

Proyecto Final – *Plan de Testing*

Hay equipo

- Red social deportiva -

Curso: 5K2

-

Año: 2019

Docentes:

- **Ingeniero Zohil**, Julio César Nelson (Adjunto)
- **Ingeniero Liberatori**, Marcelo Sadí (JTP)
- **Ingeniera Jaime**, María Natalia (JTP)

Grupo Nº 1:

- ☐ **Alcubilla**, Mauricio – Legajo: 46231
- ☐ **Angonoa**, Franco – Legajo: 63390
- ☐ **Monetto**, Ignacio – Legajo: 60418
- ☐ **Pozzo**, Nicolás – Legajo: 57893
- ☐ **Retamozo**, Agustín Julián – Legajo: 53360
- ☐ **Rugiero**, Franco – Legajo: 58232

HISTORIA DEL DOCUMENTO

Fecha	Versión	Comentarios	Autor
23 Abril 2019	0.1	Versión inicial.	HayEquipo
15 Mayo 2019	0.2	Correcciones sobre el proyecto indicadas por el profesor tutor.	HayEquipo

TABLA DE CONTENIDOS

1	INTRODUCCIÓN	5
1.1	Propósito	5
1.2	Objetivos del Plan	5
1.3	Alcance del Testing	6
1.4	Criterios de Entrada	6
1.5	Criterios de Salida	7
1.6	Criterios de Suspensión y Retomar Actividad	7
2	Test del Sistema	8
2.1	Estrategia de Testing del Sistema	8
2.2	Pruebas en Operación Normal	8
2.2.1	Funcionalidad Grupo 1: Prueba Funcional	9
2.2.2	Funcionalidad Grupo 2: Prueba de Sistemas	9
2.2.3	Funcionalidad Grupo 3: Prueba de Integración	9
2.2.4	Funcionalidad Grupo 4: Pruebas de Aceptación	10
2.2.5	Funcionalidad Grupo 5: Prueba No Funcional	10
2.2.6	Funcionalidad Grupo 6: Prueba Estructural	11
2.2.7	Funcionalidad Grupo 7: Pruebas de Confirmación (Prueba relacionada con cambios)	11
2.2.8	Funcionalidad Grupo 8: Pruebas de Regresión (Prueba relacionada con cambios)	11
2.3	Criterios de Éxito/Fracaso de Pruebas	11
2.4	Entregables	12
3	Configuración del Test	13
3.1	Hardware	13
3.2	Software	13
3.3	Ambiente de Prueba	14

4	Tareas	15
4.1	Actividades	15
4.2	Responsabilidades	15
4.2.1	Responsabilidades del Grupo de Desarrollo	15
4.2.2	Responsabilidades del Grupo de Testing.....	15
4.2.3	Responsabilidades del Grupo	15
4.3	Planificación.....	16
	Glosario (Definiciones y Siglas)	17
	ANEXO	18
	<i>Planilla de Ejecución</i>	<i>18</i>
	<i>Planilla de Defectos.....</i>	<i>19</i>
	<i>Test Case.....</i>	<i>20</i>

1 INTRODUCCIÓN

1.1 Propósito

HayEquipo es una red social destinada al vínculo entre personas, que practican deportes a nivel amateur, y complejos deportivos, que ofrecen servicios en sus establecimientos, con el fin de poder armar grupos de personas que puedan practicar un deporte y reservar canchas en los distintos establecimientos deportivos registrados en esta red como así también en lugares públicos.

1.2 Objetivos del Plan

El Plan de Prueba o Plan de Testing del Sistema especifica los procesos de test y de verificación que se realizarán con el objeto de:

- Identificar defectos y fallas.
- Medir rendimiento.
- Evaluar la calidad
- Determinar el cumplimiento de los requerimientos.

Los objetivos de este plan son:

- Definir y detallar todas las tareas que se desarrollarán para probar el sistema.
- Definir el plan y la persona o grupo responsable de cada tarea.
- Definir las herramientas de prueba y el ambiente necesario a la conducción de las actividades de testing.
- Definir los ítems y funcionalidades que serán probados.
- Identificar los defectos en el software durante diferentes estados de comportamiento no deseados.

1.3 Alcance del Testing

El *Plan de Testing* del sistema es una especificación de alto nivel de los requerimientos funcionales y de calidad que serán probados, del ambiente y la estrategia de testing, de las responsabilidades y de los criterios de éxito.

El alcance del testing del sistema es probar la funcionalidad completa de la red social HayEquipo.

Los casos de prueba y los criterios de éxito serán derivados de este plan general y serán especificados en el documento de Especificaciones de Testing del Sistema.

Este documento describe los siguientes tópicos:

- Flujo de requerimientos (que no sean ambiguos, claros),
- Análisis (redacción, descripción de roles, identificación de actores casos de usos y sus objetivos),
- Diseño (descripción de trazos finos, trazos gruesos, diagramas de comunicación y de secuencia; sólo los más significativos),
- Implementación (pantallas principales y cursos normales de todas las pantallas),
- Prueba (pruebas de integración, prueba de sistema e identificación de mejoras y correcciones, pruebas de stress, pruebas de volumen)
- Despliegue (manual de instalación, manual de usuario y manual de procedimientos).

1.4 Criterios de Entrada

Para poder comenzar la fase de pruebas del sistema, se deben cumplir los siguientes criterios:

- Test unitarios realizados y completados para cada componente del sistema.
- Sistema completamente integrado.
- Todo el hardware está correctamente instalado, configurado y funcionando adecuadamente.
- La preparación de datos está lista.

- Calidad de Software: La calidad está constituida por
 - Atributos funcionales de calidad:
 - Funcionalidad: correctitud y completitud de los requisitos del usuario.
 - Atributos NO funcionales de calidad:
 - Fiabilidad: el sistema mantendrá su capacidad y funcionalidad a lo largo de un período de tiempo.
 - Usabilidad: fácil de usar, fácil de aprender, conforme a normas y uso intuitivo.

1.5 Criterios de Salida

- 100% de la ejecución de los test cases completados.
- 99% pasados satisfactoriamente de un total de 100%. (PASSED)
- 100% de los defectos de prioridad crítica resueltos.

1.6 Criterios de Suspensión y Retomar Actividad

Si alguna prueba no está cumpliendo el propósito para el cual fue creada esta podrá ser eliminada permanentemente del plan de testing. A su vez, si se descubre alguna falla en una funcionalidad del sistema, las pruebas que requieran de esa funcionalidad serán suspendidas hasta que el problema sea solventado.

2 Test del Sistema

El sistema por probar se define *desde* que se hacen las invitaciones para realizar una actividad deportiva determinada en un espacio público o privado, *hasta* la realización de la reserva incluyendo cuando se inicia y finaliza la actividad deportiva.

2.1 Estrategia de Testing del Sistema

Se debe planificar las pruebas necesarias en cada iteración, incluyendo las pruebas de unidad, integración y las pruebas de sistema. Las pruebas de unidad y de integración son necesarias dentro de la iteración, mientras que las pruebas de sistema son necesarias sólo al final de la iteración.

Se realizará Testing Funcional con macros en Word y en Excel para definir los pasos que se realizan para llevar a cabo una determinada prueba, comparando el resultado obtenido con el resultado esperado y en el caso de detectar un defecto se documenta para que el desarrollador proceda a realizar correcciones, y finalmente volver a realizar la prueba para verificar que fue resuelto el defecto. Luego se hace un seguimiento con reporte de defectos para saber la cantidad de defectos detectados y si fueron solucionadas en su totalidad.

2.2 Pruebas en Operación Normal

Pruebas Funcionales, Pruebas de Sistemas, Integración (tipo especial es Incremental de tipo ascendente o descendente; Integración no Incremental: se prueba cada módulo por separado y luego se integran todos de una vez), Pruebas de Aceptación, Prueba No Funcional, Prueba Estructural, Pruebas de Confirmación y Pruebas de Regresión (Se aplican a las pruebas funcionales, no funcionales y estructurales), pruebas de Stress (pruebas para observar el tiempo en que la aplicación comienza a fallar o es incapaz de responder a las peticiones. Son pruebas de carga o rendimiento, pero superando los límites esperados en el ambiente de producción y/o determinados en las pruebas) y pruebas de Volumen (son las pruebas en las que el sistema está sometido a grandes volúmenes de datos).

2.2.1 Funcionalidad Grupo 1: Prueba Funcional.

Basada en Casos de Uso: realización de grafos de Prueba a partir de los trazos finos de los Casos de Uso y a partir de los grafos se redactan los Casos de Prueba por el/los camino/s positivos y negativos que representan la funcionalidad más relevante del negocio.

2.2.2 Funcionalidad Grupo 2: Prueba de Sistemas.

Se prueba el comportamiento del sistema completo como está definido en el alcance del programa desarrollado.

Alcances:

- Gestionar la creación y ciclo de vida de partidos o encuentros deportivos, como también los estados que van transitando.
- Gestionar foro para cada encuentro.
- Gestionar información estadística y reportes.
- Gestionar los perfiles de usuario.
- Gestionar las reservas.
- Gestionar invitaciones por parte de los deportistas.
- Gestión de Equipos.
- Emitir lugares y disponibilidad de fechas y horas.
- Gestionar publicidades.

2.2.3 Funcionalidad Grupo 3: Prueba de Integración

Probar interfaces entre componentes, interacciones a diferentes partes del producto tales como el S.O., sistema de archivos, hardware o interfaces entre sistemas. Por ejemplo, probar el rendimiento que es una prueba no funcional específica, es parte a su vez de una prueba de integración.

2.2.4 Funcionalidad Grupo 4: Pruebas de Aceptación

Una vez que el sistema esté completado y terminado, y previamente se hayan realizado las pruebas de integración y de sistema, se realizan las pruebas de aceptación que son los usuarios finales y operadores del software funcionando.

2.2.5 Funcionalidad Grupo 5: Prueba No Funcional

- Interfaces de Usuarios: Compatibles, amigables, fáciles de entender, menús coherentes.
- Prueba de rendimiento o performance:
 - Capacidad de Almacenamiento (100Mb),
 - Capacidad de Procesamiento: 128 MB de RAM,
 - Capacidad de Ancho de Banda: 512 KBPS dedicados.
- Prueba de Seguridad: Niveles
 - Seguridad de Aplicación: Login de Usuarios, integridad de los datos.
- Prueba de Stress: Verificar el acceso concurrente de 100 personas al programa desde un browser y siga respondiendo con la misma performance.
- Prueba de Volumen: Verificar el acceso concurrente de 10 personas al programa desde un browser y siga respondiendo con la misma performance.
- Pruebas de Configuración: verificar que la aplicación puede ser accedida desde distintos browsers.
- Pruebas de Instalación: Probar que la aplicación puede verse desde Navegadores Google Chrome Versión 73.0.3683.103 (32 Bits) o superior; Navegador Firefox versión 59.0.2 o superior; Opera versión 51.0.2830.34 o superior; Navegador Microsoft Edge versión 44.17763.1.0 o superior.
- Usabilidad: Verificar que los menús se pueden acceder con facilidad y se ven correctamente.

2.2.6 Funcionalidad Grupo 6: Prueba Estructural

Se llevará a cabo en el caso de los casos de uso abstractos para probar el correcto funcionamiento del sistema.

Prueba de Caja Blanca: Se realiza una inspección minuciosa al código en busca de defectos y se dividen en cobertura de decisión (ciclos if, else, for, while, do while, for each, etc), cobertura de sentencias (campo de acciones e impresiones de datos), coberturas múltiples, cobertura de condición (probar por verdadero y por falso el valor contenido dentro de la decisión), cobertura de condición/decisión (probar por verdadero la condición y la decisión y viceversa).

2.2.7 Funcionalidad Grupo 7: Pruebas de Confirmación (Prueba relacionada con cambios)

Se aplicará cuando se detecte un defecto y este sea arreglado, para comprobar que el software debe ser probado de nuevo para confirmar que el defecto original ha sido retirado exitosamente.

2.2.8 Funcionalidad Grupo 8: Pruebas de Regresión (Prueba relacionada con cambios)

Son las pruebas repetidas de un programa ya probado, después de la modificación para descubrir cualquier defecto introducido luego de un cambio, sobre todo cuando se hace una actualización del sistema o de uno de sus componentes, al igual que al hacerse un cambio importante del sistema. Por ejemplo, versiones nuevas de los navegadores, o cambio de S.O.

2.3 Criterios de Éxito/Fracaso de Pruebas

Criterios de Éxito/Fracaso de Pruebas. Se ha considerado que una prueba fue exitosa si esta cumple con todos los parámetros definidos en el caso de prueba correspondiente. Los resultados que arroja la prueba es que si el resultado obtenido es igual al resultado esperado la prueba Pasa caso contrario Falla.

2.4 Entregables

Se confeccionará un documento de Resultado de las Pruebas.

3 Configuración del Test

Esta sección establece los componentes del ambiente de testing.

3.1 Hardware

El hardware con que se va a realizar el testing es con las siguientes máquinas con sus respectivas características:

- Computador personal con procesador AMD Athlon 64 * 2 Dual Core Processor 3800 + 2.01 Ghz, con 1 módulo de memoria de 2 GB RAM, 250 GB de disco rígido y además de contar un teclado, mouse y monitor para la realización de pruebas.
- PC de escritorio con un procesador Intel Core i5 de 2.9 GHZ, un módulo de memoria RAM de 4 GB, disco duro de 1 TB y sistema operativo WIndows 10 de 64 bits.
- Notebook con un procesador Intel Core i3 de 2.0 GHz, un módulo de memoria RAM de 4 GB, disco duro de 500 GB y sistema operativo Windows 10 de 64 bits.
- Notebook con un procesador Intel Core i5(7ma generación) de 2.7 GHz, un módulo de memoria RAM de 4 GB, disco duro de 500 GB y sistema operativo Windows 10 de 64 bits.
- Computador personal con procesador Intel Pentium 2.90 GHz con memoria RAM de 4 GB, 1 TB de disco rígido y sistema operativo Windows 8 de 64 bits.

3.2 Software

- Sistema Operativo Windows 7, 8 y 10
- Mobile: Android, IOS
- SQL Server 2014
- Visual Studio 2013 / 2015
- EAP v12.1.1224
- GitHub
- Navegadores Google Chrome Versión 73.0.3683.103 (32 Bits); Navegador Firefox versión 59.0.2; Opera versión 51.0.2830.34; Navegador Microsoft Edge versión 44.17763.1.0.

3.3 Ambiente de Prueba

Microsoft Office 2013 / 365 ProPlus – Word y Excel.

Pruebas unitarias JUnit – Unit Testing de Visual Studio.

4 Tareas

4.1 Actividades

La secuencia de actividades para probar el sistema es:

1. Ejecución de las pruebas en operación normal.
2. Corrección de errores.
3. Ejecución de las pruebas que fallaron en la primera pasada.
4. Repetición de los pasos 3 y 4 hasta que se logre un 99% de éxito y el 1% restante esté plenamente justificado.
5. Escritura del reporte.

4.2 Responsabilidades

Esta sección establece las responsabilidades de cada grupo que participa en la fase de pruebas.

4.2.1 Responsabilidades del Grupo de Desarrollo

- Ejecutar las pruebas unitarias.
- Ejecutar y probar la integración de bajo nivel.
- Corregir los problemas reportados.

4.2.2 Responsabilidades del Grupo de Testing

- Planificar las pruebas del sistema.
- Configurar el ambiente de prueba.
- Ejecutar las pruebas del sistema.
- Escribir el reporte de testing.

4.2.3 Responsabilidades del Grupo

- Proveer recursos.
- Aceptación final y aprobación de la liberación del producto.

4.3 Planificación

Casos de prueba de Casos de Uso.

Reporte de defectos.

Describir la planificación estimada de las actividades de testing.

Glosario (Definiciones y Siglas)

- **Calidad:** Conjunto de propiedades y de características de un producto o servicio, que le confieren su aptitud para satisfacer unas necesidades explícitas e implícitas.
- **Caso de testeo (test case):** conjunto de entradas, precondiciones para la ejecución y salidas esperadas desarrolladas con el objetivo de testear algo concreto del software (ejecutar un camino del programa en particular, verificar la conformidad de un requisito concreto, detectar tipos de errores específicos).
- **Defecto:** Paso, proceso, o definición de dato incorrecto. Ausencia de cierta característica.
- **Error:** una acción humana que produce resultados incorrectos.
- **Falla:** Resultado de ejecución incorrecto. Es el producido por el software distinto al resultado esperado.
- **Pruebas de Configuración (configuration testing):** Testeo de un sistema bajo diferentes configuraciones de:
 - Hardware: discos duros, impresoras, CPU, sensores, tarjetas gráficas, tarjetas de sonido, etc.
 - Sistemas operativos y/o versiones de un sistema operativo
 - Bases de datos, etc.
- **Prueba Exitosa:** Test que descubre defectos.
- **Pruebas de Instalación:** Prueba que indica las características del software que influyen en el esfuerzo requerido para instalarlo en un entorno especificado.
- **Prueba de Seguridad:** Prueba de un sistema de software la capacidad de prevenir acceso no autorizado.
- **Prueba de stress (stress testing):** Prueba del comportamiento del sistema bajo cargas muy altas con el objetivo de romper el sistema y encontrar los límites del sistema.
- **Prueba de Volumen:** Pruebas en las que el sistema está sometido a grandes volúmenes de datos.

ANEXO

Planilla de Ejecución

Planilla de Ejecución						
Fecha	Caso de Prueba	Estado	Tester	Defecto	Tiempo Ejecución	Tiempo Preparación

Planilla de Defectos

Nro de
Defecto:

Autor:

Fecha
Apertura:

Fecha Cierra:

Tipo:

Categoria:

Tipo de Defecto:

Severidad:

Prioridad:

Título:

Descripcion:

Adjunto:

Version:

Actividad:

Estado:

Build
Asigando:

Test Case:

Test Case

Id		Nombre	<nombre del caso de prueba>			
Requerimiento		<título del requerimiento>				
Autor		<autor>				
Descripción		<objetivo del caso de prueba>				
Prioridad		Alto		Medio		Bajo
Tipo de Testing						

Precondiciones:

Datos de Testing:

Pasos	Resultado Esperado
1.	
2.	
3.	
4.	

Post Condiciones:

Comentarios:

Historial de Revisión

Versión	Fecha	Detalle de Cambios	Autor