Documento de Microdiseño

**CASO DE USO: Buscar tesis por categoría**

Versión 1.0

Historial de Versiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 20/10/2014 | 1.0 | Versión preliminar como una propuesta de desarrollo. | Akira Tabuchi |
| 19/11/2014 | 1.1 | Modificación luego de SC0001 | Akira Tabuchi |
|  |  |  |  |

|  |  |
| --- | --- |
| **Detalle de Diseño N°** | 01 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Elaborado Por:** | **Akira Tabuchi** | | | |
| **Descripción General:** | Este documento está orientado a servir de guía en el desarrollo de la funcionalidad:    ***Buscar tesis por categoría:***     * Esta funcionalidad tiene como finalidad la búsqueda de una determinada tesis o conjunto de tesis según los siguientes campos: Titulo, descripción, autor, tema, fecha. | | | |
| **Referencias:** | **Requerimientos:** | SisCoTe\_ECU\_BT.docx  Requerimiento funcional RF-05 | | |
| **Diseño de Interfaz Usuario:** | SisCoTe\_ECU\_BT.docx  Prototipos asociados al Caso de Uso # 1 | | |
| **Diseño de**  **Arquitectura:** | SisCote\_DD.docx | | |
| **Otros:** | Ninguno | | |
| **Clases**  **Requeridas:** | **Nombre de la Clase** | | **Stereotype** | **Acción** |
| DAOTesisImpl | | DAO | Crear |
| DAOTesis.java | | DAO Interface | Crear |
| Documento.java | | Domain Class | Crear |
| AdministrarTesis.java | | Servicie | Crear |
| **Clases Java Swing** | **Nombre del archivo de vista (JSF)** | | | **Acción** |
| UIAdministración.java | | | Crear |
| **Descriptores Hibernate - XML** | **Nombre del archivo descriptor Hibernate - XML** | | | **Acción** |
| hibernate.cfg.xml | | | Crear |
| Hibernate.reveng.xml | | | Crear |
| **Tablas:** | **Nombre de Tabla** | | **Tipo de Objeto** | |
| CATEGORIA (select) | | Tabla | |
| DOCUMENTO (select, insert, update) | | Tabla | |
| TEMA (select, insert) | | Tabla | |
| AUTOR (select, insert, update) | | Tabla | |
| ESPECIALIDAD (select) | | Tabla | |
| AREA\_INVESTIGACIÒN (select, insert, update) | | Tabla | |

|  |  |
| --- | --- |
| **Detalle de Implementación a Nivel de Clases:** | **pe.com.siscote.model.Documento**     * Clase java utilizada para mapear y tener acceso la información de la tabla Documento.      * **Métodos**:     Dado que se trata de una clase de dominio que solo será utilizado como un objeto consultador de información, los únicos métodos a implementar son los accesores: ***Getters* y *Setters*** por cada propiedad/atributo de la clase.     * 1. **public TipoDatosAtributo** getNombreAtributo()   2. **public void** setNombreAtributo()      * **Acción:**     Buscar |
| **Detalle de implementación a nivel de Servicios** | **pe.com.siscote.model.service.AdministrarTesis**     * Clase que contiene la lógica de negocio del sistema, con las operaciones de de agregar, modificar, eliminar y buscar documentos de tesis.      * **Métodos**:      * 1. **public void** AgregarTesis()   2. **public Boolean** EliminarTesis()   3. **public Boolean** ModificarInfoTeis()   4. **public Boolean** ModificarDocTeis()   5. **public Documento** BuscarTesis(Documento)      * **Acción:**     Buscar |
| **Detalle de implementación a nivel de DAOs** | **pe.com.sirh.model.dao.DAOTesis**     * Interfase utilizada para desacoplar las capas implementadas en el sistema (capa de *Datos* de la capa *Negocio*).      * **Métodos:**      * 1. **public void** crearDocumento(Documento)   2. **public void** modificarDocumento(Documento)   3. **public Documento** consultarDocumento(**INT** idDocumento) 4. **public List**<Documento> listarDocumento()      * **Acción:**     Buscar    **pe.com.sirh.model.dao.impl.DAOTesisImpl**     * Clase de acceso a datos que extiende de la clase Hibernate ***y HibernateUtil*** e implementa la interfase ***DAOTesis***. Implementa métodos de acceso a los las clases de dominio descritas en el hibernate.cfg.xml.      * **Métodos:**      * 1. **public void** crearDocumento(Documento)   2. **public void** modificarDocumento(Documento)   3. **public Documento** consultarDocumento(**INT** idDocumento) 4. **public List**<Documento> listarDocumento()        * **Acción:**     Buscar |
| **Detalle de Implementación a Nivel de JSF:** | **UIAdministración.java**   * **Descripción:**   Clase swing de la vista que ofrece la funcionalidad de administrar los documentos de tesis, permite registrarlos, modificarlos o buscar.    Contiene los campos en un formulario:    Nombres del autor  Título del documento  Palabra clave del documento  Año de publicación del documento  Descripción del documento  Área de investigación del documento   * **Acción:**     Buscar |
| **Detalle de Implementación a Nivel de Tablas:** | **Tabla DOCUMENTO**    SELECT \* FROM ‘documento’ WHERE  ‘id’ = ‘codigo’ or  ‘vc\_titulos’=’ titulos’ or  ‘vc\_fecha`=’ fecha or  ‘vc\_especialidad’=’ ‘especialidad’ or  ‘vc\_categoria’=’ ‘categoría’, |