Proyecto logO

Informe Final de Verificación

Version 1.4

Historia de revisiones

Fecha	Versión	Descripción	Autor
24/11/2013	1.0	Informe Final de Verificación	Nicolas Ramponi
25/11/2013	1.1	Revisión SQA	Ezequiel Jardim
25/11/2013	1.2	Informe Final de Verificación Estadísticas	Nicolas Ramponi
25/11/2013	1.3	Revisión SQA	Ezequiel Jardim
02/12/2013	1.4	Actualización de estadísticas	Nicolas Ramponi

Índice

1.1.1.1. Dado que este proyecto fue realizado utilizando la técnica SCRUM, se dejó afuera la verificación de documentos, dejando lugar únicamente a la verificación de calidad realizada por parte del Responsable de Calidad Ezequiel Jardim.	4
Versión 1.0	4
Pruebas Unitarias.....	4
Pruebas de Integración.....	4
Pruebas del Sistema	4
Versión 2.0	5
Pruebas Unitarias.....	5
Pruebas de Integración.....	5
Pruebas del Sistema	5
Versión 3.0	5
Pruebas Unitarias.....	5
Pruebas de Integración.....	5
Pruebas del Sistema	6
Versión 4.0	6
Pruebas Unitarias.....	6
Pruebas de Integración.....	6
Pruebas del Sistema	6
Versión 5.0	6
Pruebas Unitarias.....	6
Pruebas de Integración.....	7
Pruebas del Sistema	7

Resumen de la verificación

Planificado vs. Realizado

En esta parte, se describirá (como se hizo en las distintas entregas del plan de Verificación de iteración y de la evaluación de la verificación) lo que se logró comparado con lo que se había planificado para cada sprint.

Sprint 1:

Para este Sprint, se había planificado realizar verificación unitaria usando revisión y testeo automático de código en parejas, asignando a cada desarrollador una pareja que se ocupe de leer el código y realizar los test automáticos de su parte y vice versa. Esto no funcionó ya que se mostró complicado de implementar para los desarrolladores, que en su mayoría preferían trabajar solos, y se terminó dejando atrás el concepto para la segunda semana de este sprint para comenzar a realizar código y verificación personal.

Para la realización de test con código, se pidió al equipo de utilizar 3 horas por semana a terminarlo, pero esto no fue suficiente, ya que hubo una curva de aprendizaje para hacer test que no se había alcanzado a superar aún, por lo tanto el cubrimiento planificado de 90 % no se logró para el tiempo de la demo.

También se planificó que el equipo de verificación realizara pruebas de sistema sobre lo implementado, pero lamentablemente no se pudo lograr mucho testeo por falta de funcionalidad en lo que se implementó durante este sprint. El testeo que se realizó se hizo el último día antes de la demo, por lo que hubo muy poco tiempo para corregirlos.

Sprint 2 :

Para este sprint, se había planificado empezar a utilizar herramientas de reporte y manejo de bugs que fueran más efectivas que el uso de una tabla Excel, pero el estudio y puesta en funcionamiento de estas herramientas se mostró muy costoso y se dejó de lado. También se dejó de lado el uso de herramientas para generación de test más amigables y la lectura de código. Esto último se dejó de lado por lo difícil que resultó hacerle el seguimiento y se entendió por parte de los desarrolladores y el Responsable de Verificación que este método era poco efectivo.

La realización de test de sistema por parte del equipo de verificación fue más holgura ya que se tuvo gran parte del último fin de semana para organizar y correr los test. También se organizó la realización de modelos de casos de prueba a lo largo del Sprint para evitar sobrecarga al final, lo que fue de mucha ayuda.

Aún no se lograron testear todas las funcionalidades que se querían testear por falta de tiempo.

Sprint 3:

Para este sprint, se planificó la realización del testeo del sistema de manera más acorde con la capacidad del equipo, y junto con la implementación de un nuevo sistema y organización de reporte y modelo de pruebas, se logró dar feedback a los desarrolladores sobre los errores encontrados mucho más temprano que antes (incluso en la primera semana) permitiendo que muchos de estos fueran solucionados antes de la demo, pero aún faltaron historias por verificar y por eso no se llegó a completar la cantidad de puntos estimados en la demo al cliente.

Sprint 4:

En este sprint finalmente se pudo completar casi en su totalidad la verificación y modelado planificado al principio del sprint, dejando poco espacio para historias sin verificar. Se llegó al demo con más del 90% de las historias validadas por el equipo de Verificación.

Sprint 5:

Para el equipo de Verificación, este sprint sirvió únicamente para testear con mayor profundidad todo el producto, haciendo las pruebas que no se pudieron hacer en los primeros Sprints y revisando el correcto funcionamiento de las funcionalidades que ya estaban validadas anteriormente. También se tuvo que testear y validar una única historia que había quedado sin validar en el sprint anterior.

El objetivo de volver a testear lo que había quedado con testeo poco exhaustivo se logró a tiempo en gran parte y se logró recuperar muchos errores que existían en el producto.

Cantidad de errores encontrados

Documentos

Dado que este proyecto fue realizado utilizando la técnica SCRUM, se dejó afuera la verificación de documentos, dejando lugar únicamente a la verificación de calidad realizada por parte del Responsable de Calidad Ezequiel Jardim.

Versiones

Versión 1.0

Pruebas Unitarias

Gráfica y Tabla de Datos

Al haber utilizado un método ágil, se dejó de lado la documentación para la verificación unitaria, dejando a los desarrolladores la agilidad de encontrar y resolver los errores en el momento.

Cantidad de verificadores: Equipo de desarrollo, 11 personas

Cantidad de horas invertidas: 76

Cobertura de Código: 0%

Pruebas de Integración

No se hicieron pruebas de integración. La verificación de la integración quedó bajo la responsabilidad de los desarrolladores.

Pruebas del Sistema

Gráfica y Tabla de Datos

Ver tabla de Reporte y Modelo de casos de prueba del Sprint 1

Cantidad de verificadores: 2

Cantidad de horas invertidas: 71,5

Versión 2.0

Pruebas Unitarias

Gráfica y Tabla de Datos

Al haber utilizado un método ágil, se dejó de lado la documentación para la verificación unitaria, dejando a los desarrolladores la agilidad de encontrar y resolver los errores en el momento.

Cantidad de verificadores: Equipo de desarrollo, 11 personas

Cantidad de horas invertidas: 68,5

Cobertura de Código: 85.64 %

Pruebas de Integración

No se hicieron pruebas de integración. La verificación de la integración quedó bajo la responsabilidad de los desarrolladores.

Pruebas del Sistema

Gráfica y Tabla de Datos

Ver tabla de Reporte y Modelo de casos de prueba del Sprint 2

Cantidad de verificadores: 2

Cantidad de horas invertidas: 56,5

Versión 3.0

Pruebas Unitarias

Gráfica y Tabla de Datos

Al haber utilizado un método ágil, se dejó de lado la documentación para la verificación unitaria, dejando a los desarrolladores la agilidad de encontrar y resolver los errores en el momento.

Cantidad de verificadores: Equipo de desarrollo, 11 personas

Cantidad de horas invertidas: 70

Cobertura de Código: 91,78 %

Pruebas de Integración

No se hicieron pruebas de integración. La verificación de la integración quedó bajo la responsabilidad de los desarrolladores. En este sprint, el responsable de verificación se empezó a seguir de cerca la integración, ya que el producto tenía que estar en condiciones de ser testeado antes de subirse a la rama Testing.

Pruebas del Sistema

Gráfica y Tabla de Datos

Ver tabla de Reporte y Modelo de casos de prueba del Sprint 3

Cantidad de verificadores: 2

Cantidad de horas invertidas: 74,5

Versión 4.0

Pruebas Unitarias

Gráfica y Tabla de Datos

Al haber utilizado un método ágil, se dejó de lado la documentación para la verificación unitaria, dejando a los desarrolladores la agilidad de encontrar y resolver los errores en el momento.

Cantidad de verificadores: Equipo de desarrollo, 11 personas

Cantidad de horas invertidas: 75,25

Cobertura de Código: 89,57 %

Pruebas de Integración

No se hicieron pruebas de integración. La verificación de la integración quedó bajo la responsabilidad de los desarrolladores. El responsable de verificación siguió de cerca la integración, ya que el producto tiene que estar en condiciones de ser testeado antes de subirse a la rama Testing.

Pruebas del Sistema

Gráfica y Tabla de Datos

Ver tabla de Reporte y Modelo de casos de prueba del Sprint 4

Cantidad de verificadores: 2

Cantidad de horas invertidas: 50,75

Versión 5.0

Pruebas Unitarias

Gráfica y Tabla de Datos

Al haber utilizado un método ágil, se dejó de lado la documentación para la verificación unitaria, dejando a los desarrolladores la agilidad de encontrar y resolver los errores en el momento.

Cantidad de verificadores: Equipo de desarrollo, 11 personas

Cantidad de horas invertidas: 3

Cobertura de Código: 92,12 %

Pruebas de Integración

No se hicieron pruebas de integración. La verificación de la integración quedó bajo la responsabilidad de los desarrolladores. El responsable de verificación siguió de cerca la integración, ya que el producto tiene que estar en condiciones de ser testeado antes de subirse a la rama Testing.

Pruebas del Sistema

Gráfica y Tabla de Datos

Ver tabla de Reporte y Modelo de casos de prueba del Sprint 4

Cantidad de verificadores: 2

Cantidad de horas invertidas: 33,5

Casos de Uso (Historias)

Historia 1.1.1/2.1.1 : Como superAdmin puedo crear usuarios y definir permisos (para futuros sign up).

Cantidad de Casos de Prueba: 14

Cantidad de errores detectados: 7

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 1.1.2/2.1.2 : Como superAdmin, quiero dar permisos a otros admin sobre proyectos.

Cantidad de Casos de Prueba: 13

Cantidad de errores detectados: 2

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 1.2.1/2.2.1 : Como usuario, quiero crear una nueva conexión. (sign in en la herramienta)

Cantidad de Casos de Prueba: 6

Cantidad de errores detectados: 6

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 1.2.2/2.2.2 : Como usuario, quiero ver el listado de herramientas conectadas.

Cantidad de Casos de Prueba: 5
Cantidad de errores detectados: 7
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 1.2.3/2.2.3 : Como usuario, quiero editar los proyectos de una conexión.

Cantidad de Casos de Prueba: 9
Cantidad de errores detectados: 3
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 1.2.4/2.2.4/3.2.1 : Como usuario, quiero ingresar horas en una tarea. (con sync)

Cantidad de Casos de Prueba: 33
Cantidad de errores detectados: 9
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 1.2.6 : Como usuario, quiero ingresar con Open Id. (con seguridad)

Cantidad de Casos de Prueba: 6
Cantidad de errores detectados: 4
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 1.2.7/2.2.6 : Como usuario, quiero ver las horas ingresadas por día y navegar entre ellas.

Cantidad de Casos de Prueba: 4
Cantidad de errores detectados: 3
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 1.2.8/2.2.7 : Como usuario, quiero poder ver las horas que me faltan ingresar en el día.

Cantidad de Casos de Prueba: 4
Cantidad de errores detectados: 1
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 1.4.1/2.4.1 : Como usuario, quiero crear una nueva conexión. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 1.4.2/2.4.2 : Como usuario, quiero ver el listado de herramientas conectadas. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 1.4.3/2.4.3 : Como usuario, quiero editar los proyectos de una conexión. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 1.4.4 : Como usuario, quiero ingresar con Open Id. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 2.1.3/3.1.1 : Como admin o superAdmin, puedo definir categorías por proyecto (no borrar usadas)

Cantidad de Casos de Prueba: 13
Cantidad de errores detectados: 2
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Historia 2.1.4 : Como usuario, quiero reporte por usuario: horas x proyecto (semana/sprint)

Cantidad de Casos de Prueba: 5
Cantidad de errores detectados: 0
Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.1.5 : Como usuario, quiero reporte por usuario: desviación en horas de tareas asignadas al usuario

Cantidad de Casos de Prueba: 5

Cantidad de errores detectados: 2

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.1.6 : Como admin, puedo ver los usuarios de los proyectos de los cuales hago reportes.

Cantidad de Casos de Prueba: 0

Cantidad de errores detectados: 0

Cantidad de verificadores: 0

Cantidad de horas invertidas: No determinado

Historia 2.2.8/3.2.2/4.2.1 : Como usuario, quiero ingresar Start y Stop y fijar límite de tiempo.

Cantidad de Casos de Prueba: 19

Cantidad de errores detectados: 10

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.2.9 : Como usuario, quiero hacer sign in con OpenID con cuentas de google y mooveit.

Cantidad de Casos de Prueba: 7

Cantidad de errores detectados: 2

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.2.11 : Como usuario, quiero modificar las credenciales de una herramienta.

Cantidad de Casos de Prueba: 12

Cantidad de errores detectados: 2

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.2.12/3.2.4 : Como usuario, quiero poder cambiar el estado de las tareas.

Cantidad de Casos de Prueba: 8

Cantidad de errores detectados: 6

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.2.13/3.2.5: Como usuario, quiero ver todos los proyectos sincronizados. (Filtrar)

Cantidad de Casos de Prueba: 14

Cantidad de errores detectados: 2

Cantidad de verificadores: 2

Cantidad de horas invertidas: No determinado

Historia 2.4.5/3.4.1 : Como usuario, quiero ingresar horas en una tarea. (adaptado al móvil)

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados: 0

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 2.4.6/3.4.2/4.4.1 : Como usuario, quiero ingresar Start y Stop y fijar límite de tiempo. (adaptado al móvil)

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados: 2

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 2.4.7 : Como usuario, quiero ver las horas ingresadas por día y navegar entre ellas. (adaptado al móvil)

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados: 0

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

**Historia 2.4.8 : Como usuario, quiero modificar las credenciales de una herramienta.
(adaptado al móvil)**

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.2 : Como usuario, quiero reporte por usuario: horas x categoría (sprint/proyecto)

Cantidad de Casos de Prueba: 6
Cantidad de errores detectados: 2
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.3 : Como usuario, quiero reporte de sprints: horas x stories y tareas

Cantidad de Casos de Prueba: 13
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.4 : Como usuario, quiero reporte de sprints: burndown chart

Cantidad de Casos de Prueba: 8
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.5 : Como usuario, quiero reporte por semana: horas x usuario

Cantidad de Casos de Prueba: 7
Cantidad de errores detectados: 2
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.6/4.1.1 : Como usuario, quiero forzar sincronización en general, y en cada recurso por separado. (por cada recurso) (SEPARAR)

Cantidad de Casos de Prueba: 6
Cantidad de errores detectados: 2
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.7/4.1.2 : Como admin, quiero un CRUD de reportes pre-armados.

Cantidad de Casos de Prueba: 17
Cantidad de errores detectados: 4
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.1.8 : Como admin, quiero desde cada reporte poder exportarlo a PDF

Cantidad de Casos de Prueba: 3
Cantidad de errores detectados: 2
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.2.8 : Como usuario, quiero ver todos los sprints asociados a un proyecto.

Cantidad de Casos de Prueba: 10
Cantidad de errores detectados: 2
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.2.9 : Como usuario, quiero ver los sprints y poder filtrarlos.

Cantidad de Casos de Prueba: 12
Cantidad de errores detectados: 3
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.2.10 : Como usuario, quiero ver todas las stories asociadas a un sprint.

Cantidad de Casos de Prueba: 10
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.2.11 : Como usuario, quiero ver todas las stories y poder filtrarlas.

Cantidad de Casos de Prueba: 16
Cantidad de errores detectados: 5
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

**Historia 3.4.3 : Como usuario, quiero poder ver las horas que me faltan ingresar en el día.
(adaptado al móvil)**

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 3.4.4 : Como usuario, quiero poder cambiar el estado de las tareas. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.3 : Como usuario, quiero reporte de sprints: horas x usuario

Cantidad de Casos de Prueba: 7
Cantidad de errores detectados: 3
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.4 : Como usuario, quiero reporte de sprints: horas x categoría

Cantidad de Casos de Prueba: 4

Cantidad de errores detectados: 3
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.5 : Sincronización automática

Cantidad de Casos de Prueba: 2
Cantidad de errores detectados: 2
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.6 : Como usuario, quiero hacer sign in

Cantidad de Casos de Prueba: 9
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.7 : Como usuario, quiero open ID

Cantidad de Casos de Prueba: 6
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.8 : Como usuario, quiero reporte de sprint: desviación

Cantidad de Casos de Prueba: 6
Cantidad de errores detectados: 6
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.1.9/5.1.1 : Como admin, quiero desde cualquier reporte poder agendar para posterior envío.

Cantidad de Casos de Prueba: 8
Cantidad de errores detectados: 0
Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.2.2 : Como usuario, quiero hacer sign in.

Cantidad de Casos de Prueba: 13

Cantidad de errores detectados: 10

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.2.5 : Como usuario, quiero ver todas las tareas asociadas a un story.

Cantidad de Casos de Prueba: 19

Cantidad de errores detectados: 2

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.2.6 : Como usuario, quiero ver todas las tareas y poder filtrarlas.

Cantidad de Casos de Prueba: 22

Cantidad de errores detectados: 2

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.2.7 : Como usuario, quiero hacer sign up.

Cantidad de Casos de Prueba: 11

Cantidad de errores detectados: 20

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.2.8 : Como usuario, quiero poder modificar mi perfil. En particular: las horas de trabajo diarias.

Cantidad de Casos de Prueba: 13

Cantidad de errores detectados: 0

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.4.2 : Usabilidad muy buena, fácil.

Cantidad de Casos de Prueba: 0

Cantidad de errores detectados: 102

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.4.3 : Buena visualización en iOS y Android.

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados: 0

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.4.4 : Como usuario, quiero ver todos los proyectos sincronizados. (Filtrar) (adaptado al móvil)

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados:

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.4.5 : Como usuario, quiero ver todos los sprints asociados a un proyecto. (adaptado al móvil)

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados: 1

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.4.6 : Como usuario, quiero ver los sprints y poder filtrarlos. (adaptado al móvil)

Cantidad de Casos de Prueba: 1

Cantidad de errores detectados: 1

Cantidad de verificadores: 1

Cantidad de horas invertidas: No determinado

Historia 4.4.7 : Como usuario, quiero ver todas las stories asociadas a un sprint. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.4.8 : Como usuario, quiero ver todas las stories y poder filtrarlas. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.4.9 : Como usuario, quiero ver todas las tareas asociadas a un story. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.4.10 : Como usuario, quiero ver todas las tareas y poder filtrarlas. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.4.11 : Como usuario, quiero forzar sincronización en general, y en cada recurso por separado. (por cada recurso) (SEPARAR) (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 1
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.4.12 : Como usuario, quiero hacer sign up. (adaptado al móvil)

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Historia 4.4.13 : Como usuario, quiero poder modificar mi perfil. En particular: las horas de trabajo diarias. (adaptado al móvil) SACAR

Cantidad de Casos de Prueba: 1
Cantidad de errores detectados: 0
Cantidad de verificadores: 1
Cantidad de horas invertidas: No determinado

Ademas de estas historias, se tienen Modelos y errores reportados que no corresponden a ninguna historia en particular :

Cantidad de Casos de Prueba: 12
Cantidad de errores detectados: 102 (usabilidad)
Cantidad de verificadores: 2
Cantidad de horas invertidas: No determinado

Evaluación

Este segmento se va a utilizar para resumir lo que se fue reportando al final de cada Sprint, aportando en cada uno, una evaluación retrospectiva comparando lo que se hizo con lo que se pudo haber hecho para mejorar:

Sprint 1:

Este Sprint sufrió mucho de la subestimación que se tuvo frente a la generación de test automáticos por parte de los desarrolladores. Las medidas necesarias no fueron tomadas por parte del responsable de verificación para asegurar que este tema estuviera entendido y manejado para cuando el proyecto entrara en su primer sprint. Esto llevó a que se corrieran pruebas de sistema al mismo tiempo que se hacían las pruebas unitarias, y todo esto cerca del final del sprint.

La forma de mejorar este Sprint hubiese sido que se pusiera como objetivo al equipo de técnicos realizar un test unitario que cubriera la mayor parte del código de los prototipos que se generaron en la fase inicial, de esa forma permitiendo que no hubiera tanta sorpresa al entrar en producción. Este fue el peor inconveniente con que se encontró la verificación de la aplicación.

Sprint 2:

Luego de superado el problema del testing unitario en gran parte durante la primera semana de este sprint, se entendió que el testing de sistema no se tenía que hacer mientras se realizaban los test unitarios ya que estos podían revelar errores que fueran más fáciles de solucionar que si se llevaran a nivel de sistema.

Esta manera de llevar a cabo el testing funcional llevó a que el testing se realizara nuevamente solo a pocos días del demo, ya que nuevamente falló la planificación para acomodar este tipo de testing dentro del ciclo de desarrollo. De todas formas, por primera vez se tuvo la chance de dar feedback a los desarrolladores lo suficientemente temprano como para permitir corregir muchos de estos errores. La forma de dar feedback (a través de un archivo Excel bastante poco intuitivo y lento para completar) fue poco efectiva, y fue un defecto de la organización que se notó muy tarde en el sprint.

Existió otro problema que concernía el uso de la Branch Testing. La misma era utilizada por ciertos desarrolladores por falta de organización en el momento de planear el uso de las branches, por lo que el equipo de verificación se encontraba a veces inseguro de lo que estaba verificando.

La realización de los modelos fue un problema en este Sprint, ya que el responsable de verificación se asignó la realización de todos los modelos, dejando al asistente modelar y realizar las pruebas de interfaz. Esta metodología se mostró muy poco efectiva, ya que se notó que al testear una parte de la funcionalidad de una historia, se podía al mismo tiempo testear otra funcionalidad, (que en este caso quedaría delegada a otro verificador, que también tendría que conocer las especificaciones de una historia) de forma que se perdía mucho tiempo y precisión de forma innecesaria.

Este último problema se podría haber encarado diferente desde un principio, pero fue algo que se aprendió a base de prueba y error. Para la comunicación con los desarrolladores al momento de dar feedback, se podría haber investigado herramientas que facilitaran esta comunicación en la fase inicial, de forma que no hubiera una curva de aprendizaje, que en este caso tomó dos sprints completos.

Sprint 3 :

Se inició este sprint utilizando un nuevo formato para generar modelos y reportes, permitiendo que se modelara de una forma menos estricta y más ágil, sin perder la especificidad. El reporte nuevo permitió que, utilizando menos campos, se pudiera llenar con errores encontrados más rápido.

Por primera vez se generó la necesidad de reportar errores por email, ya que el testeo se hizo también por primera vez, a lo largo del sprint y no en los últimos días, en donde todo el equipo se encontraba en un mismo sitio.

Los mails tuvieron un formato especial que permitió a los verificadores recibir feedback de los desarrolladores y vice versa.

El Excel con el reporte también se utilizó como una herramienta de comunicación, permitiendo a los desarrolladores modificar el estado del bug según si este estaba en proceso de ser arreglado o arreglado y disponible en la branch accesible al equipo de verificación.

Otra cosa importante es que se le dio al Responsable de Verificación la responsabilidad de manejar la Branch Testing, donde estaba todo el código listo para verificar. Al tener único acceso a esta branch, no existían conflictos entre lo que se estaba verificando y lo que se estaba implementado, permitiendo apartar la generación de ciertos errores innecesarios.

Se decidió también que la subida de código a la Branch Testing se haría una vez cada 48 horas por el tiempo que tomaba este proceso, en donde se tomaban y analizaban todas las estadísticas de cubrimiento de código y funcionamiento de test unitarios.

Hubo pocas cosas por mejorar en este Sprint. El único problema que se identificó es que en ciertas ocasiones, la subida de código a la Branch Testing una vez cada 48 horas era a veces muy lento como para dar feedback rápido a los desarrolladores.

Sprint 4 :

En este sprint, se terminó por realizar el testing de una manera muy satisfactoria, dejando pocos comentarios por hacer.

Una de las mejoras más importantes fue el compromiso por parte del equipo de desarrollo de entregar material para testear lo suficientemente temprano como para que se pudiera hacer un testing y una eventual recuperación de un bug encontrado. Esto, y el hecho que se empezó a hacer en casos de urgencia testeo en la Branch Desarrollo (utilizada por el equipo de desarrollo únicamente), permitió que se detectaran y corrigieran los errores lo suficientemente rápido como para llegar al demo con la gran mayoría de las historias validadas por el equipo de verificación.

Sprint 5 :

El testeo que se realizó en este sprint fue básicamente para terminar con el testeo de historias pobremente testeadas en el pasado. El testeo se realizó sin problemas, ya que la metodología de trabajo ya estaba bien establecida y no había que esperar a que se terminaran las historias.

Informe final de las métricas de Pruebas Cubiertas

Para esta sección, se cuenta únicamente con el cubrimiento de código realizado por el testing utilizando test automáticos, cuyo número es el siguiente:

Cubrimiento del Backend: 95,51 %