

Tarea 1

Jose Fernando Molina Yordan Jimenez
Leonardo Mendoza Lisa Weber

October 1, 2016

1 Características, atributos y pesos

External and Internal Quality		
Característica	Atributo	Peso(Alto, Medio, Bajo)
Funcionalidad	Precision	Alto
	Seguridad	Medio
	Maturity	Alto
Confiabilidad	Tolerancia a Fallos	Medio
	Recuperabilidad	Medio
	Interoperabilidad	Alto
Usabilidad	Entendibilidad	Medio
	Aprendibilidad	Alto
	Atractividad	Alto
Eficiencia	Comportamiento Temporal	Medio
	Utilizacion de Recursos	Medio
Mantenibilidad	Testabilidad	Bajo
	Cambiabilidad	Medio
	Analizabilidad	Bajo
Portabilidad	Coexistencia	Bajo
	Reemplazabilidad	Bajo
	Adaptabilidad	Medio

2 Motivacion de la escogencia de los atributos

- **Precision:** Es de suma importancia para un club tan importante como Boca Juniors que el producto implementado pueda analizar de manera precisa todos los datos que puedan ser considerados de importancia para el club. La precisión del sistema es uno de los pilares fundamentales para la aceptación o rechazo del sistema de análisis automático de vídeos de fútbol a implementar.
- **Seguridad:** Al club Boca Juniors le interesa tener una ventaja sobre sus rivales, por lo tanto la confidencialidad, la integridad y el acceso a los datos son de alta importancia para mantener al club por encima de sus rivales. Los datos del análisis no deben ser accesibles para quien no tiene derechos o intereses con el club.

- **Interoperabilidad:** El sistema de analisis automatico de videos de futbol debe ser integrado con los sistemas existentes del club, esto con la finalidad de que el club pueda utilizar su plataforma ya existente con la menor cantidad de cambios o adaptaciones posibles, haciendo que el sistema de análisis sea una herramienta mas de la estrategia de TI del club.
- **Tolerancia a Fallos:** El software construido debe estar pensado para soportar fallos en su ejecución causados por situación o entradas anómalas debido a la naturaleza de tiempo real que tiene el proyecto. Para el análisis de un video previamente grabado tal vez la tolerancia a los fallos puede ser más relajada, pero pensando en análisis de video en tiempo real es inaceptable que en medio partido el sistema deje de funcionar, pues se pierden valiosos segundos de datos que podrían ser utilizados para el análisis de datos.
- **Recuperabilidad:** Motivación similar al punto anterior. Además, la recuperabilidad del sistema es importante debido a que los partidos de fútbol tienen un tiempo finito, y en caso de falla es vital que el sistema pueda recuperarse rápidamente y con la menor cantidad de interacción por parte del operario .
- **Maturity:** El sistema debe comportarse de manera confiable y debe proveer datos confiables que ayuden al equipo Boca Juniors a lograr sus objetivos durante la temporada. Fallos en el software deben ser mínimos y las fallas en el sistema implementado no deben interferir con el correcto funcionamiento del departamento de informática ya establecido de la compañía.
- **Entendibilidad:** Los usuarios que utilizaran el sistema dia a dia no son especialistas en el área de las TI, son personeros del club como la junta directiva o el cuerpo tecnico. Es por este motivo que se espera que el software sea simple y sencillo de entender su funcionamiento, para de esta manera asegurar que el software cumple con las funcionalidades requeridas por los diferentes stakeholders del club Boca Juniors y cumplir las expectativas que se tienen.
- **Aprendibilidad:** El sistema realizado debe ser amigable para el usuario y debe proveer facilidades para el aprendizaje del mismo. Como fue mencionado, los usuarios del sistema serán personeros del cuerpo tecnico y junta directiva, los cuales pueden no estar familiarizados con la utilización de los sistemas de información, por lo que se espera facilidad de uso del sistema y una curva de aprendizaje baja.
- **Atractividad:** Se espera una interfaz gráfica atractiva, que inste al usuario a utilizar la herramienta y que al mismo tiempo facilite el entendimiento y las funcionalidades. Para el usuario que no es del área de TI la interfaz gráfica tiene algunas veces mayor importancia que lo lógica de implementación, porque para ellos el sistema es una caja negra con la cual interactúan por medio de esta interfaz, lo que significa que es todo lo que ven del sistema, y lo juzgan de acuerdo a esta apariencia junto con su funcionalidad. La manera de mostrar los datos puede ayudar a la interpretación de los mismos y la toma de decisiones basados en ellos.

- **Comportamiento temporal:** Observar la eficiencia con la que el software analiza los pixeles, esto con el fin de mostrar resultados claros acerca del procesamiento estos al equipo de desarrollo del Boca Juniors.
- **Utilización de recursos:** La directiva del club le interesaría saber la cantidad de memoria que necesita el dispositivo que ejecutaría el análisis del video para de esta manera tener datos claros sobre la posibilidad de comprar un sistema dedicado o alquilar servicios en la nube.
- **Testabilidad:** Medir la calidad de las pruebas preparadas por nuestro equipo para la aplicación, cuando son ejecutadas por un usuario externo a nuestro equipo, principalmente si es un miembro del equipo de desarrollo del Boca Junior es importante para que en futuras versiones del producto se puedan tener pruebas base las cuales realizar para verificar el sistema.
- **Cambiabilidad:** Establecer un porcentaje que demuestre lo editable que son las clases o si brindan una complejidad de cambio a la hora de realizar mantenimiento por parte de un usuario externo, como por ejemplo un miembro del equipo de desarrollo del Boca Juniors. El equipo de desarrollo debe asegurar que futuros cambios en el software sean implementables por agentes externos que pudieron no haber sido parte del desarrollo inicial del producto.
- **Analizabilidad:** Medir lo comprensible que puede ser el software construido por nuestro equipo a la hora de entrar a la etapa de mantenibilidad, esto desde un punto de visto externo, para que futuros mantenimientos sucedan de la mejor manera posible.
- **Coexistencia:** A los usuarios futuros de la entidad deportiva del Boca Juniors es importante informarles si alguna inconsistencia puede suceder mientras ejecuta el sistema conjunto a otros programas (páginas web), esto con el fin de asegurar una afectación mínima de sistemas externos en caso de falla del sistema en cuestión.
- **Reemplazabilidad:** Indicar a los directivos del Boca Juniors un índice de los reemplazos que genera el ingreso de este sistema, basándose en las funcionalidades aplicadas en nuestro sistema comparándose con las aplicaciones que se usan actualmente en la entidad.
- **Adaptabilidad:** Mostrar a las distintas partes interesadas de la entidad deportiva el Boca Juniors la capacidad del software a ejecutarse y adaptar a distintos navegadores que se utilizan o podrían a un futuro cercano dentro de los equipos informáticos.

3 Tabla de metricas

Quality in use measurement category				
Característica	Atributo	Metrica	Nivel Requerido	Herramienta
Funcionalidad	Precision	Deteccion de Jugadores	$\sum_{n=1}^{100} \frac{A_n}{B_n}$ donde A_n = Jugadores detectados en el cuadro N B_n = Cantidad de jugadores en el cuadro N	JUnit
	Seguridad	Auditabilidad de acceso	$\frac{A}{100}$ donde A=Cantidad de veces que el login fue exitoso	Fortify
	Interoperabilidad	Integracion de los Modulos	$\frac{A}{B}$ donde A = Pruebas de integracion exitosas B = Cantidad de pruebas de integracion realizadas	JUnit
Confiabilidad	Tolerancia a Fallos	Grado de manejo de excepciones	$\frac{A}{B}$ donde A = Cantidad de excepciones manejadas B = Cantidad de excepciones que pueden ocurrir	Java API & JUnit
	Recuperabilidad	Disponibilidad	$\frac{T_o}{T_o + T_r}$ donde T_o = Tiempo operativo T_r = Tiempo en reparaciones	Java API & JUnit
	Maturity	Deteccion de fallas	$\frac{A}{B}$ donde A = Cantidad de pruebas exitosas B = Cantidad de pruebas realizadas	Java API & JUnit
Usabilidad	Entendibilidad	Compleitud de descripcion	$\frac{A}{B}$ donde A = Cantidad de requerimientos funcionales explicados B = Total de requerimientos funcionales	Reqview
	Aprendibilidad	Facilidad de aprendizaje	$\frac{A}{B}$ donde A = Tiempo total tomado B = Cantidad de actividades	Microsoft Excel
	Atractividad	Atractividad de la interfaz grafica	$\frac{A}{B}$ donde A = Suma de las clasificaciones de cada rúbrica B = Puntaje máximo	SurveyMonkey

Quality in use measurement category				
Característica	Atributo	Metrica	Nivel Requerido	Herramienta
Eficiencia	Comportamiento temporal	Pixeles analizados por minuto	$\frac{\sum_{n=1}^{100} \frac{H*W*F}{T_n}}{100}$ donde H = Alto del video en pixeles W= Ancho del video en pixeles F = Frames por segundo T _n =Tiempo en segundos que tardo el analisis del frame N	JUnit
	Utilizacion de recursos	Uso de memoria	$\frac{\sum_{n=1}^{100} A_n}{100}$ donde A _n = Cantidad maxima de memoria utilizada en la ejecucion N	JProfiler
Mantenibilidad	Testabilidad	Reinicio en Pruebas	$\frac{A}{B}$ donde A = Numero de pruebas cuando se reinicio B = Cantidad total de pruebas	Microsoft Excel
	Cambiabilidad	Complejidad de Mantenimiento	$\frac{A}{B}$ donde A = Cambios que presentaron dificultad de cambio B = Cantidad total de cambios	Metrics3
	Analizabilidad	Capacidad de diagnostico de errores	$\frac{A}{B}$ donde A = Errores dificiles de detectar B = Cantidad total de errores	Microsoft Excel

Quality in use measurement category				
Característica	Atributo	Metrica	Nivel Requerido	Herramienta
Portabilidad	Coexistencia	Tiempo en suceder colapsos	$\frac{T_a}{T_o}$ donde T_a = Tiempo con comportamiento anormal T_o = Tiempo operativo	Java API & JUnit
	Reemplazabilidad	Funciones reemplazadas	$\frac{A}{B}$ donde A = Numero de metodos reemplazadas B = Cantidad total de metodos en el modulo	Selenium
	Adaptabilidad	Adaptacion del sistema a navegadores	$\frac{A}{B}$ donde A = Numero de navegadores sopor-tados B = Cantidad to-tal de navegadores probados	Ghostlab

4 Explicacion de las metricas

Metrics explanation			
Metrica	Proposito	Metodo de aplicacion	Medicion
Deteccion de Jugadores	Medir la precisión del algoritmo al momento de reconocer los jugadores de los equipos que se encuentran en el terreno de juego.	Tomar 100 frames válidas (toma televisiva del campo de juego) del video por analizar y comparar la cantidad de jugadores detectados contra un archivo de ground-truth con la cantidad verdadera de jugadores que se encuentran en el frame.	≥ 0.8
Auditabilidad de acceso.	Medir la facilidad con la que un auditor o alguna persona interesada pueda obtener la información acerca de los logins y de los usuarios que utilizaron el sistema.	Realizar login 50 veces bajo diferentes usuarios. Probar con cada usuario un acceso con la contraseña correcta y otro con la contraseña incorrecta. Por cada intento de acceso, el sistema debe escribir en la bitácora el nombre de usuario que intentó hacer login y si este fue exitoso o no.	≥ 0.95
Integración de los módulos.	Medir la capacidad que tiene cada módulo del sistema de interactuar entre sí y aprobar las pruebas de integración definidas.	Correr todas las pruebas de integración definidas en el plan de pruebas. Dividir la cantidad de pruebas que fueron exitosas entre el total de pruebas realizadas.	≥ 0.8
Grado de manejo de excepciones.	Medir el grado al que se manejaron las diferentes excepciones que pueden generarse durante la ejecución del programa.	Por cada invocación de un método, revisar todas las posibles excepciones que pueden generarse de la llamada y asegurarse que todas hayan sido manejadas por un bloque try-catch.	≥ 0.75
Disponibilidad	Medir el tiempo que el sistema es capaz de mantenerse funcionando sin fallos y el tiempo que toma realizar una reparación en caso de fallo.	Probar el producto bajo un funcionamiento normal. Monitorear el tiempo que trabaja sin problemas y, en caso de fallo, el tiempo que se toma en volverlo a poner a funcionar.	≥ 0.9
Deteccion de fallas	Enumerar el número de fallas detectadas en la revisión del producto y compararla con el número de pruebas que fueron utilizadas para medir dichas fallas.	Realizar todas las pruebas unitarias y de integración definidas en el plan de pruebas. Tomar la cantidad de pruebas que fueron exitosas y dividir las entre todas las pruebas realizadas.	≥ 0.9

Metrics explanation			
Metrica	Proposito	Metodo de aplicacion	Medicion
Compleitud de la descripcion	Medir que tan completa se encuentra la descripcion de la usabilidad y de las funciones del sistema.	Comparar la lista de requerimientos funcionales contra los manuales de usuario, las paginas de ayuda y tutoriales realizados. Cuántas de las funcionalidades vienen descritas en alguno de estos artefactos? Tomar la cantidad de funcionalidades que vienen descritas en los artefactos y dividirlos entre la cantidad de requerimientos.	≥ 0.95
Facilidad de aprendizaje.	Medir la facilidad que tienen los usuarios para encontrar y utilizar las funcionalidades del sistema.	Tomar un usuario que nunca haya utilizado el sistema. Pedirle que realice una serie de actividades funcionales con el software y medir el tiempo que le toma hacerlo. Luego, dividir entre la cantidad de funciones para obtener el tiempo promedio que le toma.	$\leq 5\text{min}$
Atractividad de la interfaz gráfica	Medir desde la perspectiva del usuario que tan atractiva es la interfaz gráfica del sistema desarrollado.	Mediante un cuestionario con rúbricas sobre la interfaz gráfica y una calificación entre 1 y 10, encuestar a 10 posibles usuarios del sistema acerca de su percepción de la interfaz gráfica.	≥ 0.8
Píxeles analizados por segundo	Cuantificar los píxeles que pueden ser analizados por minuto en el sistema, esto en un sistema operativo dado y con recursos preestablecidos.	Ejecutar 50 pruebas de análisis de videos donde se cuantifique término de tiempos y píxeles cuántos pueden ser ejecutados por minuto.	≥ 24000000
Uso de memoria	Obtener la cantidad de memoria necesaria para ejecutar el análisis del video.	Ejecutar 100 veces el análisis de videos, de los cuales se almacenará el número de bytes mayor utilizado en cada prueba. Luego se obtiene un promedio.	$\leq 80\%$ de la memoria disponible

Metrics explanation			
Metrica	Proposito	Metodo de aplicacion	Medicion
Reinicio en pruebas	Captar cuando el usuario realiza una prueba y tuvo que reiniciar esta para observar detalladamente lo que sucedía en la aplicación.	Mientras se realizan pruebas externas observar el comportamiento del ejecutante cuando debe reiniciar una prueba, e ir a revisar el código fuente para observar detalladamente lo que sucede.	≥ 0.85
Complejidad de mantenimiento	Analizar si en las clases del sistema de análisis de video, presentan algún tipo de complejidad a la hora de realizar cambios, esto con el fin de analizar dichas clases para mejor la cambiabilidad.	Analizar el comportamiento del usuario externo que se encuentra realizando cambios en alguna clase, y observar cuando presenta mayor dificultad a la hora de enfrentar algún cambio.	≤ 0.2
Capacidad de diagnóstico de errores	Mostrar un índice a los futuros usuarios encargados de mantenimiento, de lo fácil de diagnosticar errores en nuestro código fuente.	Mientras el software se encuentra en etapa de pruebas, se observa la complejidad que presenta el usuario al analizar el origen del problema o error detectado.	≤ 0.2
Tiempo en suceder colapsos	Obtener un tiempo promedio de ocurrencias de errores en el sistema de análisis de videos, cuando se ejecuta con otros sistemas (páginas web).	Ejecutar pruebas del sistema ejecutándose concurrentemente con otros sistemas, y medir el tiempo que falla la pagina o hay pérdida significativa de eficiencia en nuestro sistema en un tiempo total.	≤ 0.25
Funciones reemplazadas	Medir los reemplazos generados por nuestro sistema que afecta directamente en el sector destinado.	Contar las funciones que se sustituyen en la entidad comparados con el total de funcionalidades que existen antes en el equipo, esto por medio de encuestas a los futuros usuarios.	≤ 0.9
Adaptación del sistema a navegadores web	Analizar el comportamiento y resultados que brinda el sistema de análisis de videos en distintos navegadores, comparando dichos resultados esperados en cada funcionalidad.	Verificar la ejecución de las funcionalidades en cada navegador y ver si son los resultados esperados o si no se cumplió la expectativa.	≥ 0.85