

Instituto Tecnológico de Costa Rica

Ingeniería en Computación

**IC6831-Aseguramiento de la Calidad
Tarea 1: Atributos de Calidad y Métricas
Profesor: Saúl Calderón Ramírez**

**Estudiantes:
Luis Alonzo Cascante Franco 2014159896
Olman Castillo Picado 2015148651
María Laura Pizarro Moreno 2015120626
Gabriel Venegas Castro 2013115967**

Segundo Semestre 2017

1. Atributos de calidad

Calidad externa e interna		
Categoría	Atributos	Peso (Alto/Medio/Bajo)
Funcionalidad	1.Seguridad de acceso	A
	2.Exactitud	A
	3.Interoperabilidad	B

1. Es importante que solo las personas con los permisos necesarios puedan modificar los datos del sistema. Cualquier otro usuario no debería tener acceso a la información confidencial.
2. El sistema debe de ser lo más preciso posible para no confundir a una persona por otra.
3. Por el momento el sistema no necesita interactuar con otros sistemas.

Calidad externa e interna		
Categoría	Atributos	Peso (Alto/Medio/Bajo)
Fiabilidad	1.Madurez	A
	2.Tolerancia a Fallos	A
	3.Capacidad de recuperacion	A

1. Es necesario que el programa sea lo más exacto posible para el reconocimiento facial.
2. Es indispensable que el programa sepa responder a algún fallo ya sea si la base de datos tuvo algún fallo o la cámara no envió bien la foto. El programa debería tener prevista alguna acción para fallos.
3. Si el sistema llega a fallar es necesario que el tiempo de reinicio sea mínimo.

Calidad externa e interna		
Categoría	Atributos	Peso (Alto/Medio/Bajo)
Eficiencia	1. Comportamiento Temporal	A
	2. Utilización de recursos	A

1. Es necesario que el reconocimiento facial lo haga lo más rápido posible para agilizar los trámites.
2. Los cálculos que realiza para el reconocimiento facial deben de ser lo más eficientes ya que la idea es agilizar los trámites, por lo que no debería usar memoria o procesador de más si existe la posibilidad de hacer los algoritmos más eficientes.

Calidad externa e interna		
Categoría	Atributos	Peso (Alto/Medio/Bajo)
Usabilidad	1. Capacidad para ser entendido	A
	2. Capacidad para ser aprendido	B
	3. Capacidad para ser operado	A

1. Es de prioridad alta ya que si el usuario no entiende cómo funciona el programa entonces el tiempo de uso va a ser mayor, por lo que el propósito de que se agilicen los trámites no va a ser tan satisfactorio.
2. No es necesario que el usuario se aprenda como funciona el sistema ya que este no va a ser de uso cotidiano, y si se logra entender con facilidad no importa si no llega a ser aprendido.
3. Es necesario que la interacción que el usuario tenga durante el proceso sea mínima, ya que se desea agilizar el proceso de trámite.

Calidad externa e interna		
Categoría	Atributos	Peso (Alto/Medio/Bajo)
Portabilidad	1. Adaptabilidad	M
	2. Instalabilidad	B
	3. Capacidad para ser reemplazado	B

1. En un principio el sistema cumplirá con su propósito principal, sin embargo tendrá una moderada capacidad de aceptar cambios en pro de permitir adaptaciones a nuevas funcionalidades si así se requiere.
2. Puesto que la aplicación se maneja con una interfaz web, no es necesario que el cliente se instale software de ningún tipo, pues puede acceder fácilmente al sistema de reconocimiento por medio de su navegador preferido.
3. Al ser un sistema cuya interfaz es web, puede ser fácilmente reemplazado por otro sistema de cualidades similares sin necesidad de realizar una excesiva cantidad de cambios y adaptaciones al nuevo sistema.

Calidad externa e interna		
Categoría	Atributos	Peso (Alto/Medio/Bajo)
Mantenibilidad	1. Estabilidad	M
	2. Capacidad para ser cambiado	M
	3. Capacidad para ser analizado	M

1. Si el sistema es actualizado, este debe de poder estar adaptado para que un cambio no haga colapsar al sistema. Por el momento un cambio no es algo indispensable por lo que no sería una prioridad alta

2. El sistema podría tener mejoras a futuro por lo que debe tener la capacidad de ser cambiado. No es algo primordial pero si es algo que pueda suceder en algún momento.
3. Si el sistema llega a manos de terceros y necesitan hacer un arreglo o una actualización, el código debe de estar lo suficientemente acomodado para que se logre entender. Por el momento no se sabe si algún tercero va a analizar el sistema, por lo que no es prioridad alta.

2. Identificación de las métricas

Categoría de medición				
Característica	Metricas	Nivel Requerido	Herramienta	Resultado
1. Efectividad	Tiempo de respuesta	Alto	JMeter	
2. Productividad	Tiempo en realizar una tarea	Alto	Screencastify	
3. Seguridad	Grado de seguridad	Alto	IEEE STD-1228	
4. Satisfacción	Nivel de adaptación	Alto	Google-forms	

2.1. Efectividad:

El tiempo de respuesta es considerado un punto de suma importancia, ya que el sistema deberá realizar procesos que pueden requerir mucho tiempo. esto puede ser percibido de mala manera por el usuario, por esto se establece el tiempo de respuesta como una métrica de importancia. Para evaluar esta métrica se planteó el uso de JMeter que me permite ver lo que dura una respuesta de parte del servidor.

2.2. Productividad:

El tiempo en que un usuario dure en realizar una tarea es de suma importancia en este sistema teniendo en consideración el entorno en el cual será ejecutado la facilidad de uso y la rapidez en la cual se puedan realizar las operaciones es de suma importancia. Para lograr analizar la interacción del usuario con la aplicación se decidió usar la herramienta Screencastify - Screen Video Recorder que es una herramienta que se puede instalar en el navegador chrome y con la cual se puede grabar las interacciones del usuario. Con esta información se puede analizar el tiempo que dura el usuario en realizar la tarea y como es la navegabilidad del sistema.

2.3. Seguridad:

Para obtener un grado de seguridad óptimo se planea seguir los lineamientos establecidos en el estándar IEEE STD-1228, con el objetivo de asegurar el cumplimiento de este atributo .

2.4. Satisfacción:

Se realizará una encuesta con respecto a la aceptación que tienen los usuarios finales con respecto al producto. Para realizar esta encuesta se utilizará la plataforma para realizar formularios google-forms.

3. Plan de evaluación de métricas

3.1. Funcionalidad

1. Seguridad de acceso: para las pruebas de seguridad de acceso se utilizará Selenium, pues esta herramienta nos permitirá programar intentos de inicio de sesión con usuarios válidos e inválidos, estudiando así el comportamiento el programa en todos los casos posibles.
2. Exactitud: con un conjunto de muestras controladas, se le realizarán pruebas al algoritmo de comparación, sabiendo de antemano quién es el individuo, de tal forma, se obtendrá un promedio de aciertos, el cuál será la métrica como tal.
3. Interoperabilidad: al no tener que operar con otros sistemas, la única prueba realizable al sistema sería con Sonarqube, para determinar el nivel de dependencia que exista entre las capas o diferentes módulos del sistema. De esta forma sabremos, si se tiene un índice bajo de dependencias será mucho más fácil acoplarlo a otros sistemas.

3.2. Fiabilidad

1. Madurez: se utilizará Orcanos software para medir este atributo.
2. Tolerancia a fallos: Para la tolerancia a fallos no hay ninguna herramienta específica a utilizar. Se ejecutarán pruebas unitarias para las mediciones.
3. Capacidad de recuperación: se simularán caídas completas del problema y se tomará tiempo de los tiempos de recuperación.

3.3. Eficiencia

1. Comportamiento temporal: Se medirá el tiempo de ejecución de los procedimientos utilizando la librería timeit de Python.
2. Utilización de recursos: El uso de recursos se medirá mediante la librería “resource” de Python, y se analizará los datos obtenidos de forma que coincidan con el uso de recursos esperado para la realización de los cálculos.

3.4. Usabilidad

1. Capacidad para ser aprendido: Mediante pruebas controladas con nuevos usuarios, se tomarán muestras de cuántas acciones les toma a cada uno realizar una función específica después de haberles indicado cómo se realiza esta función, ya sea mediante un ejemplo o una guía hablada.
2. Capacidad para ser entendido: Se realizará pruebas controladas con usuarios que nunca hayan usado el software para medir qué tanto entienden de la función del software una vez se les explica dicha función.
3. Capacidad para ser operado: Por medio de pruebas con usuarios que conocen el funcionamiento del software, se medirá el tiempo que le toma a cada usuario realizar funciones específicas. Poca interacción conlleva a una mayor facilidad de operación.

3.5. Portabilidad

1. Adaptabilidad:
2. Instalabilidad: Para comprobar y garantizar la capacidad del sistema de ser ejecutado en distintos entornos, se probarán las funciones desde distintos sistemas operativos con distintos navegadores.
3. Capacidad para ser reemplazado. Mediante el uso de un sistema básico secundario, se mostrará cómo el sistema principal es fácilmente reemplazable al cambiar la interfaz web del principal por la del secundario.

3.6. Mantenibilidad

1. Estabilidad: para este atributo también analizaremos la dependencia entre módulos con Sonarqube, pues a menor dependencia entre módulos o capas, más fácil será hacer cambios sin alterar el comportamiento del resto del sistema.
2. Capacidad para ser cambiado: la misma herramienta Sonarqube con el mismo propósito.
3. Capacidad para ser analizado: la misma herramienta Sonarqube pero con el propósito y funcionalidad de detectar olores de código y código limpio.