Documento de Microdiseño

**CASO DE USO: Registrar Subasta**

Sistema de Subastas en Línea

Versión 1.0

Historial de Versiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 08/06/2014 | 1.0 | Versión preliminar como una propuesta de desarrollo. | Leonidas Muñoz |

|  |  |
| --- | --- |
| **Detalle de**  **Diseño N** | 04 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Elaborado por:** | Leonidas Muñoz | | | | |
| **Descripción General:** | * Este documento sirve de guía al desarrollo de la funcionalidad: Registrar subasta * Caso de uso Registrar subasta:   Permite al usuario ingresar información acerca de la subasta e información del artículo a subastar. | | | | |
| **Referencias:** | **Requerimientos:** | SSEL-LR.xlsx  RF-02 – Registrar subasta | | | |
| **Diseño de Interfaz de Usuario:** | SSEL-ECU-RS.docx  Especificación de Casos de Uso Registrar subasta | | | |
| **Diseño de Arquitectura:** |  | | | |
| **Clases Requeridas:** | **Nombre de la clase** | | **Stereotype** | | **Acción** |
| RegistroSubastaController.java | | Controlador | | **registrarSubasta**() |
| SubastaServicio.java (SPI)  SubastaServicioImpl.java | | Interfaz Servicio (SPI – Service Provider Interface)  Implementación | | **registrarSubasta**(idUsuario : Integer, subasta : Subasta):Integer |
| SubastaDAO.java  SubastaDAOImpl.java | | Modelo | | **ingresar**(subasta : Subasta) : Boolean  **ingresar**(artículo : Articulo) : Boolean |
| Subasta.java  Articulo.java  Imagen.java | | Controlador | | Getters  Setters |
| **Vistas** | **Nombre del archivo de vista** | | | | **Acción** |
| regSubastas.xhtml | | | | **registrarSubasta** |
| **Tablas:** | **Nombre de Tabla** | | | **Tipo de Objeto** | |
| Usuario | | | Tabla | |
| **Detalle de Implementación a Nivel de Clases:** | Usuario.java   * Clase en java que implementa el modelo de Usuario con las operaciones básicas de usuario. * Acción: Autenticar, recuperar, registrar.   Subasta.java   * Clase en java que implementa el modelo de Subasta con las operaciones de listar, crear y eliminar. * Acción: listar, crear, eliminar   Articulo.java   * Clase en java que implementa el modelo de Artículo con las operaciones de listar, crear y eliminar. * Acción: listar, crear, eliminar   SubastarController.java   * Clase en java que implementa al controlador de la funcionalidad de crear una subasta. * Acción: crear, eliminar. | | | | |
| **Detalle de implementación a Nivel de Vista:** | subastas.html   * Acción: crear, eliminar * Descripción:   Se encarga de renderizar la vista que muestra el formulario donde se ingresara los datos de la subasta y del articulo a subastar como nombre de articulo, tipo de articulo, monto base, fecha de Inicio, fecha de fin, y el botón para registrar y otro para cancelar. | | | | |
| **Detalla de Implementación a Nivel de Tablas:** | **Tabla de Usuario**  CREATE TABLE IF NOT EXISTS ‘usuario’ (  ‘idusuario’ int(11) NOT NULL,  `vc\_nombreusuario` varchar(60) NOT NULL,  `vc\_nombres` varchar(60) NOT NULL,  ‘vc\_dni’ varchar(10),  ‘vc\_correoe’ varchar(40),  ‘vc\_telefono’ int(12),  ‘fl\_saldo’ float,  ‘vc\_domicilio’ varchar(80),  ‘dt\_fnacimiento’ date,  ‘vc\_contrasena’ varchar(11),  ‘bo\_estado’ bool,  PRIMARY KEY (`idUsuario`),  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;  **Tabla de Subasta**  CREATE TABLE IF NOT EXISTS ‘subasta’ (  ‘in\_idSubasta’ int(11) NOT NULL,  ‘dt\_inicioubasta’ date,  ‘dt\_finsubasta’ date,  ‘fl\_preciobase’ float NOT NULL,  ‘vc\_descripcion’ varchar(1024),  ‘vc\_telefonocontacto’ varchar(12),  ‘vc\_lugarentrega’ varchar(255),  ‘in\_estado’ int(1),  PRIMARY KEY (‘in\_idsubasta’),  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;  **Tabla de Articulo**  CREATE TABLE IF NOT EXISTS ‘producto’ (  ‘in\_idarticulo’ int(11) NOT NULL,  ‘vc\_nombre’ varchar(60),  ‘vc\_descripcion’ varchar(60),  PRIMARY KEY (‘idProducto’),  ) ENGINE=InnoDB DEFAULT CHARSET=latin1; | | | | |