

Escuela de Ingeniería en Computación

Avantek Software

**“Análisis y aplicación de procesos de aseguramiento de la calidad
en el desarrollo de software”**

**Informe de Proyecto de Graduación para optar por el grado de
Bachiller en Ingeniería en Computación**

Nergil Andrés Rodríguez González

200418962

Prof. Asesor: Marlen Treviño Villalobos

San Carlos, 2011

Tabla de contenido

| | |
|--|-----------|
| Organización | 5 |
| Descripción del Problema | 6 |
| Personal Involucrado | 7 |
| Riesgos y Contingencias | 8 |
| Objetivos y Alcances | 9 |
| Enunciado de la Solución | 10 |
| Tecnologías y herramientas | 11 |
| Productos Esperados | 12 |
| Productos de la Fase de Conceptualización | 13 |
| Test Plan para el proyecto Caravan | 13 |
| Lista de Requerimientos..... | 14 |
| Tipos de Pruebas | 15 |
| Entregables de QA..... | 16 |
| Responsabilidad sobre las actividades | 17 |
| Estimación de los casos de Prueba..... | 17 |
| Creación de los Casos de Prueba..... | 19 |
| Investigación inicial acerca de la automatización de las pruebas | 20 |
| Ejecución del Primer Ciclo de Pruebas | 20 |
| Ejecución del Ciclo de Pruebas sobre Android | 22 |
| Otras Actividades | 23 |
| Total Control..... | 23 |
| Echo Being – Vision Board | 26 |
| Cámara de Comercio | 28 |
| Krome..... | 29 |
| Grass Master | 30 |
| CMI | 31 |
| Productos de los Ciclos de Pruebas | 35 |
| Conclusiones | 37 |
| Glosario | 38 |
| Anexos | 39 |
| Referencias | 40 |

Índice de Tablas

| | |
|--|----|
| Tabla 1: Riesgos y Contingencias..... | 8 |
| Tabla 2: Lista de Requerimientos..... | 14 |
| Tabla 3: Unidades que se pondrán a prueba | 15 |
| Tabla 4: Tipo de pruebas | 15 |
| Tabla 5: Documentos para los criterios de aceptación | 16 |
| Tabla 6: Entregables de QA..... | 16 |
| Tabla 7: Responsabilidad sobre las actividades | 17 |
| Tabla 8: Resultados obtenidos en Caravan en el primer ciclo de pruebas | 21 |
| Tabla 9: Resultados del ciclo de pruebas sobre la versión Android..... | 22 |
| Tabla 10: Información reportada en Jira..... | 35 |

Índice de Figuras

| | |
|--|----|
| Ilustración 1: Organigrama de Avantek Software | 5 |
| Ilustración 2: Ejemplo de creación de caso de prueba | 19 |
| Ilustración 3: Cambios en la Colorimetría..... | 31 |
| Ilustración 4: Cambios en los Indicadores Gráficos..... | 32 |
| Ilustración 5: Filtro para los Indicadores Gráficos..... | 32 |
| Ilustración 6: Mapa estratégico..... | 33 |
| Ilustración 7: Indicadores de cada uno de los objetivos del Mapa Estratégico | 33 |
| Ilustración 8: Detalle los Indicadores | 34 |

Resumen Ejecutivo

El presente documento es el informe de la práctica de especialidad, la cual se basa en la aplicación de técnicas y tecnologías que contribuyen al aseguramiento de la calidad. Además, el desarrollo de pruebas automatizadas que son de gran ayuda en este proceso de aseguramiento de la calidad.

“El aseguramiento de calidad del Software (SQA) es el set de actividades (incluidos la facilitación, entrenamiento, medición y análisis) requeridas para proveer la confianza adecuada de que los procesos están establecidos y son continuamente mejorados con el objetivo de producir un producto o servicio que cumple con los requerimientos y está listo para ser usado.”

Estas actividades, necesarias para proporcionar la confianza adecuada, no son solamente una revisión exhaustiva del producto final para verificar su calidad, son mucho más que eso, son los procesos y herramientas utilizadas a lo largo de todo el ciclo de vida del proyecto para asegurarse de que es desarrollado mediante procesos y métodos de probada eficacia, además capturar y manejar cualquier problema o desacuerdo que pudiera surgir, conjuntamente también se sigue un proceso adecuado para reporte, análisis y solución de estas anomalías, lo que facilita la mejora continua de los procesos o actividades internas que nos ayudan a conseguir un software de calidad.

Organización

Avantek Software, es una empresa nacida el 7 de septiembre de 2009 y comenzó sus operaciones el 1 de octubre de 2009, de una alianza estratégica entre Avantica Technologies y Northtek Software. Avantek Software está ubicada en Alajuela, San Carlos, Ciudad Quesada.

Avantek Software se especializa en desarrollar software de calidad en las aplicaciones web y dispositivos móviles, con las últimas tecnologías disponibles en el mercado.

La empresa se divide en tres áreas a nivel estructural, el departamento administrativo, el departamento comercial y el Departamento de Producción.

Su modelo de negocio como empresa de alta tecnología es la tecnología de información basada en los siguientes servicios:

- * Servicios de desarrollo de software.
- * Outsourcing.
- * Software a la medida.



Ilustración 1: Organigrama de Avantek Software

La práctica de especialidad se realizará en el departamento de Aseguramiento de la Calidad (QA) de Avantek Software.

Descripción del Problema

En el mundo del desarrollo del Software para ser competitivos los proyectos deben no sólo culminar en el menor tiempo posible y consumiendo la menor cantidad de recursos posibles, sino que también este producto debe cumplir con ciertos niveles de calidad que garanticen que el producto no solo cumple con los requerimientos sino que fue desarrollado siguiendo un proceso específico a lo largo de todo su desarrollo, probado y mejorado a lo largo de experiencias pasadas para el aseguramiento de la calidad de los productos o servicios.

Bajo esta premisa el aseguramiento de la calidad del software (QA, por sus siglas en inglés) debe valerse de metodologías, procesos y tecnologías ágiles y eficientes a lo largo de todo el ciclo de vida de los proyectos, para garantizar que dichos proyectos son desarrollados siguiendo los procesos correctos, los cuales buscan capturar defectos, errores de conceptualización, confusiones, malos entendidos, establecer métricas, coordinar revisiones periódicas, etc. en etapas tempranas, siendo este el mejor momento para analizar, planificar, y solucionar dichos asuntos, ya que está comprobado que consume más recursos, tiempo y dinero buscar soluciones a estos problemas, en muchos casos apresuradas y no ideales, ocasionando no solo pérdidas a las empresas, demoras en las fechas de entrega, sino también productos deficientes y disconformidad en el cliente, lo cual en si mismo causa que el producto no sea de calidad.

Cabe destacar también, el hecho de que los procesos de QA no deben verse como el anticuado, ineficiente y obsoleto método de testing en el cual al final del proyecto cuando ya estaba finalizado el producto se procedía a hacer cambios, este proceso lo realizaban otras personas en otros departamentos y no eran ni siquiera considerados parte del equipo de trabajo. Más bien, en la actualidad los Departamentos de QA son considerados parte vital del equipo de trabajo y su participación a lo largo de todo el ciclo de vida de los proyectos es considerada garante de productos de alta calidad y procesos de madurez y confianza.

Personal Involucrado en el Proyecto

| <i>Participante</i> | <i>Departamento</i> | <i>Puesto</i> | <i>Actividad</i> |
|----------------------------------|----------------------------------|-----------------|--|
| Nergil Andrés Rodríguez González | Aseguramiento de la Calidad (QA) | Practicante | <ul style="list-style-type: none"> • Establecer Plan de Trabajo. • Desarrollar Proyecto. • Presentar Informes. |
| Mario Nuñez Rodríguez | Aseguramiento de la Calidad (QA) | QA Lead | <ul style="list-style-type: none"> • Ayuda en toma de decisiones y orientación del proyecto. • Revisa tanto el desarrollo del proyecto como del informe. |
| Marlen Treviño Villalobos | ITCR | Profesor Asesor | <ul style="list-style-type: none"> • Seguimiento del proyecto • Asesora al practicante • Revisa informes |
| Carlos Montero Vasquez | Desarrollo | Practicante | <ul style="list-style-type: none"> • Desarrollador de la aplicación |
| Rodrigo Vargas | Desarrollo | Project Manager | <ul style="list-style-type: none"> • Análisis de requerimientos. • Dirigir el desarrollo del software. |

Riesgos y Contingencias

Tabla 1: Riesgos y Contingencias

| Riesgo | Estrategia de Mitigación | Impacto |
|---|--|---------|
| Los retrasos en la entrega de elementos completos a probar por parte del departamento de Desarrollo afectarían los plazos de prueba y la calidad final de la versión. | Comunicación con producción y desarrollo para notificar de los retrasos y ajuste de versión de la aplicación para permitir que las actividades de prueba se realicen correctamente. | Alto |
| Los retrasos en el tiempo de respuesta para la corrección de errores críticos podrían tener un impacto en las fechas del proyecto. | Fuerte gestión de la resolución de errores se requerirían por parte del departamento de Desarrollo para garantizar los errores son corregidos y están disponibles para una nueva prueba en el tiempo programado. | Alto |
| El equipo de pruebas, desarrollo o equipos de PM requieren uno o del otro y no están disponibles. Esto podría retrasar las actividades del proyecto. | Asegurar la disponibilidad de los diferentes equipos durante las etapas críticas del proyecto ó al menos garantizar que pueden ser contactados, | Medio |
| Características de los elementos a probar no están en condiciones de ser probadas. | El equipo de pruebas registrará las características no probadas y pedirá a la PM evaluar el riesgo empresarial de la liberación de las características no probadas. | Bajo |
| Malos entendidos o falta de detalle en los requerimientos del proyecto. | Asegurar una buena comunicación entre el departamento de desarrollo y producción y el departamento de QA. | Alto |
| Falta de seguimiento de las pulgas reportadas a lo largo de las diferentes versiones de la aplicación. | Asegurar una buena comunicación entre el departamento de desarrollo y producción y el departamento de QA. | Alto |
| Aparición de dependencias inesperadas, esto requeriría una revisión de los escenarios de prueba y los casos de prueba. | Informes acerca de las dependencias se actualiza y comunican sin demora a fin de permitir la revisión oportuna de los escenarios de prueba y casos de prueba. | Bajo |

Objetivos y Alcances

Objetivo General

Comprender y aplicar las metodologías, tecnologías, herramientas y buenas prácticas existentes en el departamento de aseguramiento de la calidad de Avantek Software, aplicando estos en un proyecto desarrollado en la empresa.

Objetivos Específicos

- Aplicar metodologías, dinámicas y las tecnologías existentes en Avantek software para seguimiento, control y acciones tomadas en los proyectos para garantizar los estándares de calidad requeridos por la empresa y los clientes.
- Elaborar un plan de pruebas para el proyecto asignado, para garantizar la calidad en las diferentes etapas del proyecto.
- Utilizar diferentes conceptos de QA lo largo de las distintas etapas de las pruebas, tales como: Análisis, Planeamiento, Implantación, Ejecución y Resultados.
- Aplicar diferentes tipos de pruebas como las pruebas de homo, pruebas exploratorias, pruebas de funcionalidad, pruebas de regresión, verificación de pulgas etc.
- Planificar e implementar la automatización de pruebas para el proyecto y utilizar las tecnologías implantadas en la empresa para su desarrollo.

Alcances

- Análisis de Requerimientos.
- Estimación de Test cases.
- Creación de Test Cases.
- Definición de Alcances de los ciclos de pruebas
- Creación de Test Plans según los alcances establecidos.
- Análisis y creación de pulgas.
- Análisis del entorno.
- Desarrollo de pruebas automáticas.
- Ejecución de ciclos de pruebas de los distintos Test Plans registrados
- Informe de resultados de las pruebas.
- Criterios de aceptación.

Enunciado de la Solución

Debido a que Avantek Software es una empresa con experiencia en el área del aseguramiento de la calidad y posee un departamento de QA que sigue creciendo e innovando, se pretende abarcar los siguientes puntos a lo largo de la práctica de especialidad:

- Retomar y profundizar acerca de los conceptos básicos de QA.
- Aprender sobre las metodologías implantadas en Avantek Software, específicamente en el departamento de QA, para el aseguramiento de la calidad y el control de los diferentes proyectos, así como las dinámicas de comunicación tanto a nivel de trabajo como a nivel social, ya que la empresa fomenta mucho las relaciones interpersonales de todos los departamentos y empleados.
- Proceso de training acerca de las diferentes tecnologías y documentaciones utilizadas en los proyectos que tiene a cargo el departamento de QA.
- Poner en práctica todos los conceptos, prácticas, metodologías, tecnologías e informes a lo largo de todo el ciclo de vida de un proyecto desarrollado en la empresa (*Caravan*).
- Investigar, planificar y desarrollar pruebas automatizadas para el testing de ciertas aéreas del software a lo largo de todo el avance del proyecto Caravan.
- Colaborar en los procesos de QA de otros proyectos que tiene a su cargo el departamento de QA, con el fin de ganar experiencia no solo técnica sino también en aéreas como la comunicación y seguimiento con personal externo a la empresa (por ejemplo desarrolladores de Avantica).

Tecnologías/Herramientas

- **TestLink:** Herramienta utilizada para el desarrollo y descripción de los casos de prueba, en ella se asigna un nombre descriptivo y un identificador a cada caso de prueba planificado en el documento de estimación, una descripción detallada de el objetivo, los pasos a seguir para su correcta revisión, el comportamiento esperado de la aplicación para cumplir con el caso de prueba, el comportamiento con el cual se considera el caso como fallido y un tiempo estimado de ejecución para cada uno de los mismos. Además, incluye diversas funcionalidades muy útiles en la planificación de ciclos de prueba, asignación de casos de prueba e informes de resultados de los ciclos de prueba, entre otros.
- **Jira:** Herramienta utilizada para la asignación de tareas, seguimiento y control de los proyectos, asignación de tareas a los recursos, reporte y estado de defectos, comentarios y estados de las tareas y reporte y control del tiempo utilizado para cada una de las tareas de todos los miembros del equipo de trabajo para cada proyecto.
- **Eclipse SDK Versión: 3.6.2:** Ambiente de desarrollo utilizado para el desarrollo de las pruebas automáticas que se aplicaran en la aplicación de Caravan.
- **Android SDK:** Son las siglas de *Android Software Development Kit*. Es un kit de desarrollo con el que podremos desde desarrollar aplicaciones, hasta ejecutar un emulador del sistema Android en la versión que queramos. En el caso de la aplicación Caravan será utilizado para configurar el emulador del sistema Android sobre el cual se ejecutaran las pruebas automáticas.
- **Herramientas de Microsoft:** A lo largo del proyecto se elaboraran documentos y se completaran plantillas utilizando diferentes herramientas de Microsoft, como por ejemplo Microsoft Office Word, Microsoft Office Excel, Microsoft Office Project.

Productos Esperados

1. Test Plan de Caravan
 - a. Lista de requerimientos.
 - b. Las unidades que se pondrán a prueba.
 - c. Tipos de Pruebas (Funcionalidad, Requerimientos, usabilidad, seguridad, etc.).
 - d. Roles y responsabilidades.
 - e. Riesgos y Contingencias.
 - f. Entregables de QA.
 - g. Responsabilidad sobre las actividades.

2. Estimación de los casos de prueba del proyecto Caravan
 - a. Especificación de los Casos de Prueba según los requerimientos tanto de la versión Móvil como de la versión WEB.
 - b. Priorización de los casos de prueba.
 - c. Estimación tanto del tiempo de creación de los casos de prueba como del tiempo de ejecución de los mismos.

3. Creación de los casos de prueba
 - a. Creación de los casos de prueba en TestLink.
 - b. Especificación de los pasos a seguir, resultados esperados y caso de error.
 - c. Tiempo estimado de ejecución de cada caso de prueba.

4. Investigación inicial acerca de la automatización de las pruebas.

Productos de la Fase de Conceptualización

Debido a que Caravan es un proyecto que presenta la particularidad particular de que surge con una idea básica de que servicio se pretende brindar pero no está establecido que funcionalidades tendrá ni de qué manera serán desarrolladas. Se encomendó al departamento de QA dar un seguimiento y aplicar los procesos necesarios para controla y guiar las características, funcionalidades y comportamientos cambiantes que se irán presentando a lo largo de todo el desarrollo del proyecto para garantizar de que al final se obtenga un producto de calidad que podrá ser comercializado debidamente por la empresa.

Test Plan para el proyecto Caravan

El documento de Test Plan es confeccionado siguiendo las plantillas que Avantek Software ha generado a lo largo de los años y experiencias aprendidas en diversos proyectos, es debido a esto, que en el presente documento solamente se presentará la información y formatos autorizados por Avantek Software a fin de garantizar que su propiedad intelectual sea debidamente salvaguardada

Durante el proceso de elaboración del test plan se obtuvieron los siguientes productos:

Lista de requerimientos:

La lista de requerimientos establece las funcionalidades y características básicas que actuaran como base para la elaboración del proyecto; como ya se menciona anteriormente por la naturaleza del proyecto estas características estas sujetas a cambios, pero representan en esencia el objetivo que se pretende alcanzar.

Tabla 2: Lista de Requerimientos

| Lista de Requerimientos |
|--|
| Servicios de API para la interacción Web/Móvil |
| Servicios de autenticación tanto para usuarios web como para usuarios móviles |
| Almacenar el historial de localización del usuario y las rutas seguidas |
| Almacenar el perfil de usuario y enviarlo en caso de una petición ya sea Web o Móvil |
| Enviar notificaciones y alertas al grupo de Caravan en caso de eventos (que un miembro del grupo se sale de la ruta establecida, si un miembro se detiene por un determinado tiempo, etc.) |
| Importar amigos de la cuenta de Facebook (y posiblemente Google) a la aplicación Móvil |
| Almacenar en el servidor la lista de amigos de cada usuario con el fin de que cuando este los seleccione de la lista para invitarlos a un grupo, el servidor envíe el correspondiente correo electrónico a los invitados de la lista |
| El servidor debe ser capaz de contestar a cualquier consulta tanto de la aplicación móvil como la aplicación Web |
| La página (web) y pantalla (móvil) de registro permiten a los usuarios de Caravan registrarse solicitando la siguiente información: Nombre de usuario, correo electrónico y contraseña. |
| El sistema de autenticación le permitirá a los usuarios de Caravan conectarse tanto en la aplicación móvil como en la aplicación Web |
| La aplicación web y móvil permitirán a los usuarios actualizar o modificar su información personal y su estatus de compartir su ubicación actual |
| La aplicación web y móvil permitirá a los usuarios importar sus amigos de redes sociales como Facebook con el fin de agregarlos a la lista de amigos de Caravan. |
| La aplicación web y móvil permitirá a los usuarios seleccionar amigos de su lista de contactos e invitarlos a un viaje o grupo |
| La aplicación web y móvil permitirá a los usuarios seguir las ubicaciones de sus amigos durante el viaje, mostrándolos mediante marcas en el mapa (solamente si el amigo esta compartiendo su localización actual). |
| La aplicación web y móvil permitirá a los usuarios decidir si desean o no compartir su ubicación actual. |
| La aplicación web y móvil permitirá a los usuarios suministrar su propio "Avatar", seleccionándolo de la lista de avatar predefinido o seleccionando su imagen de la red social. |
| Todo grupo o viaje tiene su propio nombre, identificador, punto de partida y destino; esta información estará disponible para los miembros del grupo y los amigos con los que este compartiendo su ubicación |

Las unidades que se pondrán a prueba:

Estas unidades constituyen las diferentes secciones o funcionalidades, estas deberán abarcar todos los requerimientos establecidos en el producto anterior. Esta lista sirve de base para la elaboración de los diferentes Test Suite en los cuales se subdividirán los casos de prueba.

Tabla 3: Unidades que se pondrán a prueba

| Unidades que se Pondrán a Prueba |
|---|
| Sistema de autenticación (tanto en la aplicación Web como en la aplicación Móvil). |
| Sistema de registro (tanto en la aplicación Web como en la aplicación Móvil) |
| Almacenamiento y servicios del servidor (perfiles de usuario y sus listas de amigos, notificaciones, alertas, servicios, etc.) |
| Editar el perfil de usuario y compartir la ubicación actual (tanto en la aplicación Web como en la aplicación Móvil). |
| Sistema de importar y almacenar amigos desde la red social (tanto en la aplicación Web como en la aplicación Móvil) |
| Sistema de creación de viaje o grupo (tanto en la aplicación Web como en la aplicación Móvil) |
| Sistema de seguimiento de viaje (sistema de localización en tiempo real, información de los usuarios en el grupo, marcas en el mapa, alertas, etc.) (tanto en la aplicación Web como en la aplicación Móvil) |

Tipos de Pruebas:

Existen diferentes tipos de pruebas posibles que a las que se somete un producto de software para determinar su nivel de calidad y detección de defectos; la siguiente tabla indica a qué tipo de pruebas se pretende someter a los diferentes productos del presente proyecto.

Tabla 4: Tipo de pruebas

| Prueba | Aplica |
|---------------------------------|---------------|
| Requerimientos | X |
| Funcionalidad | X |
| Seguridad | X |
| Usabilidad | X |
| Rendimiento | |
| Aceptación | |
| (Ambiente de producción) | |

Documentos para los criterios de aceptación:

La siguiente tabla contiene los diferentes documentos que serán entregados y utilizados como criterios de aceptación.

Tabla 5: Documentos para los criterios de aceptación

| Documentación | Herramienta |
|-----------------------------|--|
| Cobertura de Test Cases | Microsoft Excel |
| Manual de Test Cases | TestLink es una herramienta web utilizada para la administración del desarrollo de un proyecto de software |
| Resultados de los Test Plan | Microsoft Excel |
| Criterios de Aceptación | Microsoft Word |

Entregables de QA:

La siguiente tabla se indica las distintas documentaciones que se harán por parte del departamento de QA.

Tabla 6: Entregables de QA

| Documentación | Entregable |
|------------------------------|--|
| Test Plan | <ul style="list-style-type: none"> Documento de Test Plan |
| Especificación de Test Cases | <ul style="list-style-type: none"> Manual Test Cases registrados en TestLink |
| Test Result | <ul style="list-style-type: none"> Reporte de TestLink Test Result QA-FO-001 |
| Métricas | <ul style="list-style-type: none"> Número de Test Cases |

Responsabilidad sobre las actividades:

Como en cualquier equipo de trabajo, existen diferentes actividades a realizar y un responsable o encargado de cumplirlas, en la siguiente tabla se especifican estas actividades y su respectivo encargado.

Tabla 7: Responsabilidad sobre las actividades

| Principales Actividades | Responsable |
|--|-------------------------------|
| Elaboración el documento de requisitos del sistema | Carlos Montero/Rodrigo Vargas |
| Elaboración el documento de requisitos de hardware y software | Carlos Montero/Rodrigo Vargas |
| Elaborar la especificación de los Casos de Prueba del sistema | Nergil Rodríguez |
| Ejecución de las pruebas | Nergil Rodríguez |
| Verificación de las pulgas resultas | Nergil Rodríguez |
| Ejecutar pruebas de regresión hasta que los análisis han cumplido todos los criterios para su aprobación | Nergil Rodríguez |
| Enviar el informe de resumen de las pruebas al equipo de gestión y el desarrollo. | Mario Núñez |

Estimación de los casos de prueba del proyecto Caravan

El proceso de estimación de casos de prueba tiene como objetivo desglosar y clasificar los requisitos previamente especificados en el Test Plan; definir los diferentes casos de prueba para cada uno de los Test Suite resultantes de la clasificación de los requisitos; además, se hace una estimación preliminar de el tiempo de confección de cada uno de los casos de prueba y el tiempo de ejecución de los mismos, así como su prioridad según las funcionalidades y requisitos especificados.

Para el proceso de estimación se siguió el siguiente formato:

| Caravan | | | Prioridad | Estimación de creación del Test Cases | Estimación de ejecución del Test Cases | |
|--------------|------------|-----------|---------------------------|---------------------------------------|--|---------|
| Móvil | | | | | | |
| | Test Suite | | | | | |
| | | Test Case | Descripción del Test Case | Alta/Media /Baja | Minutos | Minutos |
| | | Test Case | Descripción del Test Case | Alta/Media /Baja | Minutos | Minutos |
| | Test Suite | | | | | |
| | | Test Case | Descripción del Test Case | Alta/Media /Baja | Minutos | Minutos |

| | | | | | | |
|------------|------------|-----------|---------------------------|-----------------|---------|---------|
| | | Test Case | Descripción del Test Case | Alta/Media/Baja | Minutos | Minutos |
| | | Test Case | Descripción del Test Case | Alta/Media/Baja | Minutos | Minutos |
| Web | | | | | | |
| | Test Suite | | | | | |
| | | | | | | |

Algunos ejemplos de los test cases son:

| <i>Opciones a Probar</i> | <i>Prioridad de Ejecución</i> | <i>Tiempo Estimado de Creación</i> | <i>Tiempo Estimado de Ejecución</i> |
|--|-------------------------------|------------------------------------|-------------------------------------|
| Verificar si se permite el acceso a la pantalla de registro | Alta | 5 | 3 |
| Verificar que la interfaz de usuario de la pantalla de registro se despliega correctamente | Alta | 5 | 10 |
| Verificar que la identidad de Caravan es claramente visible y reconocible en la pantalla (Logo, Nombre, etc.). | Baja | 5 | 3 |
| Verificar que los campos, textos y botones están alineados y no ocultan texto | Media | 5 | 5 |
| Verificar que todos los campos tienen la longitud de tamaño permitido | Media | 5 | 10 |
| Verificar que la opción de compartir la localización actual está por defecto en "ON" | Media | 5 | 5 |
| Verificar el formato de la dirección de correo electrónico | Alta | 5 | 10 |

Creación de los casos de prueba

La creación de los casos de prueba se realiza utilizando la herramienta TestLink, la cual facilita la creación de los mismos y ayuda en la creación de los Test Plan y su respectiva ejecución.

A lo largo de esta fase de creación de casos de prueba, los mismos se especifican siguiendo la estructura definida durante el proceso de estimación de casos de prueba, además, se agrega una descripción, se detallan los pre-requisitos y los pasos a seguir durante cada caso de prueba y también se especifican los resultados esperados y los resultados erróneos.

Ejemplo:

[Ver anexo número 4](#)

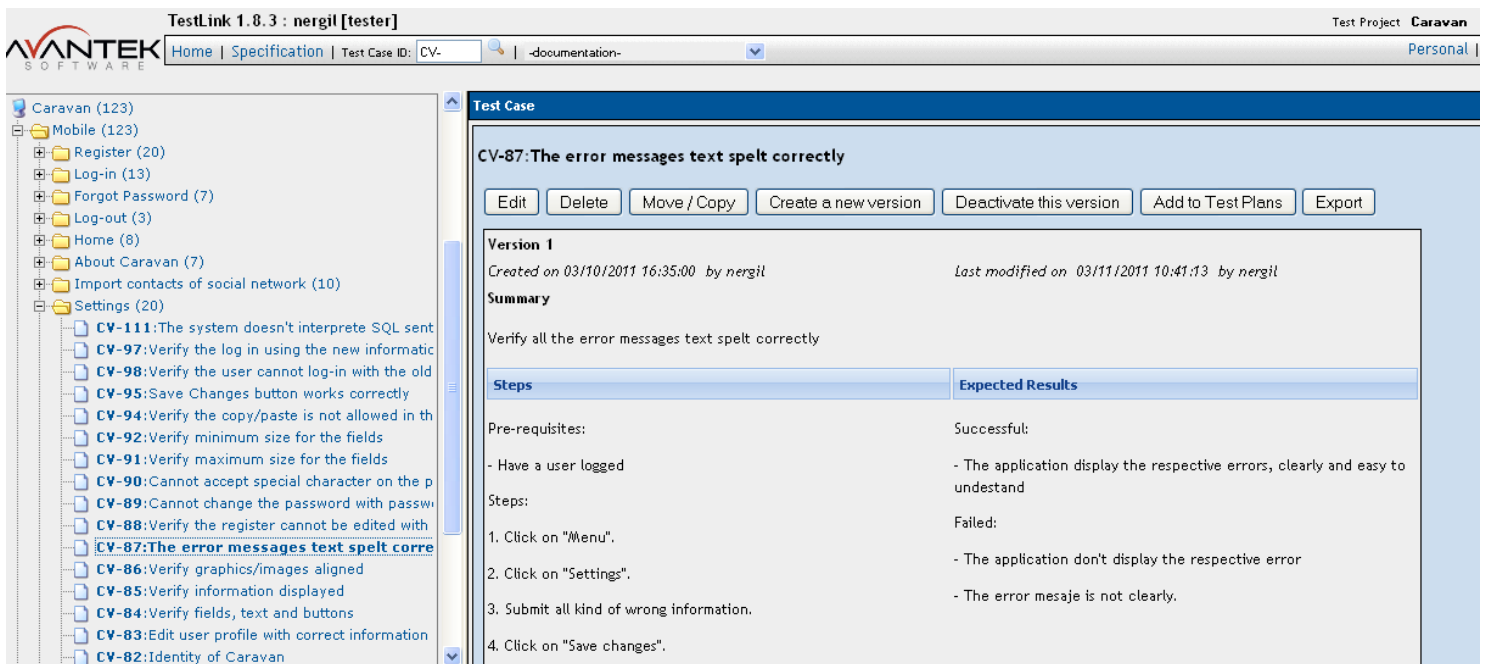


Ilustración 2: Ejemplo de creación de caso de prueba

Investigación inicial acerca de la automatización de las pruebas

Como parte del proceso de aseguramiento de la calidad se pretende implementar un proceso de automatización de las pruebas de software; con este fin se estableció que las pruebas automáticas se desarrollaran utilizando la herramienta **Selenium RC** bajo el ambiente de desarrollo **Eclipse** utilizando el lenguaje de desarrollo **JAVA**.

Como producto de esta investigación inicial se obtiene los siguientes documentos adjuntos:

[Installation and basic tutorial of Selenium IDE](#)

[Configuring Selenium RC in Eclipse](#)

[Basic Automated Test Case in Eclipse](#)

Ejecución del primer ciclo de pruebas

Una vez que se nos hizo entrega del Build correspondiente a la primera iteración de Caravan, nos avocamos a la tarea de ejecutar este primer ciclo de pruebas:

- Creación y envío de la estimación y alcance del ciclo de pruebas, utilizando la plantilla existente en Avantek para estos propósitos.
- Ejecución de los ciclos de prueba correspondientes (hasta el momento solamente se han ejecutado los casos de prueba en el navegador Firefox), utilizando la herramienta *Test Link* para guiar el proceso de pruebas y reportar el estatus de cada uno de los casos de prueba correspondientes a este ciclo de pruebas.
- Registro en la herramienta Jira de todos los defectos encontrados en la aplicación durante los ciclos de pruebas.
- Elaboración y envío del documento de resultados de las pruebas, utilizando la plantilla existente en Avantek para estos propósitos. En este caso, ya que Caravan es principalmente el proyecto de práctica de especialidad si es importante especificar con más detalle los resultados obtenidos en este ciclo de pruebas:

Tabla 8: Resultados obtenidos en Caravan en el primer ciclo de pruebas

| <i>Métrica</i> | <i>Resultado Obtenido</i> |
|---|---------------------------|
| Fecha de inicio del ciclo de pruebas. | 03/05/2011 |
| Fecha de conclusión del ciclo de pruebas. | 04/05/2011 |
| Horas utilizadas en el ciclo de pruebas | 11 |
| Total de Test Cases | 75 |
| Satisfactorios | 51 (68%) |
| Fallidos | 11 (15%) |
| No ejecutados (blocked) | 13 (17%) |
| Cantidad de bugs reportados | 11 |
| <i>Critical</i> | 0 |
| <i>Major</i> | 2 |
| <i>Normal</i> | 7 |
| <i>Minor</i> | 2 |
| <i>Trival</i> | 0 |
| <i>Enhancement</i> | 0 |

- Ejecutar la codificación de los casos de prueba automáticos correspondientes a este primer build; utilizando la herramienta Selenium RC y su librería para el lenguaje Java, en el ambiente de desarrollo Eclipse. Como resultado de este proceso se obtuvo:
 - El desarrollo de un archivo de configuración para la ejecución de los casos de prueba automáticos.

Tabla 9: Información del archivo de configuración para los casos de prueba automáticos

| Parámetro | Función |
|-----------------------------|--|
| RESULT_FILE_ENCODING | Codificación del archivo en el cual se almacenaran los resultados de la ejecución de los casos de prueba automáticos (ejemplo: UTF-8) |
| OPENQA_URL | URL a la cual se conectará el servidor Selenium para ejecutar los casos de prueba (ejemplo: <code>http://192.168.100.252/</code>) |
| PORT | Puerto mediante el cual se accederá a la URL en la cual se desea ejecutar los casos de prueba automáticos (ejemplo: 4444) |
| RESULT_PATH | Directorio en el cual se almacenaran los archivos que informan los resultados de la ejecución de los casos de prueba automáticos. (ejemplo: <code>C:\\Results of the test</code>) |
| BROWSER | Browser en el cual se ejecutarán los casos de prueba automáticos (ejemplo: firefox) |

- 41 Casos de prueba automáticos correspondientes a las funcionalidades implementadas en el primer build de Caravan.

Ejecución del ciclo de pruebas sobre la versión Android

El día 23 de mayo del 2011 se nos hace entrega del build correspondiente a la versión de Caravan para dispositivos Android, por lo cual se inició inmediatamente el ciclo de pruebas correspondiente:

- Creación y envío de la estimación y alcance del ciclo de pruebas, utilizando la plantilla existente en Avantek para estos propósitos.
- Ejecución de los ciclos de prueba, utilizando la herramienta *Test Link* para guiar el proceso de pruebas y reportar el estatus de cada uno de los casos de prueba correspondientes a este ciclo de pruebas.
- Registro en la herramienta Jira de todos los defectos encontrados en la aplicación durante los ciclos de pruebas.
- Elaboración y envío del documento de resultados de las pruebas, utilizando la plantilla existente en Avantek para estos propósitos. En este caso, ya que Caravan es principalmente el proyecto de práctica de especialidad si es importante especificar con más detalle los resultados obtenidos en este ciclo de pruebas:

Tabla 10: Resultados del ciclo de pruebas sobre la versión Android

| <i>Métrica</i> | <i>Resultado Obtenido</i> |
|---|---------------------------|
| Fecha de inicio del ciclo de pruebas. | 23/05/2011 |
| Fecha de conclusión del ciclo de pruebas. | 24/05/2011 |
| Horas utilizadas en el ciclo de pruebas | 9.45 |
| Total de Test Cases | 75 |
| Satisfactorios | 45 (60%) |
| Fallidos | 12 (16%) |
| No procesados | 10 (13%) |
| No ejecutados (blocked) | 8 (11%) |
| Cantidad de bugs reportados | 13 |
| <i>Critical</i> | 0 |
| <i>Major</i> | 3 |
| <i>Normal</i> | 8 |
| <i>Minor</i> | 2 |
| <i>Trival</i> | 0 |
| <i>Enhancement</i> | 0 |

Otras Actividades

TotalControl

Este proyecto presenta la característica especial de sub dividirse en proyectos más cortos debido a las diferentes plataformas sobre las cuales son desarrolladas las aplicaciones (Android, iPhone). Estas aplicaciones deberán pasar por los diferentes procesos de aseguramiento de la calidad a fin de presentar al cliente un producto de calidad y que logre satisfacer sus exigencias.

Android VidCam:

- Colaborar en la ejecución de un ciclo de pruebas de un modulo en especifico en 2 versiones, pruebas exploratorias a otras 5 versiones y verificación de pulgas de un proyecto en particular desarrollado sobre la plataforma *Android*.

Tareas:

- Training acerca de la aplicación, funcionalidades, comportamiento esperado y defectos encontrados que deberán ser revisados nuevamente (luego de la respectiva corrección por parte del desarrollador).
- Revisión de los casos de prueba existentes, y en especial los correspondientes a los defectos previamente encontrados, el Test Plan correspondiente a la nueva funcionalidad a probar.
- Creación de los casos de prueba correspondientes a la nueva funcionalidad a probar, dichos casos de prueba fueron desarrollados en la herramienta "*Test Link*", además, la elaboración y entrega del Scope respectivo para este ciclo de pruebas a ejecutar, esto siguiendo las plantillas utilizadas en Avantek para este fin .
- Ejecución del Test plan y casos de prueba en 2 de las 7 versiones de la aplicación, además pruebas exploratorias sobre las restantes versiones
- Confección y entrega del documento de resultados de los ciclos de pruebas ejecutados, utilizando también para este fin una plantilla de Avantek diseñada para este propósito.

- Ingreso en la herramienta *Jira* de los defectos encontrados y reportar como verificados de los defectos solucionados. Especificando tanto la descripción, como los pasos para reproducir el defecto, el comportamiento esperado y el comportamiento encontrado de cada uno de los defectos, además evaluando tanto la severidad como la prioridad de los mismos.

Android VidCam (Tablet):

- Posteriormente al ciclo de pruebas sobre la plataforma Android, se determino que se debían efectuar pruebas acerca del comportamiento de la aplicación sobre el dispositivo Android Tablet, con este fin de debía preparar de un ambiente de pruebas para estudiar el comportamiento de la aplicación en este novedoso dispositivo y así poder aclarar ciertas dudas al cliente.

Tareas:

- Instalación y configuración del ambiente de pruebas para la aplicación, este ambiente de pruebas fue configurado utilizando la librería de Android para *Netbeans*.
- Instalación de la aplicación (versión DEMO) en el emulador preparado anteriormente.
- Ejecución de pruebas exploratorias sobre la aplicación en el emulador, así como análisis del comportamiento de la misma ante este nuevo ambiente.
- Reporte de comportamiento de la aplicación en este dispositivo.

iPhone VidCam:

- En la versión para el dispositivo iPhone de la aplicación VidCam previamente se habían creado algunos casos de prueba para las funcionalidades básicas. Se le informa al departamento de QA acerca de las nuevas funcionalidades que deberán ser probadas

Tareas:

- Elaboración de los casos de pruebas faltantes para las nuevas funcionalidades. Para esto nuevamente se utilizó la herramienta *Test Link*.
- Elaboración y entrega del documento de Alcance del ciclo de pruebas siguiendo la plantilla utilizada en Avantek para este propósito.

- Posteriormente se nos hace entrega del build que contiene las nuevas funcionalidades y se procede con el ciclo de pruebas previamente planificado. Este ciclo de pruebas se nos solicitó previamente que se ejecutara sobre la versión de VidCam 32 debido a que esta era suficientemente representativa para esta revisión preliminar.

Tareas:

- Ejecución del ciclo de pruebas previamente planificado para el aseguramiento de la calidad de las nuevas funcionalidades de esta aplicación.
 - Registro en la herramienta *Jira* de todos los defectos encontrados en la aplicación, especificando tanto la descripción, como los pasos para reproducir el defecto, el comportamiento esperado y el comportamiento encontrado de cada uno de los defectos, además evaluando tanto la severidad como la prioridad de los mismos.
- Posteriormente el cliente envía un video de retroalimentación en el cual muestra los defectos detectados durante sus propias pruebas.

Tareas:

- Estudiar el video suministrado por el cliente y reproducir el error en el iPhone utilizado para pruebas a fin de tener una mejor idea de sus efectos y causas. (esta tarea se llevó a cabo en conjunto con el desarrollador que tenía a cargo la corrección de los defectos encontrados en la aplicación).
- Registro en la herramienta *Jira* de todos los defectos encontrados así como agregar comentarios del comportamiento observado y sus posibles causas a fin de colaborar en el proceso de solución de los defectos encontrados. Especificando tanto la descripción, como los pasos para reproducir el defecto, el comportamiento esperado y el comportamiento encontrado de cada uno de los defectos, además evaluando tanto la severidad como la prioridad de los mismos.

EchoBeing – Vision Board

- Inicialmente se requería de un ciclo de pruebas sobre las funcionalidades básicas desarrolladas hasta el momento, para poder preparar el ciclo de pruebas correspondiente a esta aplicación fue necesario efectuar un análisis de los alcances y características de la aplicación, con dicho propósito se verificaron los documentos correspondientes a estas características y alcances, en especial los *Mockups* proporcionados por el cliente

Tareas:

- Creación de los Test Cases, en la herramienta *Test Link*, para las pruebas de las funcionalidades desarrolladas hasta el momento en esta aplicación, utilizando como referencia el documento formal de funcionalidades a desarrollar y los *Mockups* proporcionados por el cliente.
- Creación y envió de la estimación y alcance del ciclo de pruebas utilizando la plantilla existente en Avantek para este propósito.
- Ejecución del Ciclo de Pruebas en los 3 diferentes browser especificados (Firefox, Safari, Internet Explorer) utilizando la herramienta *Test Link* para guiar el proceso de pruebas y reportar el estatus de cada uno de los casos de prueba correspondientes a este ciclo de pruebas.
- Registro en la herramienta Jira de todos los defectos encontrados en la aplicación, especificando tanto la descripción, como los pasos para reproducir el defecto, el comportamiento esperado y el comportamiento encontrado de cada uno de los defectos, además evaluando tanto la severidad como la prioridad de los mismos.
- Creación y envió del documento de resultado de las pruebas, utilizando la plantilla existente en Avantek con este propósito.

- Posteriormente se efectuaron las correcciones correspondientes a los defectos encontrados en el ciclo de pruebas anterior y le corresponde al departamento de QA iniciar el proceso de Bug Verification a fin de garantizar que los defectos han sido solucionados satisfactoriamente.

Tareas:

- Ejecutar la verificación de que los Bugs previamente detectados en la aplicación han sido solucionados satisfactoriamente (en los 3 diferentes browser especificados).
 - Registro en la herramienta Jira de todos los defectos encontrados en la aplicación durante el proceso de bug verification , especificando tanto la descripción, como los pasos para reproducir el defecto, el comportamiento esperado y el comportamiento encontrado de cada uno de los defectos, además evaluando tanto la severidad como la prioridad de los mismos.
 - Elaboración y envió del documento de resultados de las pruebas, utilizando la plantilla existente en Avantek con este propósito.
- Más tarde se implementaron nuevas funcionalidades para esta aplicación y nuevamente se le solicito al departamento de QA que efectuara las pruebas correspondientes para esta aplicación.

Tareas:

- Creación de los casos de prueba correspondientes a las nuevas Funcionalidades de la aplicación utilizando la herramienta *Test Link*.
- Bug Verification de los defectos previamente detectados (en los 3 diferentes browser especificados).
- Ejecución de los casos de prueba correspondientes a las nuevas funcionalidades de la aplicación en los 3 diferentes browser especificados, utilizando la herramienta *Test Link* para guiar el proceso de pruebas y reportar el estatus de cada uno de los casos de prueba correspondientes a este ciclo de pruebas.
- Registro en la herramienta Jira de todos los defectos encontrados en la aplicación durante el ciclo de pruebas.
- Elaboración y envió del documento de resultados de las pruebas, utilizando la plantilla existente en Avantek con este propósito.

Cámara de Comercio

- Para este proyecto se debe inicial con el proceso normal de aseguramiento de la calidad, en este caso, no existía ningún caso de prueba previamente documentado para esta aplicación. También es importante señalar que para este proyecto se solicitó que las pruebas fueran efectuadas en los siguientes browsers: Firefox, Internet Explorer 7 é Internet Explorer 8.

Tareas:

- Creación de los casos de prueba correspondientes, utilizando la herramienta *Test Link*.
 - Informe de los defectos encontrados exploratoriamente durante la creación de los casos de prueba respectivos. Al ser estos defectos detectados en una revisión no formal y ya que la aplicación aun se encontraba en etapa de desarrollo, el informe de los defectos sencillamente se completo utilizando un correo electronico
- Más tarde se informa que para este primer build también se iba a incluir una nueva funcionalidad que previamente no fue incluida en la elaboración de los casos de prueba.

Tarea:

- Creación de los casos de prueba correspondientes al Directorio Comercial utilizando la herramienta *Test Link*.

- Posteriormente se nos hace entrega de este primer build para así poder iniciar el oportuno ciclo de pruebas.

Tareas:

- Creación y envió de la estimación y alcance del ciclo de pruebas, utilizando la plantilla existente en Avantek para estos propósitos.
- Ejecución de los ciclos de prueba correspondientes a los distintos browsers, utilizando la herramienta *Test Link* para guiar el proceso de pruebas y reportar el estatus de cada uno de los casos de prueba correspondientes a este ciclo de pruebas.
- Registro en la herramienta Jira de todos los defectos encontrados en la aplicación durante los ciclos de pruebas.
- Elaboración y envió del documento de resultados de las pruebas, utilizando la plantilla existente en Avantek para estos propósitos.

Krome

- Debido a que este es un proyecto muy grande y que ya lleva muchísimo tiempo de desarrollo se debía iniciar con una capacitación acerca de las funcionalidades y herramientas previamente desarrolladas para el uso de esta aplicación, con la idea no solamente de colaborar en la creación, planificación y ejecución de ciclos de pruebas sino también en la automatización de los casos de pruebas correspondientes a algunos módulos en específico de dicha aplicación:

Tareas:

- Training referente al proyecto Krome así como las herramientas y tecnologías implicadas, tanto para MAC como para Windows.
- Training de automatización de Krome, en el caso de este proyecto la automatización de los casos de prueba es llevada a cabo utilizando la herramienta Selenium RC para el lenguaje Java, en el ambiente NetBeans.

- Posteriormente se asigno la tarea de codificar los casos de prueba automáticos correspondientes a los módulos de Photo Uploader Web HTML y Facebook Sharing para una posterior presentación del avance del proyecto en general

Tareas:

- Codificación de los casos de prueba automáticos del Photo Uploader WEB HTML, poniendo en práctica lo aprendido durante el training correspondiente, utilizando las instrucciones y lenguaje correspondientes a la herramienta Selenium RC junto al lenguaje Java en el ambiente NetBeans.
- Codificación de los casos de prueba automáticos del Facebook Sharing, utilizando las mismas tecnologías.
- Colaborar en la presentación del Demo de Automatización presentado ante el Project Manager de dicho proyecto, el cual se llevo a cabo mediante una video conferencia.

Grass Master

- En este proyecto previamente se habían ejecutado ciclos de pruebas y existían una cantidad de bugs previamente reportados ya habían pasado por el proceso de bug fixing y debían ser verificados.

Tareas:

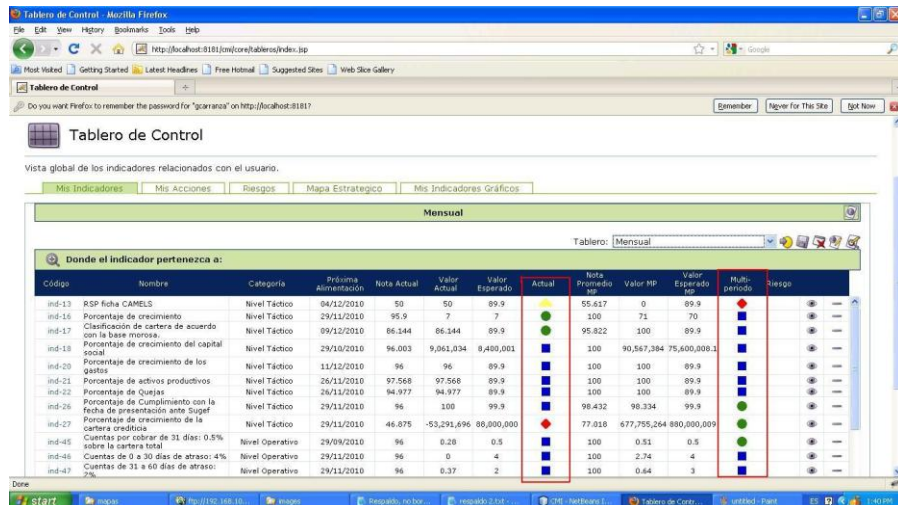
- Instalación de la herramienta necesaria para el uso de la aplicación (Cristal Reports).
- Ejecución del proceso de bug verification
- Registro en la herramienta Bugzilla de todos los defectos encontrados en la aplicación durante los ciclos de pruebas.

CMI

- Este es un proyecto bastante antiguo de esta empresa que a lo largo de los años ha pasado por actualizaciones y mejoras constantes. Se desea implementar una serie de mejoras a esta aplicación y se me encomendó la tarea de realizar estas mejoras.

Tareas:

- Implementar los cambios en la forma en que son desplegadas las colorimetrías, en este caso se debía utilizar el lenguaje *HTML* en conjunto con la tecnología *JSP*, además se utilizó también la herramienta *PhotoShop*.



The screenshot shows a web browser window displaying a dashboard. The main content is a table with the following columns: Código, Nombre, Categoría, Próxima Actualización, Nota Actual, Valor Actual, Valor Esperado, Actual, Nota Promedio, Valor MP, Valor Esperado MP, Multi-periodo, and Acción. The 'Actual' column contains colored circles (green, yellow, red) representing different status levels. The 'Multi-periodo' column contains colored squares (green, yellow, red) representing different status levels. The table lists various indicators such as 'RSP Riba CAMELS', 'Porcentaje de crecimiento', 'Clasificación de cartera de acuerdo con la base morosa', etc.

Ilustración 3: Cambios en la Colorimetría

- Implementar cambios en la forma de desplegar de los Gráficos de Indicadores, utilizando las mismas tecnologías.

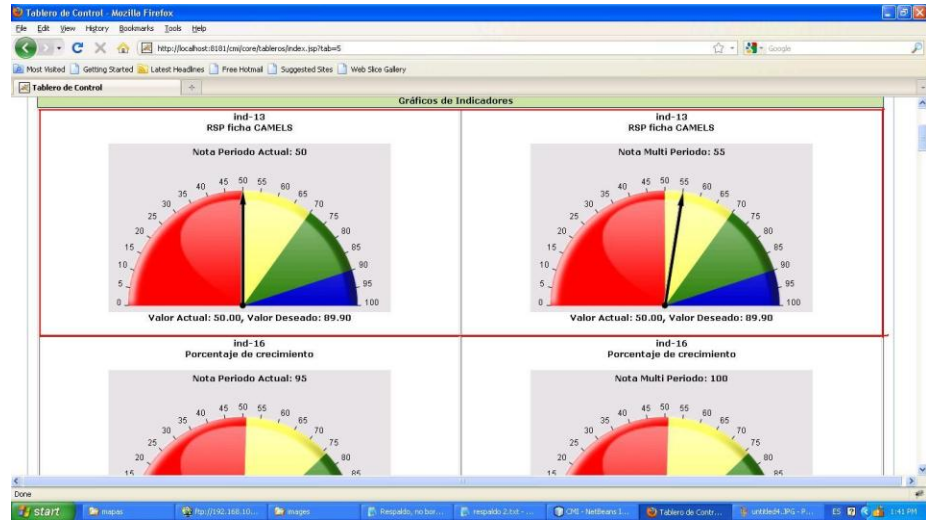


Ilustración 4: Cambios en los Indicadores Gráficos

- Implementar un filtro para el modulo de indicadores, con la idea de mejorar la usabilidad para el usuario, debido a que la gran cantidad de gráficos tornaba un poco lento la generación y despliegue de los mismos por lo que se decidió implementar este filtro para filtrar los indicadores gráficos en gráficos anuales, mensuales, trimestrales, etc.

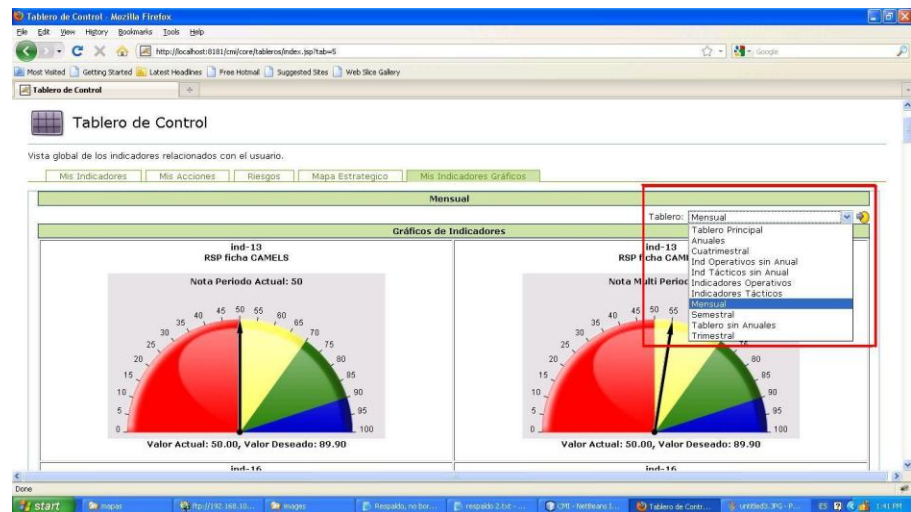


Ilustración 5: Filtro para los Indicadores Gráficos

- Implementar el modulo de Mapa estratégico.

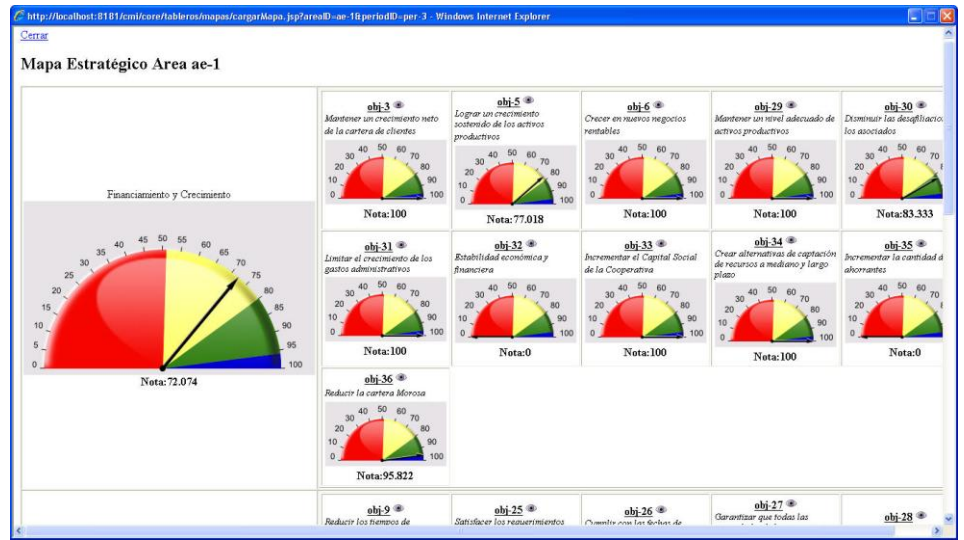


Ilustración 6: Mapa estratégico

- Implementar pantalla para desplegar los indicadores de cada objetivo

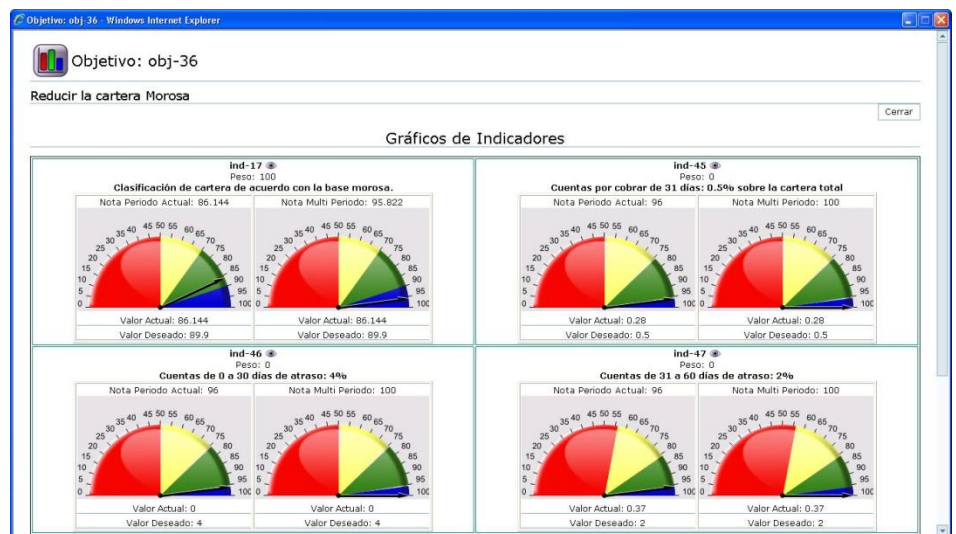


Ilustración 7: Indicadores de cada uno de los objetivos del Mapa Estratégico

- Desplegar los detalles de cada Indicador.



Ilustración 8: Detalle los Indicadores

- Implementar la funcionalidad de imprimir o exportar reporte (en proceso).

Productos de los ciclos de pruebas

Como productos de los ciclos de pruebas el departamento de QA hace entrega de 2 documentaciones principalmente; la primera es el reporte en si mismo efectuado en la herramienta Jira de todos los defectos encontrados a lo largo del ciclo de pruebas, y la segunda es el documento de “Test Result” ó resultados de las pruebas entregado directamente al PM de el proyecto implicado. Para ambos informes se utiliza un formato en específico y una información detallada que se desea documentar, a continuación un resumen de la información que abarca ambas documentaciones:

Reporte de bugs en Jira

En la herramienta Jira se deben reportar todos los defectos (bugs) encontrados a lo largo del ciclo de vida del proyecto. El reporte de estos bugs debe efectuarse de tal manera de que toda la información relevante al mismo quede debidamente documentada para diversos propósitos, esta información abarca básicamente lo siguiente:

Tabla 11: Información reportada en Jira

| | |
|-------------------|---|
| ID del Bug | Asignado automáticamente por Jira |
| ”Summary” | Resumen descriptivo |
| Description | Descripción detallada de en qué consiste el defecto encontrado, pasos a seguir para reproducir el error, comportamiento esperado y comportamiento encontrado. |
| Bug Type | Tipo de bug, ya sea defecto, error o mejora |
| Severity | Severidad del defecto localizado, que tanto afecta a las funcionalidades de la aplicación. |
| Component/s | Componente ó componentes que se ven afectados por el bug |
| Affects Version/s | Versión ó versiones de la aplicación en las cuales es localizado el defecto |
| Priority | Prioridad que tiene el bug localizado, ya sea urgente, critico, alta, media, baja o trivial. |
| Assignee | Desarrollador al cual le será asignado el defecto. |

Adicionalmente se puede reportar también datos tales como: URL, descripción del entorno, archivos adjuntos, tiempo estimado de corrección del defecto.

Test Result

Para la generación de este documento se utiliza, como ya fue mencionado anteriormente, la plantilla existente en Avantek, dicha plantilla contiene la información que necesitan tanto el PM como el QAL. Este documento básicamente pretende abarcar lo siguiente:

- Test Plan utilizado en el ciclo de pruebas.
- Build utilizado en el ciclo de pruebas.
- Fecha de inicio y finalización del ciclo de pruebas y el tiempo utilizado.
- Total de casos de prueba (ejecutados, fallidos, no ejecutados)
- Estado del Proyecto al iniciar el ciclo de pruebas (bugs abiertos, cerrados, verificados, etc.).
- Estado del Proyecto al finalizar el ciclo de pruebas.
- Severidad y prioridad de los defectos encontrados.

Conclusiones

A lo largo de los años Avantek Software ha incursionado en el área del Aseguramiento de la calidad del software, ha desarrollado sus propios procesos, métricas, herramientas y sistemas, los cuales han evolucionado hacia la obtención de altos niveles de calidad de todos sus productos; pero al mismo tiempo se han dado cuenta de que es un proceso de aprendizaje que debe mantenerse siempre al día con las mejores prácticas y tecnologías por lo cual siguen apostando por el desarrollo de esta rama como uno de sus principales mecanismos para su evolución y crecimiento.

Las tareas del departamento de QA son de vital importancia y de gran responsabilidad, ya que somos la última línea entre el producto desarrollado y el cliente, cualquier defecto o fallo encontrado posteriormente será responsabilidad del departamento de QA ya que somos los encargados de su temprana detección y reporte. Además, esta última línea es la que determina el nivel de calidad del producto que se le hará entrega al cliente y esto, al final, es lo que podría determinar el éxito o fracaso de un producto y la pérdida o ganancia de un cliente para la empresa y futuros proyectos por parte del mismo.

El personal de QA debe de ser personas comunicativas y emprendedoras, que estén siempre pendiente de las características de los proyectos y la comunicación con los diferentes equipos de trabajo, además, deben de ser personas auto-retadoras, que siempre estén en busca de nuevas formas o mejoras en la ejecuciones de ciclos de prueba, técnicas de estimación, métricas, etc. Con este fin la empresa realiza actividades y capacitaciones que ayudan a asimilar las experiencias vividas por los miembros del departamento en cada uno de los proyectos, esto facilita aun más el mejoramiento y preparación del equipo de trabajo.

Glosario

Bug: (= fallo) Una manifestación de un error en el software.

Error: la acción humana que produce un resultado incorrecto.

Resultados Esperados: (= Predicción ó Salidas Esperadas) El comportamiento previsto por la especificación de un objeto en condiciones especiales.

Test Plan: un acta del proceso de planificación de las pruebas que detalla el grado de independencia del probador, el entorno de prueba, las técnicas de caso de prueba de diseño y técnicas de prueba de medición a utilizar, y la justificación de su elección.

Test Case: Un conjunto de pasos, predicciones de ejecución y resultados esperados desarrollados para un objetivo particular, como para ejercer un escenario de programa en particular o para verificar el cumplimiento de un requisito específico.

Test Suite: Una colección de uno o más casos de prueba para el software bajo prueba.

Pasos: (= escenario de prueba, steps) Una secuencia de acciones ejecutables desde un punto de entrada a un punto de salida.

QA: Aseguramiento de la calidad, o QA (Quality Assurance) para abreviar, es el monitoreo y evaluación sistemáticos de los diversos aspectos de un proyecto o servicio para maximizar la probabilidad de que las normas mínimas de calidad están siendo alcanzados por el proceso de producción.

Testing: El proceso de prueba de software para comprobar si se cumplen los requisitos especificados y para detectar errores.

Release: El release de un software es la distribución del mismo, su documentación y materiales de soporte.

JSP: Es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

HTML: Siglas de *HyperText Markup Language* (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web.

PM: Project Manager.

QAL: Quality Assurance Lead.

Anexos

1. Para instalar Selenium IDE e iniciar con ejemplos básicos podemos utilizar el siguiente enlace:

[Installation and basic tutorial of Selenium IDE.docx](#)

2. Para configurar Selenium RC en Eclipse podemos utilizar el siguiente enlace:

[Configuring Selenium RC in Eclipse.docx](#)

3. Ejemplo básico de un caso de prueba automatizado utilizando Selenium RC en Eclipse:

[Basic Automated Test Case in Eclipse.docx](#)

4. Test cases desarrollados

[CV_test_plan-2011-04-01.doc](#)

5. Cronograma

[Cronograma.mpp](#)

Referencias

1. **TestLink** “*TestLink, the Web-based Test management and software testing execution tool*”
[Online] Available in:
<http://blog.testlink.org/> [22/02/2011]
<http://www.teamst.org/> [22/02/2011]
2. **Carrier Path Avantek Software (QA)**
[Career Path Avantek 2011 \(QA\).pdf](#)
3. **Selenium** “Selenium is a suite of tools specifically for testing web applications.”
<http://seleniumhq.org/> [22/02/2011]
[Tutorial Selenium RC](#)
4. **Eclipse**
<http://www.eclipse.org/>
5. **Cámara de Comercio:**
<http://camaradecomerciozn.com/sitio/>
6. **Krome:**
<http://www.kromepotos.com/>