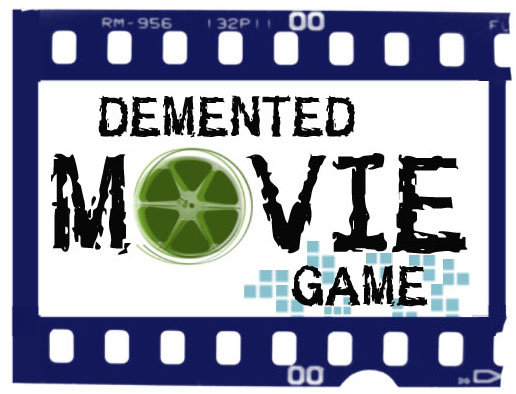
Versión 1.0

|  |
| --- |
| Ana María González Urueta Carlos Fernando Jaramillo Ortiz María Ximena Narváez Barrera Tatiana Alejandra Oquendo Garzón Víctor Hugo Villalobos Rodríguez Laura Catalina Zorro Jiménez |



|  |  |
| --- | --- |
| IMind | PLAN DE PRUEBAS |

TABLA DE CONTENIDO

[LISTA DE TABLAS 3](#_Toc229897092)

[LISTA DE FIGURAS 4](#_Toc229897093)

[**1.** **INTRODUCCIÓN** 5](#_Toc229897094)

[1.1. Objetivos 5](#_Toc229897095)

[1.2. Estrategia de pruebas 5](#_Toc229897096)

[1.3. Alcance 5](#_Toc229897097)

[1.4. Material de referencia 5](#_Toc229897098)

[1.5. Definiciones y acrónimos 6](#_Toc229897099)

[2. ARTEFACTOS DE PRUEBA 6](#_Toc229897100)

[2.1. Módulos del programa 7](#_Toc229897101)

[2.2. Procedimientos de Usuario: Jugador y Administrador 8](#_Toc229897102)

[**3.** **CARACTERÍSTICAS A SER PROBADAS** 9](#_Toc229897103)

[3.1. Requerimientos funcionales 9](#_Toc229897104)

[4. CARACTERÍSTICAS QUE NO SERÁN PROBADAS 14](#_Toc229897105)

[4.1. Requerimientos de desempeño 14](#_Toc229897106)

[4.2. Requerimientos no funcionales 14](#_Toc229897107)

[5. APROXIMACIÓN 15](#_Toc229897108)

[5.1. Pruebas Unitarias 16](#_Toc229897109)

[**5.2.** **Pruebas de Integración** 16](#_Toc229897110)

[**5.3.** **Pruebas de Sistema** 16](#_Toc229897111)

[**6.** **ANEXOS** 16](#_Toc229897112)

[**7.** **REFERENCIAS DE LA GUÍA** 16](#_Toc229897113)

# LISTA DE TABLAS

[Tabla 1. Material de referencia 5](#_Toc229897153)

[Tabla 2.Formato Módulos de Pruebas 6](#_Toc229897154)

[Tabla 3. Módulos de Pruebas 8](#_Toc229897155)

[Tabla 4. Formato Pruebas a Documentos de Usuario 9](#_Toc229897156)

[Tabla 5. Pruebas a Documentación 9](#_Toc229897157)

[Tabla 6. Control de prueba de requerimientos funcionales 14](#_Toc229897158)

[Tabla 7. Características No Probadas 14](#_Toc229897159)

[Tabla 8. Validación Requerimientos No Funcionales 15](#_Toc229897160)

[Tabla 9. Prueba Unitaria 16](#_Toc229897161)

# LISTA DE FIGURAS

[Ilustración 1. Descripción Campos Modulo de Pruebas 6](#_Toc229897169)

[Ilustración 2.Pruebas requerimientos 10](#_Toc229897170)

[Ilustración 3 . Items de pruebas de requerimientos 10](#_Toc229897171)

[Ilustración 4. Comprobación de Requerimientos 15](#_Toc229897172)

1. **INTRODUCCIÓN**

## Objetivos

Elaborar la documentación del plan de pruebas que permita al equipo de trabajo de IMind realizar validación y verificación de cada uno de los componentes del producto de software que se está desarrollando, guiados por los resultados arrojados por el proceso de recolección, análisis descripción y trazabilidad de requerimientos que se especifica en el documento SRS (*Ver documento SRS[IMind]\_V4.3(LineaBase)*).

Además pretende detectar el mayor número de defectos, errores y fallas que se presenten en la ejecución de la aplicación, en un porcentaje del 80%, debido al poco tiempo para realizar las pruebas, y finalmente probar la funcionalidad del sistema, lo que asegura la calidad del producto en el momento que se hace entrega al cliente.

## Estrategia de pruebas

Con el fin de realizar de forma ordenada las pruebas del producto de software, se tomará como guía la tabla de trazabilidad que se describe en el documento SRS (*Ver documento SRS[IMind]\_V4.3(LineaBase)*), debido a que en esta tabla se tienen cada uno de los requerimientos, definidos para el desarrollo y la implementación del sistema, asociados a los artefactos que se han entregado al cliente. Para llevar a cabo esta tarea se utilizará una lista de chequeo que permitirá saber el estado de cada requerimiento en las pruebas que se realicen, determinando si cumple o no y facilitará su control al tener la ubicación precisa y la relación entre ellos.

## Alcance

Las pruebas serán realizadas durante el proceso de implementación del sistema y el orden en que se realizarán será:

1. Pruebas unitarias
2. Pruebas de integración
3. Pruebas del sistema

Que serán explicadas más adelante en este mismo documento.

## Material de referencia

|  |  |
| --- | --- |
| **Material de Pruebas** | **Descripción** |
| **Documento de Pruebas IronWorks** | **Autor:** IronWorks |
| **Titulo del Articulo:** Plan de Pruebas V1.0 |
| **Año de Publicación:** 2007 |
| **Paginas:** 24 pág. |
| **Mejoramiento del proceso de Pruebas y corrección de defectos de software en un ambiente globalizado.** | **Autor:** Grupo de construcción de software Universidad de los andes. |
| **Título:** Plan de pruebas |
| **Año de publicación:** 2009 |

Tabla . Material de referencia

## Definiciones y acrónimos

* **Error**: “*Manifestación de un defecto durante la ejecución. Significa el sistema está en un estado tal que el procedimiento adicional del sistema conducirá a una falla.”*[2]
* **Componente: “***Una parte del sistema que puede ser aislada para pruebas*. *Un componente puede ser un objeto, un grupo de objetos, o uno o más subsistemas.”*[2]
* **Corrección:** Cambio a un componente. Su objetivo es reparar un defecto.[2]
* **Defecto: *“****Es una equivocación de diseño o código que puede causar comportamiento anormal del componente.”*[2]
* **Falla (Fracaso):** Desviación del comportamiento producido por uno o más errores.[2]

# ARTEFACTOS DE PRUEBA

Los módulos de prueba se especificarán a partir de los componentes encontrados dentro del Diagrama de Componentes de la aplicación y en particular en el diseño del sistema, que a su vez se relacionan de manera directa con la arquitectura escogida por el equipo y por el mismo cliente(ver documento *SDD[IMind]\_V2.6\_ (LineaBase)*). El formato diseñado por IMind\* se presenta a continuación:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MODULOS** | **SUBSISTEMA** | **PRUEBAS ASIGNADAS** | **DESCRIPCIÓN** | **ENCARGADO** |
|  |  |  |  |  |
|  |  |  |  |  |

Tabla 2.Formato Módulos de Pruebas

\**Inspirado en IronWorks*

Donde,

Ilustración 1. Descripción Campos Modulo de Pruebas

## Módulos del programa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MODULOS** | **SUBSISTEMA** | **PRUEBAS ASIGNADAS** | **DESCRIPCIÓN** | **ENCARGADO** |
| Cliente | USUARIO | Conexión al servidor | La conexión de cualquier jugador se presenta cuando este mismo ingresa su usuario, su contraseña y la Ip del servidor correctamente. | Director de Proyecto |
| Transacciones | Los usuarios realizan diferentes tipos de transacciones simultáneamente. |
| Solicitudes de estadísticas | Los usuarios realizan consultas que en varios casos necesita actualización de datos constante |
| Gestión de partida | Los usuarios tipo jugador pueden, dentro de una partida, guardar (manteniendo todos los datos de la misma) o retirarse (terminándola por completo). Por otro lado, el administrador puede retirar a un jugador del sistema. |
| Servidor | APLICACIÓN | Conexión multiusuario | El sistema permite comunicación cliente- servidor y cliente- cliente. Ping para probar que esto se realice correctamente. | Director de Configuración y Documentación |
| Número de clientes limite dentro de la aplicación | El sistema debe permitir un máximo de 49 jugadores dentro del sistema, para evitar un fallo del servidor de la aplicación |
| Conexión al servidor | Para que un usuario pueda ingresar a la aplicación, se pide el usuario, contraseña e ip del computador servidor. Se valida que estos datos sean correctos. |
| Chat | Los usuarios pueden enviarse mensajes unos a otros. Si todos pueden ver el mensaje enviado por un jugador, esta acción es exitosa. |
| Data Access Object | APLICACIÓN | Acceso a la información de los clientes | El administrador debe acceder y gestionar a la información de los usuarios registrados. | Director de Calidad y Manejo de Riesgos |
| Información de ubicación de los usuarios | El administrador puede ver en qué parte del sistema se encuentra un usuario en un momento dado. |
| Transacciones | La aplicación debe realizar de manera correcta la transacción pedida por cualquier usuario |
| Persistencia de Datos | PERSISTENCIA DE DATOS | Actualizaciones | La persistencia debe ser actualizada al realizarse un cambio dentro de una partida(con relación a los puntajes), o dentro de un registro, o una modificación a algún perfil | Analista de Requerimientos |

Tabla 3. Módulos de Pruebas

## Procedimientos de Usuario: Jugador y Administrador

IMind tomará 4 características importantes para cada documento entregado al usuario que son: Claridad, Coherencia, Correctividad, Coherencia, Completitud. En el siguiente formato se presentan los documentos a entregar, las pruebas a realizar por característica y el encargado de las mismas, por documento.

|  |  |  |  |
| --- | --- | --- | --- |
| **DOCUMENTOS** | **PRUEBAS ASIGNADAS** | **DESCRIPCIÓN** | **ENCARGADO** |
|  |  |  |  |
|  |  |  |  |

Tabla 4. Formato Pruebas a Documentos de Usuario

La siguiente tabla muestra entonces que:

|  |  |  |  |
| --- | --- | --- | --- |
| **DOCUMENTOS** | **PRUEBAS ASIGNADAS** | **DESCRIPCIÓN** | **ENCARGADO** |
| Manual de Usuario | Claridad | Se debe encontrar todas las instrucciones y procedimientos a seguir por el usuario, relacionados con la aplicación, de manera fácil para cualquier tipo de usuario. | Administrador de Configuración y Documentación |
| Coherencia | No existen contradicciones o ambigüedades en las secciones del documento. |
| Correctividad | No hay errores de ortografía, de hipervínculos entre secciones o algún parecido. |
| Completitud | Debe existir en el documento toda la información necesaria para el usuario. |
| Manual de Administrador | Claridad | Se debe encontrar todas las instrucciones y procedimientos a seguir por el usuario, relacionados con la aplicación, de manera fácil para cualquier tipo de usuario. | Administrador de Configuración y Documentación |
| Coherencia | No existen contradicciones o ambigüedades en las secciones del documento. |
| Correctividad | No hay errores de ortografía, de hipervínculos entre secciones o algún parecido. |
| Completitud | Debe existir en el documento toda la información necesaria para el usuario. |

Tabla . Pruebas a Documentación

1. **CARACTERÍSTICAS A SER PROBADAS**

## Requerimientos funcionales

Para probar los requerimientos funcionales se realizo una lista de verificación respecto a los requerimientos que son cumplidos en la última versión del prototipo de Demented Movie Game.

Ilustración .Pruebas requerimientos

Donde

Ilustración 3 . Items de pruebas de requerimientos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Localización final | |  |  |
| Id requerimiento | **Subsistema** | **Componente** | **Estado** | **Funcionalidad** |
| IMR001 |  |  |  | Registro y autenticación |
| IMR002 |  |  |  | Registro y autenticación |
| IMR003 |  |  |  | Registro y autenticación |
| IMR004 |  |  |  | Registro y autenticación |
| IMR005 |  |  |  | Registro y autenticación |
| IMR006 |  |  |  | Registro y autenticación |
| IMR007 |  |  |  | Registro y autenticación |
| IMR050 |  |  |  | Registro y autenticación |
| IMR051 |  |  |  | Registro y autenticación |
| IMR052 |  |  |  | Registro y autenticación |
| IMR053 |  |  |  | Registro y autenticación |
| IMR008 |  |  |  | Jugar |
| IMR009 |  |  |  | Jugar |
| IMR010 |  |  |  | Jugar |
| IMR011 |  |  |  | Jugar |
| IMR012 |  |  |  | Jugar |
| IMR013 |  |  |  | Jugar |
| IMR014 |  |  |  | Jugar |
| IMR015 |  |  |  | Jugar |
| IMR016 |  |  |  | Jugar |
| IMR017 |  |  |  | Jugar |
| IMR018 |  |  |  | Jugar |
| IMR019 |  |  |  | Jugar |
| IMR020 |  |  |  | Jugar |
| IMR021 |  |  |  | Jugar |
| IMR022 |  |  |  | Jugar |
| IMR023 |  |  |  | Jugar |
| IMR024 |  |  |  | Jugar |
| IMR025 |  |  |  | Jugar |
| IMR026 |  |  |  | jugar |
| IMR027 |  |  |  | Jugar |
| IMR028 |  |  |  | Jugar |
| IMR029 |  |  |  | Jugar |
| IMR030 |  |  |  | Jugar |
| IMR031 |  |  |  | Jugar |
| IMR032 |  |  |  | Jugar |
| IMR033 |  |  |  | Jugar |
| IMR034 |  |  |  | Jugar |
| IMR035 |  |  |  | Jugar |
| IMR036 |  |  |  | Jugar |
| IMR037 |  |  |  | Jugar |
| IMR038 |  |  |  | Jugar |
| IMR039 |  |  |  | Jugar |
| IMR040 |  |  |  | Jugar |
| IMR041 |  |  |  | Jugar |
| IMR042 |  |  |  | Jugar |
| IMR043 |  |  |  | Jugar |
| IMR044 |  |  |  | Jugar |
| IMR045 |  |  |  | Jugar |
| IMR046 |  |  |  | Jugar |
| IMR047 |  |  |  | Jugar |
| IMR048 |  |  |  | Jugar |
| IMR049 |  |  |  | Jugar |
| IMR054 |  |  |  | Jugar |
| IMR055 |  |  |  | Jugar |
| IMR056 |  |  |  | Jugar |
| IMR057 |  |  |  | Jugar |
| IMR058 |  |  |  | Jugar |
| IMR059 |  |  |  | Jugar |
| IMR060 |  |  |  | Jugar |
| IMR061 |  |  |  | Jugar |
| IMR062 |  |  |  | Consultas |
| IMR063 |  |  |  | Consultas |
| IMR064 |  |  |  | Consultas |
| IMR065 |  |  |  | Consultas |
| IMR066 |  |  |  | Consultas |
| IMR067 |  |  |  | Consultas |
| IMR068 |  |  |  | Administración |
| IMR069 |  |  |  | Administración |
| IMR070 |  |  |  | Administración |
| IMR071 |  |  |  | Administración |
| IMR072 |  |  |  | Administración |
| IMR073 |  |  |  | Administración |
| IMR074 |  |  |  | Administración |
| IMR075 |  |  |  | Administración |

Tabla 6. Control de prueba de requerimientos funcionales

# CARACTERÍSTICAS QUE NO SERÁN PROBADAS

Dentro de las características que no se van a probar se encuentran los requerimientos de desempeño y los requerimientos no funcionales. Simplemente se llevara un control de desarrollo que muestre si se cumplió con el requerimiento o no.

|  |  |  |
| --- | --- | --- |
| Característica | Justificación | Módulos afectados/consecuencias |
| Requerimientos de desempeño | La realización de pruebas para probar dichos requerimientos son difíciles de realizar, pero se tiene la ventaja que implícitamente al momento de probar los requerimientos funcionales se verifica algunos de estos | Se puede afectar la funcionalidad de jugar, perdiendo control en la cantidad de jugadores en un juego |
| Requerimientos no funcionales | No se probaran puesto que la gran mayoría de estos requerimientos son difíciles de probar por las implicaciones | No se tendrá certeza si algunas características cumplen con lo definido del sistema, las implicaciones que mas podrían afectar es el modulo de seguridad y disponibilidad. |

Tabla . Características No Probadas

## Requerimientos de desempeño

El control que se le tuvo sobre los requerimientos de desempeño estáticos y requerimientos de desempeño dinámicos se realizo mediante la comprobación del cumplimiento de dichos requerimientos mediante el uso de la trazabilidad. Ver anexo 1

## Requerimientos no funcionales

Para el control de los requerimientos no funcionales, se realizo una verificación de la última versión del prototipo de Demented Movie Game.

Ilustración . Comprobación de Requerimientos

|  |  |  |
| --- | --- | --- |
| Id requerimiento | Estado | Atributo |
| IMR083 |  | Confiabilidad |
| IMR084 |  | Confiabilidad |
| IMR085 |  | Disponibilidad |
| IMR086 |  | Disponibilidad |
| IMR088 |  | Seguridad |
| IMR089 |  | Seguridad |
| IMR090 |  | Seguridad |
| IMR091 |  | Seguridad |
| IMR092 |  | Mantenibilidad |
| IMR093 |  | Mantenibilidad |
| IMR094 |  | Portabilidad |
| IMR095 |  | Portabilidad |

Tabla . Validación Requerimientos No Funcionales

# APROXIMACIÓN

## Pruebas Unitarias

|  |  |  |  |
| --- | --- | --- | --- |
| **NOMBRE** | Prueba Unitaria | **IDENTIFICADOR** | IMP01 |
| **ACTIVIDADES** | Análisis de requerimientos del sistema.  Diseñar casos de prueba utilizando JUnit.  Validar y efectuar la aprobación del diseño de los casos de prueba. | | |
| **TIEMPO ESTIMADO** |  | | |
| **MÉTODOS O HERRAMIENTAS** | JUnit | | |
| **ENTREGABLES** | Informe generado por las pruebas de JUnit en formatos XML o HTML que muestran si funciona correctamente o no. | | |

* 1. **Pruebas de Integración**

|  |  |  |  |
| --- | --- | --- | --- |
| **NOMBRE** | Prueba de Integración | **IDENTIFICADOR** | IMP02 |
| **ACTIVIDADES** |  | | |
| **TIEMPO ESTIMADO** |  | | |
| **MÉTODOS O HERRAMIENTAS** | JUnit | | |
| **ENTREGABLES** | Informe generado por las pruebas de JUnit en formatos XML o HTML que muestran si funciona correctamente o no. | | |

* 1. **Pruebas de Sistema**

|  |  |  |  |
| --- | --- | --- | --- |
| **NOMBRE** | Prueba Unitaria | **IDENTIFICADOR** | IMP01 |
| **ACTIVIDADES** | Análisis de requerimientos del sistema.  Diseñar casos de prueba utilizando JUnit.  Validar y efectuar la aprobación del diseño de los casos de prueba. | | |
| **TIEMPO ESTIMADO** |  | | |
| **MÉTODOS O HERRAMIENTAS** | JUnit | | |
| **ENTREGABLES** | Informe generado por las pruebas de JUnit en formatos XML o HTML que muestran si funciona correctamente o no. | | |

Tabla . Prueba Unitaria

1. **ANEXOS**
2. **REFERENCIAS DE LA GUÍA**

[1]   IEEE Computer Society, IEEE Standard For Software Test Documentation, Disponible en http://ieeexplore.ieee.org/xpls/abs\_all.jsp?arnumber=573169, Visitada el Lunes 18 de Mayo del 2009.

[2] Sommerville I, INGENIERÍA DE SOFTWARE. Séptima Edición. Madrid. España: Pearson Educación; 2005.

[3] Bruegge B, Dutoit AH, INGENIERÍA DE SOFTWARE ORIENTADO A OBJETOS. Primera Edición. Naucalpan. México: Pearson Educación; 2002.

[4] Armando Botero Vila, Ivan Felipe Camero Padilla, Angela María Chaves Moreno, Andrés Galvis Rodríguez, Mauricio Nomesque Silva y Diana Carolina Ramirez Osorio. SMARTRUMMY-Q, SMARTWARE.

[5] Grupo de Construcción de Software Universidad de los Andes, Planes de Prueba, Disponible en: http://chie.uniandes.edu.co/~gsd/index.php?option=com\_content&task=category&sectionid=8&id=101&Itemid=183