

Aseguramiento de la Calidad del Software: Avance 2, proyecto I, y II semestral

M. Sc. Saúl Calderón Ramírez
Instituto Tecnológico de Costa Rica,
Escuela de Ingeniería en Computación,
PAttern Recognition and MACHine Learning Group (PARMA-Group)

20 de septiembre de 2016

En el presente avance, se desarrollarán las funcionalidades críticas del proyecto, usando criterios de aseguramiento de la calidad.

Fecha de entrega: 8 de Octubre.

Entrega: A través del TEC digital.

Valor: 5 % del proyecto I semestral.

Parte I

Avance 2 del proyecto

Después de ver los resultados del PoC, y una reunión del equipo técnico, junto con miembros del comité directivo del Boca Juniors, han decidido iniciar el proyecto. El equipo de informáticos interno del Boca Juniors, a partir del documento de **necesidades** (enunciado) del proyecto ha dispuesto las siguiente necesidades para los dos proyectos como prioritarias:

1. Necesidades prioritarias para el proyecto 1

1. Cargar un video digital almacenado en la dirección provista por el usuario.
2. Descargar o en su defecto generar el video con los «blobs» de los jugadores detectados.
 - a) Para este avance puede implementar únicamente la segmentación o únicamente la clasificación por color, usando contornos manualmente segmentados.

- b) Descargar el video con los «blobs» de los jugadores o los blobs manualmente segmentados, clasificados por color.
- 3. Calcular métrica de precisión de la segmentación o clasificación: Cargar un conjunto de imágenes con los «blobs» manualmente marcados por color,
 - 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.

- 1) Para el caso de implementarse la clasificación no supervisada, usar una métrica basada en la tasa de fallos y aciertos («F1 score», precision y «recall»).

2. Necesidades prioritarias para el proyecto 2

- 1. Cargar un video digital almacenado en la dirección provista por el usuario.
- 2. Generar un archivo en formato *xml*, *yaml*, o *csv* que especifique en el frame en el que se detecta un corte.
 - a) Implementar el cálculo de la función de distancias entre histogramas consecutivos, usando la distancia de Bhattacharyya.
 - b) Implementar el método de umbralización de tal funcional, para detectar los cortes.
- 3. **Métrica de exactitud en los resultados:** Cargar un archivo de *ground truth* en el que el número de frame en el que sucedió un corte esté manualmente marcado, y que permita además cuantificar la cantidad de falsos positivos y negativos en la detección de cortes, cuantificando los falsos positivos y negativos.

3. Requerimientos para el «sprint» fijados por el cliente (Boca Juniors y su departamento de IT)

El equipo interno del Boca Juniors ha fijado los requerimientos de calidad para el siguiente «sprint».

1. De acuerdo a las necesidades prioritarias fijadas por el Boca Juniors, actualizar la prioridad de los requerimientos asociados en el documento SyRS (debe adjuntar el documento SyRS actualizado).
2. Cada unidad de diseño debe estar asociada a al menos un requerimiento (del documento SyRS entregado anteriormente).
 - a) Agregar en el documento de esta asignación una sección de **validación de diseño** donde en una tabla, por cada ítem de diseño, se asocie al menos un requerimiento del SyRS.
3. Todo el código producido debe seguir el estándar fijado por el equipo de desarrollo al inicio.
 - a) Se debe utilizar al menos una herramienta para verificar el estándar de codificación y/o documentación.
 - 1) Incluya en el presente documento, una **sección que detalle el modo de uso de la herramienta de verificación de la codificación y/o documentación**.
 - b) (10 Puntos extra) Utilizar una herramienta como Jenkins junto con git para condicionar la modificación del repositorio a la verificación de la codificación y documentación
 - 1) Incluya en el presente documento, una sección que detalle el modo de uso de la herramienta Jenkins.
4. El repositorio debe contener todos los items de configuración del software, y el manual debe elicitar el procedimiento para obtener el mismo.
 - a) Incluya un **procedimiento corto de como obtener la versión actual del sistema**, junto con pantallazos que lo demuestren (puede basarse en la tarea anterior).
5. Se deben cuantificar al menos dos métricas que correspondan a los atributos (requerimientos funcionales y no funcionales) del programa.
 - a) Al menos una de las métrica debe utilizar una herramienta que lo mida automáticamente: Incluya la documentación sobre como usar la herramienta para cuantificar la métrica.
6. Deben existir al menos cuatro pruebas unitarias, todas deben satisfacerse al finalizar el sprint.
 - a) Se debe utilizar la herramienta *junit* para construir y validar en cada construcción del proyecto las unidades.
 - b) Incluya en el presente documento una sección de **especificación de pruebas unitarias** que defina las entradas (esperadas y atípicas) y salidas (esperadas).