Resumen ejecutivo

Este documento, tiene como propósito definir las pautas, obligaciones, funcionalidades y la estrategia que seguirá para la entrega del proyecto Primate.

El objetivo de este plan es establecer pasos detallados de las pruebas para garantizar la calidad de los componentes del sistema, para su correcto funcionamiento.

Alcance de pruebas

Elementos a probar

|  |  |
| --- | --- |
| Capa | Componentes |
| Vista | [home.scala.html](http://home.scala.html/) |
| index.scala.html |
| catalog.scala.html |
| producto.scala.html |
| user.scala.html |
| gestionarProducto.scala.html |
| gestionarUsuario.scala.html |
| pedido.scala.html |
| estadoPedido.scala.html |
| pago.scala.html |
| Controllador | ProductController.java |
| UserController.java |
| HomeController.java |
| OrderController.java |
| PaymentController.java |
| StatesController.java |
| Modelo | User.java |
| Person.java |
| Product.java |
| Payment.java |
| Order.java |
| Rutas | routes |

Recursos

Por limitaciones de recursos se genera el entorno de pruebas en la mismo servidor que en el servidor de producción, se utilizara la base de datos inicialmente generada para entorno de producción, utilizando los mismos recursos de software y hardware previsto para este entorno.

Requerimientos de entorno - Hardware

* AMD FX 8320 a 3.5GHz
* 8192 MB de memoria RAM.
* 2438000 MB de espacio libre en el disco duro.
* Conexión de internet de 10MB/s download y 2 MB/s upload.

Requerimientos de pruebas - Software

Play Framework es un marco de trabajo (framework) muy completo que nos permite realizar test desde el mismo proyecto, realizando pruebas directas a los componentes creados sin ningún problemas, utilizando Junit y otras diferentes extensiones.

Además utilizaremos de Jmeter como herramienta de pruebas de carga y rendimiento.

Este es el resumen de todo el software utilizado en el entorno de pruebas para realizar dicha gestión.

* MySql Database 5.7.16
* Play framework 2.5.x
* JAVA version 1.5.1.101
* SBT 0.13
* Jquery 3.1.1
* Bootstrap 3.3.7
* Ebean 3.0.2
* coffeescript 1.0.0
* Sistema operativo: windows 10 profesional (Anniversary Update)
* Jmeter 3.0

Personal

* Alejandro Daza Truque - Tester General.
* Chirstian - Tester de rendimiento

Planificación para la pruebas

|  |  |
| --- | --- |
| Información general | |
| Identificador de caso de uso: | CU - 01 |
| Nombre de caso de uso: | Gestionar Usuarios |
| Descripcion Prueba | Validar usuarios |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Debe haber almenos un usuario ya registrado |  |
| Descripcion de casos de prueba | |
| Caso: | Verificar la validacion del ingreso del usuario al sistema |
| Servicios y Componentes Probados | |
| index.scala.html | |
| User.java | |
| LoginController.java | |
| Instrucciones de prueba | |
| 1 Ingresar datos invalidos y validos al template de ingreso | |
| 2. Verificar que se lee la base de datos para hacer su validadion | |
| 3. Cargar sesion del usuario | |
| Criterios de aceptacion | |
| 1 Que la sesion del usuario se carge correctamente | |
| 2 Que no depliegue ningun mensaje de error | |

|  |  |
| --- | --- |
| Informacion general | |
| Indentificador de caso de uso: | CU - 01 |
| Nombre de caso de uso: | Gestionar Usuarios |
| Descripcion Prueba | CRUD Usuarios |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Para despues de instruccion 1 tener mas de un usuario. | |
| Descripcion de casos de prueba | |
| Caso: | Verificar CRUD de usuarios del sistema |
| Servicios y Componentes Probados | |
| user.scala.html , gestionarUsuario.scala.html | |
| User.java | |
| UserController.java | |
| Instrucciones de prueba | |
| 1. Registrar Usuario | |
| 1.0 Ingresar a pagina de registro usuario | |
| 1.1. Ingresar datos al formulario de registro | |
| 1.3 Click en registrar | |
| 2. Buscar Usuario | |
| 2.1 Ingresar a la pagina gestionar Usuario | |
| 2.2 Ingresar numero id. | |
| 2.3 Click a buscar. | |
| 3. Actualizar Usuario | |
| 3.1 Ingresar datos actualizados en el formulario desplegado por buscar. | |
| 3.2 Dar click en actualizar usuario | |
| 4. Eliminar Usuario | |
| 4.1 Dar click en boton eliminar en el formulario desplegado por buscar. | |
| Criterios de aceptacion | |
| 1.1 Que cargue la pagina correctamente | |
| 1.2 Que no despliegue ningun mensaje de error en ninguno de sus campos | |
| 1.3 Verificar si el modelo se creo con la informacion del cliente y guardo esa informacion en la base de datos. | |
| 2.0 Que carge la pagina correctamente | |
| 2.1 que no muestre ningun error en el campo id | |
| 2.2 Que el query para el modelo este correcto, que la informacion que se cargue sea la correcta. | |
| 3.1 Que ningun campo del formulario no tenga ningun mensaje de error | |
| 3.2 Que el query para actualizar el usuario en la base de datos este correcto, que actualize los campos en capos correctamente. | |
| 4.1 Que el query para eliminar el usuario en la base de datos este correcto, que se remueva el usuario de la base de datos y despliegue el mensaje de respuesta correcto. | |

|  |  |
| --- | --- |
| Descripcion de casos de prueba | |
| Caso: | Verificar CRUD productos del sistema |
| Indentificador de caso de uso: | CU - 01 |
| Nombre de caso de uso: | Gestionar Usuarios |
| Descripcion Prueba | CRUD Usuarios |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Para despues de instruccion 1 tener mas de un Producto. | |
| Descripcion de casos de prueba | |
| Caso: | Verificar CRUD de productos del sistema |
| Servicios y Componentes Probados | |
| product.scala.html , gestionarProducto.scala.html | |
| Producto.java | |
| ProductController.java |  |
| Instrucciones de prueba |  |
| 1. Registrar Producto |  |
| 1.0 Ingresar a pagina de registro de producto |  |
| 1.1. Ingresar datos al formulario de producto |  |
| 1.3 Click en registrar |  |
| 2. Buscar Producto |  |
| 2.1 Ingresar a la pagina gestionar Producto |  |
| 2.2 Ingresar numero id del producto. |  |
| 2.3 Click a buscar. |  |
| 3. Actualizar Producto |  |
| 3.1 Ingresar datos actualizados en el formulario desplegado por buscar. | |
| 3.2 Dar click en actualizar producto |  |
| 4. Eliminar Producto |  |
| 4.1 Dar click en boton eliminar en el formulario desplegado por buscar. | |
| Criterios de aceptacion |  |
| 1.1 Que cargue la pagina correctamente. | |
| 1.2 Que no despliegue ningun mensaje de error en ninguno de sus campos. | |
| 1.3 Verificar si el modelo se creo con la informacion del producto y guardo esa informacion en la base de datos. | |
| 2.0 Que carge la pagina correctamente. | |
| 2.1 Que no muestre ningun error en el campo id. | |
| 2.2 Que el query para el modelo este correcto, que la informacion que se cargue sea la correcta. | |
| 3.1 Que ningun campo del formulario no tenga ningun mensaje de error. | |
| 3.2 Que el query para actualizar el producto en la base de datos este correcto, que actualize los campos en capos correctamente. | |
| 4.1 Que el query para eliminar el producto en la base de datos este correcto, que se remueva el usuario de la base de datos y despliegue el mensaje de respuesta correcto. | |

|  |  |
| --- | --- |
| Informacion general | |
| Indentificador de caso de uso: | CU - 03 |
| Nombre de caso de uso: | Gestionar Pedido |
| Descripcion Prueba | Procesos de Pedido |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Debe haber cargado la pagina de catalogo. | |
| Descripcion de casos de prueba | |
| Caso: | Verificar el correcto funcionamiento de los procesos del pedido |
| Servicios y Componentes Probados | |
| pedido.scala.html | |
| Pedido.java | |
| PedidoController.java | |
| Instrucciones de prueba | |
| 1. Ingresar a pagina del catalogo de productos. | |
| 2. Click en agregar a pedido cuando no haya ningun producto agregado. | |
| 2.1 Click en agregar al pedido cuando por lo menos haya un producto. | |
| 2.2 Click en remover en la ventana adicional del pedido. | |
| 3. Click en confirmar pedido. | |
| Criterios de aceptacion | |
| 1. Que la pagina del catalogo se depliegue correctamente con todos los productos disponibles actualmente y sus respetivos botones. | |
| 2 Que cree el modelo correctamente, y despliegue la ventana adicional del pedido. | |
| 2.1 Que agregue el producto y desplegue correctamente la lista de producto en la ventana adicional. | |
| 2.2 Que se remueva el producto de la lista del pedido y depliegue correctamente la lista. | |
| 3. Guarda todos los atributos del pedido, crea su modelo y guarda los datos en la bd. | |

|  |  |
| --- | --- |
| Informacion general | |
| Indentificador de caso de uso: | CU - 04 |
| Nombre de caso de uso: | Gestionar Pagos |
| Descripcion Prueba | Validar Pagos |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Debe haber el pedio seleccionado | |
| Descripcion de casos de prueba | |
| Caso: | Verificar la compra y pago del pedido para continuar con la produccion de este |
| Servicios y Componentes Probados | |
| pago.scala.html | |
| peido.java | |
| pagoController.java | |
| Instrucciones de prueba | |
| 1. Dar click en confirmar pedido en el catalogo. | |
| 2. Seleccionar el metodo de pago. | |
| 3. Ingresar la informacion del formulario dependiendo de el metodo de pago seleccionado. | |
| 4. Dar click en confirmar pago. | |
| Criterios de aceptacion | |
| 1. Que la pagina de pagos se depliegue correctamente con sus componentes. | |
| 2. Que al seleccionar metodo de pago se pespliegue el formulario correctamente. | |
| 3. Que no se despliegue ningun mensaje de error en el formulario. | |
| 4. Que se pueda confirmar transaccion en la cuenta bancaria, que se cree el modelo correctamente con sus respectivas asociasiones, y se guarde la informacion en la db. | |

|  |  |
| --- | --- |
| Informacion general | |
| Indentificador de caso de uso: | CU - 05 |
| Nombre de caso de uso: | Gestionar Produccion y Entrega |
| Descripcion Prueba | Validar cambio de estados de produccion |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Ingresar a la sesion operador y seleccionar pedido disponible. | |
| Descripcion de casos de prueba | |
| Caso: | Verificar que los estados del pedido se guarde correctamente y la interfaz se deplegue correctamente. |
| Servicios y Componentes Probados | |
| estadoPedido.scala.html | |
| pedido.java | |
| pedidoController.java | |
| Instrucciones de prueba | |
| 1. Ingresar a la pagina de gestion de estados del pedido. | |
| 2. Seleccionar pedido disponible. | |
| 3. Dar click en el boton siguiente estado o estado anterior. | |
| Criterios de aceptacion | |
| 1. Que se desplegue correctamente la pagina con todo sus componentes. | |
| 2. Que se genere correctamente el query para la obtener la informacion, que se cree el modelo correctamente, y se genere la cista respectiva a ese pedido (resumen del pedido y estado actual) | |
| 3. Que se genere el modelo correctamente con el estado actual, se guarde el estado del pedido en la base de datos para ser revisado por el cliente, si se encuentra en el pedido finalizado se devuelve a la pagina para buscar pedido o seleccionarlo. | |

|  |  |
| --- | --- |
| Informacion general | |
| Indentificador de caso de uso: | CU - 05 |
| Nombre de caso de uso: | Gestionar Produccion y Entrega |
| Descripcion Prueba | Desplegar correctamente el estado de produccion. |
| Responsable: | Alejandro Daza |
| Prerrequisitos | |
| Ingresar a la sesion de cliente y seleccionar estado de mi pedido. | |
| Descripcion de casos de prueba | |
| Caso: | Verificar que la barra de estado de su pedido este despleguando correctamente le estado actual comparando con el estado que se encuentre en la base de datos. |
| Servicios y Componentes Probados | |
| estadoPedido.scala.html | |
| pedido.java | |
| pedidoController.java | |
| Instrucciones de prueba | |
| 1. Ingresar a la pagina de mi pedido. | |
| Criterios de aceptacion | |
| 1. Que se desplegue correctamente la pagina con todo sus componentes, que se refresque el componente de la barra cada media hora para verificar estado actual del producto y de la opcion de compartir al finalizar la entrega. | |