|  |  |
| --- | --- |
| **Detalle de**  **Diseño N** | 05 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Elaborado por:** | Jorge Delgado | | | | |
| **Descripción General:** | Este documento sirve de guía al desarrollo de la funcionalidad:  Caso de uso Visualizar Subastas:   * Permite al usuario visualizar las subastas registradas en el sistema. | | | | |
| **Referencias:** | **Requerimientos:** | SSEL-LR.xlsx  RF-07 – Visualizar Subasta | | | |
| **Diseño de Interfaz de Usuario:** | SSEL-ECU-VS.docx  Especificación de Casos de Uso Visualizar Subasta | | | |
| **Diseño de Arquitectura:** |  | | | |
| **Clases Requeridas:** | **Nombre de la clase** | | **Stereotype** | | **Acción** |
| Subasta.java | | Modelo | | Crear, eliminar |
| Producto | | Modelo | | Crear, listar, eliminar |
| Usuario | | Modelo | | Recuperar |
| ListarSubastasController.java | | Controlador | | listar |
| **Vistas** | **Nombre del archivo de vista** | | | | **Acción** |
| Listar\_subastas.html | | | | listar |
| **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   Producto.java   * Clase en java que implementa el modelo de Producto 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 listar las subastas. * Acción: listar. |
| **Detalle de implementación a Nivel de Vista:** | Listar\_subastas.html   * Acción: listar * Descripción:   Se encarga de renderizar la vista que muestra el listado de subastas registradas en el sistema. |
| **Detalla de Implementación a Nivel de Tablas:** | **Tabla de Usuario**  CREATE TABLE IF NOT EXISTS ‘usuario’ (  ‘idUsuario’ int(11) NOT NULL,  `nombre` varchar(30) NOT NULL,  ‘apellido’ varchar(20) NOT NULL,  ‘sexo’ varchar(1) NOT NULL,  ‘estado’ int(1),  ‘correo’ varchar(20),  ‘telefono’ int(10),  ‘username’ varchar(20),  ‘contrasena’ varchar(8),  ‘saldoDisponible’ doublé,  ‘fechaRegistro’ date,  ‘direccion’ varchar(50),  PRIMARY KEY (`idUsuario`),  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;  **Tabla de Subasta**  CREATE TABLE IF NOT EXISTS ‘subasta’ (  ‘idSubasta’ int(11) NOT NULL,  ‘precioBase’ double NOT NULL,  ‘precioProductoEnSubasta’ double NOT NULL,  ‘estadoSubasta’ int(1),  ‘fechaInicioSubasta’ date,  ‘fechaFinSubsta’ date,  ‘tipoPago’ varchar(2),  ‘cantidadPostores’ int(6),  ‘cantidadApuestas’ int(6),  PRIMARY KEY (‘idSubasta’),  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;  **Tabla de Producto**  CREATE TABLE IF NOT EXISTS ‘producto’ (  ‘idProducto’ int(11) NOT NULL,  ‘nombre’ varchar(20),  ‘descripcion’ varchar(100),  ‘precioBase’ double NOT NULL,  ‘estado’ int(1),  ‘fechaRegistro’ date,  ‘URL\_imagen’ varchar(50),  PRIMARY KEY (‘idProducto’),  ) ENGINE=InnoDB DEFAULT CHARSET=latin1; |