

Geolocalización de Sitios de Interés Para Aplicaciones Móviles

G-SIAM



Plan de Aseguramiento de Calidad del Software SQAP

VERSIÓN

1.1



Contenido

1	INTRODUCCIÓN.....	3
2	SECCIONES.....	3
2.1	SECCIÓN 1 – PROPÓSITO	3
2.2	SECCIÓN 2 – DOCUMENTOS DE REFERENCIA	3
2.3	SECCIÓN 3 – GESTIÓN	3
2.4	SECCIÓN 4 – DOCUMENTACIÓN.....	3
2.5	SECCIÓN 5 – ESTÁNDARES, PRÁCTICAS Y CONVENCIONES	4
2.5.1	ESTÁNDAR DE DOCUMENTACIÓN	4
2.5.2	ESTÁNDAR DE CÓDIGO	6
2.6	SECCIÓN 6 – REVISIONES	7
2.7	SECCIÓN 7 – GESTIÓN DE CONFIGURACIÓN DE SOFTWARE	7
2.8	SECCIÓN 8 – INFORME DE PROBLEMAS Y ACCIÓN CORRECTIVA	7
2.9	SECCIÓN 9 – HERRAMIENTAS.....	8
2.9.1	ISSUE TRACKING	8
2.9.2	METRICS.....	8
2.9.3	WIKI.....	8
2.10	SECCIÓN 10 – CONTROL DE CÓDIGO	8
2.11	SECCIÓN 11 – CONTROL DE MEDIOS.....	9
2.12	SECCIÓN 12 – CONTROL DEL PROVEEDOR.....	9
2.13	SECCIÓN 13 – REGISTRO, MANTENIMIENTO Y RETENCIÓN	9
3	MÉTRICAS.....	9
3.1	MÉTRICAS DE CÓDIGO.....	9
3.2	MÉTRICAS DE ESFUERZO	10



1 Introducción

El objetivo del Plan de de aseguramiento de la calidad es establecer por escrito los objetivos, métodos y responsabilidades asociados a la calidad del software.

El plan de aseguramiento de la calidad que utiliza el equipo de proyecto, se basa en el resumen de la norma IEEE Std-1986, visto en la asignatura “*Ingeniería de Software II*” en el tercer año de la carrea de “Licenciatura en Informática”.

2 Secciones

2.1 Sección 1 – Propósito

En el presente Plan de SQA se describe la estrategia para el aseguramiento de la calidad durante el proyecto. El objetivo de este plan es garantizar un estándar de calidad mínimo, tanto proceso de construcción del software “G-SIAM”, como en el producto final, entendiéndose por producto el software más su documentación.

2.2 Sección 2 – Documentos de referencia

Norma IEEE Std-1986.

2.3 Sección 3 – Gestión

Existe un rol de encargado de las actividades de SQA, llamado “Encargado de SQA” que no participa en tareas de desarrollo. El Encargado de SQA es responsable de generar la documentación correspondiente así como de redactar el Plan, y hacer directamente las tareas de control y seguimiento.

Ya que el equipo de trabajo es de sólo 2 personas y no se dispone de un recurso externo que pueda realizar estas tareas, las mismas serán realizadas por alguna de las personas del equipo.

2.4 Sección 4 – Documentación

La documentación del proyecto consiste en un conjunto principal de documentos denominados carpeta principal, y otro grupo de documentos complementarios denominados anexos. En la carpeta principal se encuentra el plan de proyecto, el documento de análisis de riesgo, el plan



de pruebas, el presente plan de SQA, el plan de SCM, el documento de especificación de requerimientos, la documentación de arquitectura, documento de estado del arte entre otros documentos de alta relevancia. En los anexos se incluyen casos de prueba, manual de instalación, manual de usuario, documento de investigación de mercado y demás información complementaria.

2.5 Sección 5 – Estándares, Prácticas y Convenciones

Los procedimientos de gestión de configuraciones se pueden observar en detalle en el Plan de SCM (Ver Documento “Plan de Gestión de Configuración de Software”). Las prácticas y procedimientos del aseguramiento de calidad están identificadas en el propio Plan de Calidad. Mediante revisiones se controla que todas las prácticas y los estándares sean cumplidos.

2.5.1 Estándar de Documentación

A continuación se describe el estándar definido para este proyecto.

- Tipo de hoja A4.
- Tipo de letra Arial.
- Tamaño de letra 11.
- Párrafos justificados.
- Los párrafos son separados por un renglón en blanco.
- Títulos 4 niveles.
- Letra de los títulos Arial.
 - Título 1 tamaño 20 negrita.
 - Título 2 tamaño 18 negrita y cursiva.
 - Título 3 tamaño 16 negrita y cursiva.
 - Título 4 tamaño 14 negrita y cursiva.
 - Título 5 tamaño 12 negrita y cursiva.
- Para la carpeta se utilizaran capítulos (Arial tamaño 22), los mismos deberán comenzar en una página exclusiva.
- Si un título queda en la última línea de la hoja, el título deberá pasar a la siguiente hoja.
- Las imágenes irán centradas y con un borde negro de ½ punto.

```
SELECT * FROM DUAL
```


- Para referirse a segmentos de código se utilizara el siguiente contenedor:
- Todos los documentos contendrán una carátula como página principal, la misma se confecciona en 3 partes:



Cabezal

 UNIVERSIDAD DE LA EMPRESA	Proyecto de Grado – Lic. En Informática
--	---

Principal

Geolocalización de Sitios de Interés Para Aplicaciones Móviles	
G-SIAM	
	
Nombre del Documento	
VERSIÓN	1.0

Pie

Universidad de la Empresa
Soriano 959 - Tel.: 2900 2442
Montevideo - Uruguay | info@ude.edu.uy



- Todas las paginas menos la carátula tendrán un cabezal y un pie de página:

Cabezal

	Geolocalización de Sitios de Interés Para Aplicaciones Móviles	G-SIAM	
	Nombre del Documento	Versión: 1.0	

Pie

	2 de 3
--	--------

- Los documentos al referirse al equipo de trabajo utilizan únicamente las siguientes opciones
 - El equipo
 - El equipo de proyecto
 - Omitiendo el sujeto hablando en tercera persona del singular
- Se adjuntan algunos ejemplos que indican el modo en que se expresan los documentos
 - "El equipo decide desarrollar un modelo incremental iterativo"
 - "Se decide desarrollar un modelo incremental iterativo"
- Como buena práctica se creó un template de documento para facilitar la creación de nuevos documentos. (Ver Documento "Template Documento.doc").

2.5.2 Estándar de Código

En cuanto a la codificación, se utiliza el estándar oficial de codificación del lenguaje Java, con la excepción de las salvedades que se listan más adelante. Este garantiza una mejor comprensión y mayor facilidad de revisión de programas, en un ambiente que involucra a más de un programador. Esto garantiza la calidad del código generado por cada uno de los miembros del equipo de trabajo.

Por más información del estándar, referirse a <http://java.sun.com/docs/codeconv/>

Los siguientes elementos puntuales se admiten y priman sobre la definición del estándar antes mencionado.

- Se admiten tabuladores de 4 espacios
- Se admiten el uso del comentario de línea (//) para varias líneas sucesivas, a modo de comentario de bloque.
- Para la estructura de control "If else - if" además del formato mencionado en el estándar se admite el siguiente:



```
if (condición) {  
    //sentencias  
}  
else  
    if (condición){  
        //sentencias  
    }
```

- Los nombres de los métodos no sólo pueden ser verbos en infinitivo, sino que también se pueden usar otras palabras del idioma español. Se persigue el objetivo de que se entienda claramente el sentido del método y se respete la regla de mayúsculas y minúsculas descrita en el estándar.

2.6 Sección 6 – Revisiones

Se define que se realizarán Revisiones Técnico Formales (RTF) al final de cada iteración. Ambos integrantes deberán controlar que el código fuente y los documentos generados se ajusten a los estándares definidos en los documentos de referencia.

2.7 Sección 7 – Gestión de Configuración de Software

Se dispone de un plan de SCM que detalla las actividades de ésta área. (Ver Documento “Plan de Gestión de Configuración de Software”).

2.8 Sección 8 – Informe de problemas y acción correctiva

Los problemas detectados se informan a través del Issue Tracking que existe en el sitio del proyecto (<http://code.google.com/p/geolocalizacion-uas/issues/list>). Esta tarea le corresponde al encargado de calidad. Adicionalmente en reuniones a las que convoca el encargado de calidad cuando cree conveniente, se realiza una devolución de los problemas más generales encontrados con el objetivo de que otros miembros del equipo no cometan los mismos errores. Dado el tamaño del equipo completo de proyecto no se cree conveniente prefijar las fechas de estas reuniones.



2.9 Sección 9 – Herramientas

2.9.1 Issue Tracking

<http://code.google.com/p/geolocalizacion-uas/issues/list>

Esta herramienta dispone de 3 tipos de templates:

- Reporte de defecto del usuario
- Reporte de defecto del desarrollador
- Solicitud de revisión

Además cuenta con varios estados para los problemas reportados y son configurables, es decir se pueden agregar más según necesidad.

2.9.2 Metrics

<http://metrics.sourceforge.net/> (actualizada al 02/12/2011)

Para la generación de métricas de código se utiliza el plugin de eclipse 'metrics', el cual analiza el código y presenta diferentes métricas. Esta herramienta permite obtener parámetros útiles para realizar un seguimiento del desarrollo. La definición completa de las métricas ofrecidas por la herramienta se puede consultar en el sitio.

2.9.3 Wiki

<http://code.google.com/p/geolocalizacion-uas/wiki/Principal>

Para tener la información del proyecto on line se utiliza la wiki que viene incluida dentro del sitio del proyecto (Google Code). En ella se incluyen:

- Herramientas
- Tutoriales
- Consejos para desarrollar en Android
- Estándares
- Glosario

2.10 Sección 10 – Control de Código

Todo lo que es versionado del código se gestiona como se explica en el Plan de SCM (Ver Documento "Plan de Gestión de Configuración de Software").



2.11 Sección 11 – Control de Medios

En el plan de SCM se detalla el control que se tiene sobre los medios (Ver Documento “Plan de Gestión de Configuración de Software”).

2.12 Sección 12 – Control del Proveedor

No aplica a este proyecto ya que no hay proveedores.

2.13 Sección 13 – Registro, Mantenimiento y Retención

El equipo ha decidido almacenar toda la documentación del proyecto sin destruir nada en ningún momento. El objeto de esto es tener acceso a todos los datos históricos en el momento que se desee. Para ello se cuenta con herramientas de versionado apropiadas, y el espacio de almacenamiento no supone un problema.

3 Métricas

Se decide obtener métricas en base al código generado y en base al esfuerzo requerido por cada tarea. Para las métricas de código las mediciones se realizarán luego de cada cierre de cada iteración. Mientras que para las de esfuerzo se realizarán al final de cada etapa y cada iteración.

3.1 Métricas de Código

- Líneas de Código (TLOC y MLOC)
- Promedio de Métodos por Clase (NOM / NOC)
- Complejidad Ciclomática McCabe (CC)
- Profundidad del Árbol de Herencia (DIT)

A través de las 2 primeras métricas (Líneas de Código y Promedio de Métodos por Clase) se pretende tener cuantificado el tamaño del desarrollo. Mientras que las últimas 2 (Complejidad Ciclomática McCabe y Profundidad del Árbol de Herencia) intentan medir la calidad de la lógica del programa generado, y el riesgo que insume su desarrollo y mantenimiento.

Es importante realizar este seguimiento para poder detectar las áreas que insumen mayor esfuerzo de desarrollo y por consiguiente mayores pruebas.



3.2 Métricas de Esfuerzo

- Esfuerzo por Actividad
- Incidencia de la Actividad (horas de la actividad / horas totales de la iteración)
- Esfuerzo por CU

Con estas métricas se obtienen datos relevantes que indican cómo se divide el esfuerzo dentro del proyecto.

Estas métricas permiten comenzar a generar datos útiles a fin de poder lograr una estimación más certera de futuros proyectos.

De este modo se podrá asignar con más exactitud los recursos y mejorar sus costos.