|  |  |  |  |
| --- | --- | --- | --- |
| C:\Users\pei\AppData\Local\Microsoft\Windows\INetCache\Content.Word\logo.png | Plan de pruebas |  |  |

# PLAN DE PRUEBAS

**Sistema a probar: Nutrition Fitness Body**

## PLAN DE PRUEBAS

1. OBJETIVO

Generar pruebas de caja negra para realizar los requerimientos funcionales del software para permitir las condiciones de entrada y salida que ejerciten o complementen los requisitos funcionales del proyecto.

DESCRIPCIÓN DEL SISTEMA

Nutrition Fitness Body es un sistema de ventas online con propiedades básicas de inventario

1. MODULOS DEL SISTEMA A PROBAR – Componente VE

Se realizarán las pruebas de caja negra en el inicio de sesión y registrarse del sistema.

1. FORMULARIOS DEL APLICATIVO A PROBAR

Se realizarán pruebas de caja negra para analizar la lógica del inicio de sesión y registrarse del aplicativo.

1. METODOLOGIA PARA LA APLICACIÓN DE LAS PRUEBAS

La metodología utilizada para realizar esta prueba cuenta con una cuenta con una funcionalidad se verifica sin tomar en cuenta la estructura interna del código como detalle de implementación de ejecuciones internos en base al software o proyecto.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| C:\Users\pei\AppData\Local\Microsoft\Windows\INetCache\Content.Word\logo.png | Plan de pruebas | | |  | |  |
| **Identificador** | | Prueba de negra 1 | **Versión** | | 1 | |
| **Responsable** | | Brandon Dilan garzon Gómez | | | | |
| **Nombre del caso de prueba** | | Pruebas de caja negra | | | | |
| **Módulo** | | | **Submódulo** | | | |
| El modulo bloque o herramienta que se utilizó para hacer esta prueba fue un diagrama de proceso. | | | Se realizó las pruebas al inicio de sesión y registrarse del aplicativo web nuttrition fitnnes body | | | |
| **Formulario** | | | | | | |
| Formulario plan de pruebas | | | | | | |
| **Descripción de la prueba** | | | | | | |
| Generar pruebas de caja negra realizan o ejercitan los requerimientos funcionales desde el exterior del módulo. | | | | | | |
| **Resultados esperados** | | | | | | |
| El software está bien diseñado sin redundancia y una buena lógica | | | | | | |
| **Resultados reales** | | | | | | |
| El software está bien diseñado sin redundancia y una buena lógica | | | | | | |
| **Error** | | | | | | |
| No se presentaron errores | | | | | | |
| **Imagen** | | | | | | |
| Anexo 1    Anexo 2    Anexo 3    Anexo 4    Anexo 5    Anexo 6    Anexo 7 | | | | | | |

**Formato 1 Plan de pruebas de usuario**

Resultados posibles de las pruebas

|  |  |  |
| --- | --- | --- |
| Nombre del resultado de la prueba | Descripción | Impacto |
| Mal diseño del código y lógica | El sistema funciona de acuerdo a lo solicitado en los requerimientos iniciales. | Bajo |
| Buen diseño del código y lógica | El sistema no funciona de acuerdo a lo solicitado en los requerimientos iniciales. | bien |

1. CRONOGRAMA DE IMPLEMENTACION DE PRUEBAS

Anexo 1

1. RESPONSABLES DE LAS PRUEBAS

Desarrollador analista y responsable de las pruebas: Brandon Dilan Garzon Gómez

1. RIESGOS

No se registraron riesgos

1. RESPONSABILIDADES

Brandon Dilan Garzon Gómez