

# Proyecto logO

## Informe de Integración

### Versión [1.2]

#### Historia de revisiones

Fecha	Versión	Descripción	Autor
24/11/2013	1.0	Creación del documento	Martín Prino
24/11/2013	1.1	Revisión SQA	Ezequiel Jardim
24/11/2013	1.2	Revisión SQA	Nicolás Ramponi

**Responsable de Integración:** Martín Prino

# Contenido

<i>1.1.1. Introducción</i> .....	3
<i>1.1.2. Metodología de desarrollo</i> .....	4
<i>1.1.3. Evaluación y comentarios</i> .....	4

### **1.1.1. Introducción**

El presente Informe de Integración busca reseñar y sintetizar el trabajo que se hizo en esta área de desarrollo. Como se señaló en el Plan realizado a principios de la etapa de elaboración, la metodología que se siguió para coordinar la integración tenía como elementos centrales el manejo de la gestión de configuración y de la verificación.

### **1.1.2. Metodología de desarrollo**

Como indicaba el Plan de Integración, el equipo trabajó "utilizando la metodología SCRUM y procurando seguir los principios básicos del desarrollo ágil". Por parte del equipo de desarrollo, se decidió no proceder con la generación de test automáticos a base de código Ruby para testear la integración. La aproximación a la detección y corrección de errores en esta etapa se realizaba haciendo pruebas de humo sobre lo que se terminaba teniendo en la branch Develop. Para la verificación de la integración se confiaba entonces en la efectividad de los test unitarios codificados y la capacidad del equipo de desarrollo de detectar y solucionar una integración fallida.

### **1.1.3. Evaluación y comentarios**

Se evalúa que la estrategia seguida de separar en ramas el desarrollo fue positiva pese a que no fue seguida estrictamente. Si bien la separación se respetó, se actuó con flexibilidad en los criterios originales (sobre todo en Develop).

Por lo general, en esta rama no se trabajó abriendo una rama paralela por historia sino que cada desarrollador trabajó en una propia. En función de esto fue necesario que los distintos aspectos de una historia (lógica, UI, test, etc) fueran integrados previos a su integración en Develop. Esta tarea estuvo a cargo de los desarrolladores implicados en cada caso. Los traspiés que ocurrían al aplicar esta metodología llevaron a que existieran instancias de código reemplazado por accidente, donde la integración generó una degradación en la capacidad de producción del equipo.

En cuanto a las ramas de Testing y Master se cumplió con sus cometidos. Con la primera de ellas se buscó que el equipo de Testing disponga de un ambiente actualizado haciendo los pasajes a medida que las historias se finalizaban. A lo largo del primer y segundo sprint, la subida a esta branch era realizada por el equipo de implementación. Luego de ver que esto generaba complicaciones para poder mantener un ambiente libre de modificaciones no monitorizadas, se decidió pasar la responsabilidad del mantenimiento de esta branch al responsable de verificación. La branch Master no fue utilizada en la mayoría del proyecto, ya que por falta de tiempo se decidió en cada demo utilizar la branch Testing para mostrar los avances.

La branch Master fue utilizada únicamente para el primer demo, y a partir del Lunes 25/11 es la que almacena el producto final.

La realización de pruebas unitarias a base de código fue una tarea que demandó mucho trabajo inesperado del lado del equipo. Supuso un desafío tecnológico que no fue previsto en la etapa inicial. No es parte de este informe el analizar en profundidad esto pero cabe señalar que estas permitieron prevenir errores lo que sin duda fue positivo para el desarrollo.