

Aseguramiento de la Calidad del Software: Prueba de concepto, proyecto semestral

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

3 de agosto de 2017

A continuación se enlistan los detalles de la prueba de concepto para el proyecto semestral .

Fecha de entrega: 20 de Agosto.

Valor: 5 % del proyecto final.

A través del TEC digital.

1. Prueba de concepto o POC (Proof of Concept)

Una prueba de concepto es una implementación mínima de una idea o concepto. Esta etapa es usual entre las organizaciones desarrolladoras de software, para probar la factibilidad de una idea; una o varias tecnologías, uno o varios algoritmos etc; para resolver una necesidad o problema específico.

Para el proyecto semestral del curso se plantea realizar una prueba de concepto, donde se debe implementar:

1. **Metodología scrum:** Mediante el uso de la herramienta libre *zoho projects*: <https://projects.zoho.com/> gestionar el desarrollo de un sprint para entregar el PoC la fecha indicada. Usar la documentación disponible en la carpeta del curso para conocer los eventos, productos y roles de la metodología scrum.
2. **Uso de una herramienta de control de versiones:** Es necesario usar una herramienta de control de versiones (subversion o git) integrada al entorno de desarrollo (en *Eclipse* o también *tortoise*). Al enlace del tec digital se debe especificar el *url* del proyecto en un repositorio *github* o *gitlab*.
3. **Un diagrama de componentes o clases de alto nivel:** Un diagrama de componentes en una herramienta *staruml* o similar donde se especifiquen componentes o clases en general. En esta etapa no es necesario el uso de

patrones de diseño. Lo único necesario es definir los componentes (en el *frontend* y *backend*) con los módulos agrupando algoritmos y funcionalidades.

4. **Implementación de al menos cuatro trozos del sistema:** Se debe probar el uso de al menos cuatro algoritmos basados en *OpenCV* u otra librería como *numpy* o *scipy* en Python (manipulación de imágenes, cálculos de matrices de covarianza, auto-vectores, etc.). Además es necesario incluir implementaciones mínimas de la interfaz web usando *AngularJs* o alguna otra tecnología para el *frontend* de su preferencia.
 - a) El *front-end* debe interactuar con el *back-end*.
 - b) Incluir además pruebas unitarias usando *unittest* <https://pythonbc.com/blog/pruebas-unitarias-unit-test-con-python/> en el entorno *eclipse* de los algoritmos implementados.
5. **Documento del PoC:** Debe incluir el diagrama de componentes de alto nivel, el análisis básico de los *trozos* implementados, además de detalles de problemas encontrados con las herramientas, y su resolución.
6. **Documentación interna HTML o LaTeX:** Debe adjuntar el html (o el archivo LaTeX) generado con la herramienta doxygen o similar, a partir del uso del estándar *javadocs* en la documentación interna.