tipo de sistema que es no se puede correr el riesgo de que alguna función esté dando datos incorrectos y que lleven a conclusiones erróneas.

## 8.3 Fiabilidad

No se puede realizar un requerimiento cuantitativo explícito dado que no se tiene certeza de cuáles son los umbrales de fiabilidad en los que se basa el método actual. Por tanto, no se conoce cuánto es lo mínimo esperado en busca de mejorar o al menos igualar la fiabilidad del proceso actual. Más adelante deberá especificarse.

## 8.4 Modos y Estados del Sistema

En este sistema no existen requerimientos de modos y estados, ya que solo existe un único estado que es en el cual va estar siempre el sistema. Dicho estado es el que contempla todas las tareas que va a realizar el usuario, contempladas en este documento.

# 9 Características Físicas

## 9.1 Requerimientos Físicos

El sistema no contiene requerimientos físicos, ya que al ser desarrollado sobre una plataforma web no se necesita una instalación o demás consideraciones que si hubiera requerido una plataforma de escritorio.

## 9.2 Requerimientos de Adaptabilidad

No se ha confirmado que el sistema deba escalar en cuanto a cantidad de usuarios o laboratorios en los que va a ser utilizado. Sin embargo, en caso de que el sistema se quiera poner en marcha en otros laboratorios, se recomienda entonces la incorporación de un servidor con mayor potencia, que permita que varios usuarios puedan llevar a cabo sus tareas sin ver afectado el desempeño. No obstante, refiere más a un tema de equipo de hardware, puesto que el sistema de software mantiene sus características.

# 10 Condiciones Ambientales

El sistema será utilizado en el Laboratorio de Quimiosensibilidad tumoral, de la Universidad de Costa Rica, del cual no se conoce mucha información. No se conoce si el equipo está expuesto a condiciones como polvo, mucho frío o mucho calor, líquidos, electromagnetismo, etc. Sin embargo, nuevamente refiere a condiciones que podrían afectar el equipo de hardware, no el sistema de software.

# 11 Seguridad del Sistema

Se implementará un sistema de login para que los usuarios puedan ingresar a la página y que no cualquier persona tenga acceso a las funciones de esta, además de las imágenes de los pacientes que son confiden-

ciales. La contraseña deberá contener letras mayúsculas y minúsculas, números y un largo mínimo de 8.

## 12 Administración de la Información

Al sistema se le deberá proporcionar como valores de entrada todo el conjunto de imágenes que van a querer ser procesadas por el sistema. El tipo de imagen que va a soportar es PNG y JPG. No existe una cantidad mínima o máxima de imágenes a proporcionar. Las imágenes proporcionadas serán almacenadas en el sistema mientras se realiza el procesamiento.

El sistema dará como salida un informe por cada imagen en donde se muestre un centroide y el área que conforma a la célula. Dicho informe será uno por cada muestra de imagen dada en la entrada. Además, puede que el sistema etiquete y coloree cada célula.

## 13 Políticas y Regulaciones

Existe la obligación de manejar las imágenes de manera anónima, es decir no se debe saber de qué persona corresponde dichas imágenes a ser procesadas. Además se debe asegurar que tampoco las imágenes tengan meta-datos que implique contar con información personal de las personas.

También se debe respetar todas las políticas y regulaciones que rigen en el centro donde se va a utilizar el sistema, en este caso el Laboratorio de Quimiosensibilidad tumoral, de la Universidad de Costa Rica.

## 14 Sostenimiento del Ciclo de Vida del Sistema

1) Evaluación de conformidad del diseño respecto a los requerimientos.

a) Actividad:

Realizar reuniones presenciales con la contraparte donde se verifique que el diseño preliminar propuesto concuerde con los requerimientos establecidos en el Documento de Especificación de Requerimientos (SyRS).

b) Objetivo:

Verificar si el diseño de la aplicación cumple con las expectativas y puntos establecidos, avances, del sistema.

c) Cronograma:

Una reunión durante cada sprint.

2) Evaluación de conformidad de la implementación respecto al diseño.

a) Actividad:

Realizar revisiones internas con el equipo de desarrollo donde se verifique que el diseño establecido concuerde con el código desarrollado para las funcionalidades específicas. El encargado de realizar dicha revisión no puede ser el miembro que desarrolló.

b) Objetivo:

Verificar que el código desarrollado por los miembros del equipo cumple con el diseño establecido y, por ende, con los requerimientos establecidos.

c) Cronograma:

Se debe realizar un pull request cada vez que se implementa una tarea.

3) Evaluación de conformidad de la implementación respecto a los requerimientos.

a) Actividad: Realizar reuniones presenciales con la contraparte donde se verifique que la implementación de las funcionalidades concuerde con los requerimientos establecidos en el Documento de Especificación de Requerimientos (SyRS).

b) Objetivo:

Verificar que la implementación de las funcionalidades cumple con las expectativas del cliente.

c) Cronograma:

Una reunión durante cada sprint.

4) Cumplimiento de estándares de codificación.

a) Actividad:

Realizar chequeos cada vez que se hacen pull requests al Github con las herramientas previamente establecidas en el apartado de estándares de codificación.

b) Objetivo:

Cumplir con los estándares para así asegurar la calidad del software.

c) Cronograma:

Cada vez que se realiza un pull request en Github.

5) Construcción y revisión de tests unitarios y de integración.

a) Actividad:

Realizar una serie de tests unitarios para verificar que lo implementado cumple con lo establecido, de una manera correcta y así disminuir la aparición de errores.

b) Objetivo:

Asegurar que las funcionalidades implementadas realizan su función específica de una forma correcta y así disminuir la aparición de errores.

c) Cronograma:

Se construyen previo a iniciar al desarrollo de una funcionalidad. Se prueban una vez finalizado el desarrollo de la funcionalidad.

## 15 Empaquetado, Manejo, Compra y Transporte

El sistema no cuenta con ningún tipo de empaquetado, manejo y transporte, ya que su distribución no será de manera física. La obtención del software será accediendo al dominio del sistema, es decir, a través de una página web. Tampoco implica alguna compra, ya que el desarrollo del sistema es de código libre.

## 16 Verificación

Para las funcionalidades del sistema se contemplan las siguientes métricas que harán la verificación de cada uno de ellas.

- REQ-1: Se le desplegará al usuario todas las imágenes que ha seleccionado en la ruta específica, así el usuario puede verificar si las imágenes brindadas son las correctas y que la cantidad de imágenes que fueron cargadas es la misma cantidad que se brindó en su inicio.
- REQ-2: Se utilizará la métrica del coeficiente de Dice o de Sørensen, el cual se puede usar para evaluar la similitud estructural entre dos imágenes binarias.
- REQ-3: Contabilizar los objetos de interés en las imágenes (células).

- REQ-4: Se le desplegará al usuario la cantidad de informes que el sistema generó y dicho número debe de calzar con la cantidad de imágenes cargadas en el inicio.
- REQ-5: Cuando el usuario inicia sesión se le mostrará en la página.

## **17 Suposiciones y Dependencias**

El sistema no tiene que hacer ninguna suposición ni depender de algo más para poder funcionar.