

LABINFO -MONITORES



FELIPE TARQUINO  
SANTIAGO SOSSA  
LEONARDO CASTRO  
ALEJANDRO VILLARRAGA  
FELIPE DIAZ

ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO  
PROCESOS DE DESARROLLO DE SOFTWARE  
BOGOTÁ, COLOMBIA  
2015

## Contenido

1.	Introducción .....	3
2.	Descripción del producto .....	3
3.	Plataformas soportadas .....	3
4.	Librerías Usadas .....	3
5.	Modelo Conceptual .....	4
6.	Modelo relacional .....	6
7.	Diagramas de secuencia .....	7
7.1.	Planeador Monitores .....	7
7.2.	Gestión de la Bitácora .....	8

## 1. Introducción

El laboratorio de sistemas de sistemas de la Escuela colombiana de Ingeniería tiene como componentes de recurso humano monitores que ayudan a los estudiantes (usuarios) de la decantura de Ingeniera de Sistemas, y a su vez coordinadores que se encargan de delegar las funciones que deben cumplir los monitores en sus respectivos turnos . Actualmente el proceso de delegar tareas a los monitores se hace de forma manual, se desea desarrollar una aplicación que agilice este proceso con el fin de aprovechar al máximo el tiempo de los monitores.

## 2. Descripción del producto

La aplicación fue desarrollada para tres tipos de usuarios (roles) coordinadores, monitores y estudiantes, quienes para poder usar la aplicación deben ingresar su código y un contraseña suministrada anteriormente en la página de inicio. La aplicación tiene distintas funcionalidades dependiendo del rol que acceda a ella, por ejemplo, los estudiantes puede reportar un problema, los monitores pueden consultar las actividades que se les recomienda realizar en su turno de trabajo y a su vez pueden reportar las labores realizadas finalmente los coordinadores pueden consultar los reportes de trabajo y consultar las labores realizadas para sacar conclusiones y que permitan mejorar el servicio prestado por el laboratorio.

## 3. Plataformas soportadas

Al ser una aplicación web la soportan las siguientes plataformas

- Windows
- Linux

## 4. Librerías Usadas

Estas son algunas de las librerías usadas a lo largo del desarrollo de la aplicación:

- Java.sql.connection
- Java.util.date

- `Java.sql.PreparedStatement`
- `Java.sql.resultSet`
- `Java.sql.exception`
- `javax.faces.bean.ManagedBean`
- `javax.faces.bean.SessionScoped`
- `java.io.IOException`
- `org.primefaces.context.RequestContext`
- `org.primefaces.event.SelectEvent`
- `javax.faces.model.SelectItem`
- `javax.faces.model.SelectItemGroup`
- `java.util.Iterator`
- `java.util.Properties`

## 5. Modelo Conceptual

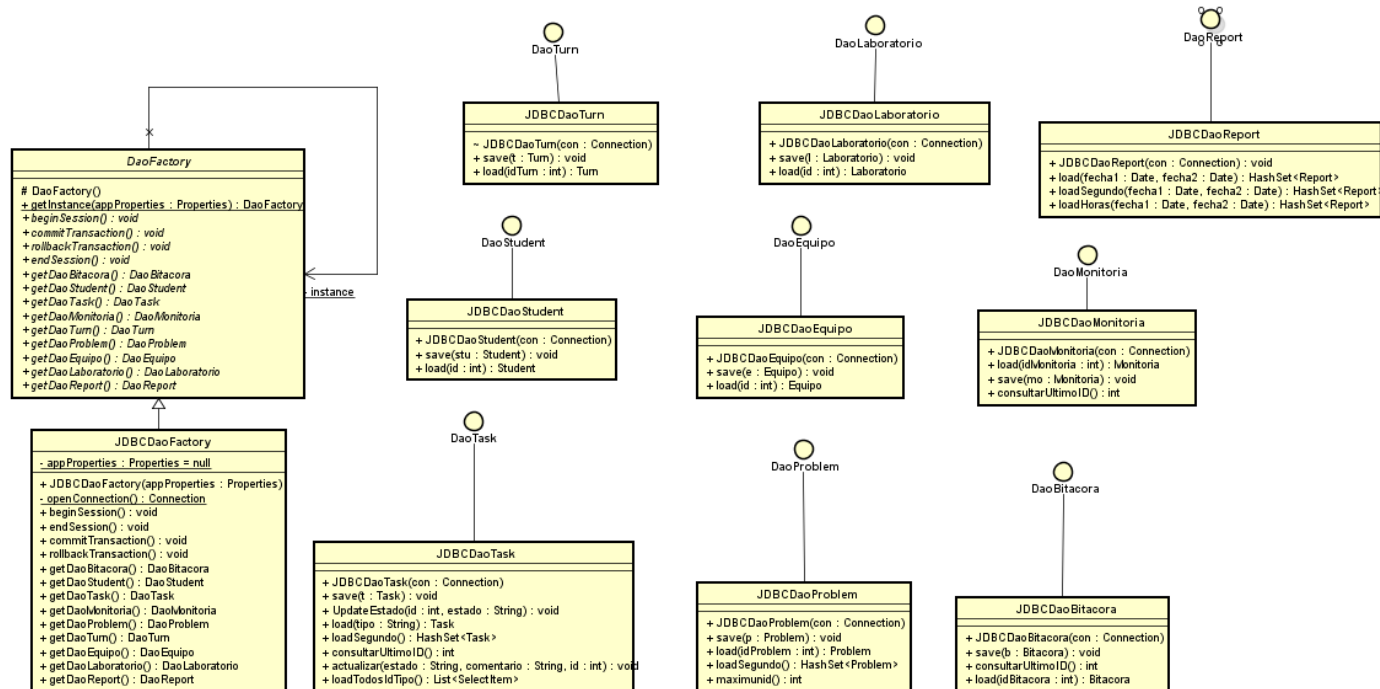
A continuación se presenta el siguiente modelo conceptual que satisface los requerimientos establecidos al inicio del proyecto por el cliente a través de las historias de uso.

### **Modelo conceptual (entidades)**

- Bitácora
- Laboratorio
- Reporte
- Monitoria
- Equipo
- Problema
- Tarea
- Estudiante
- Turno



Extendido

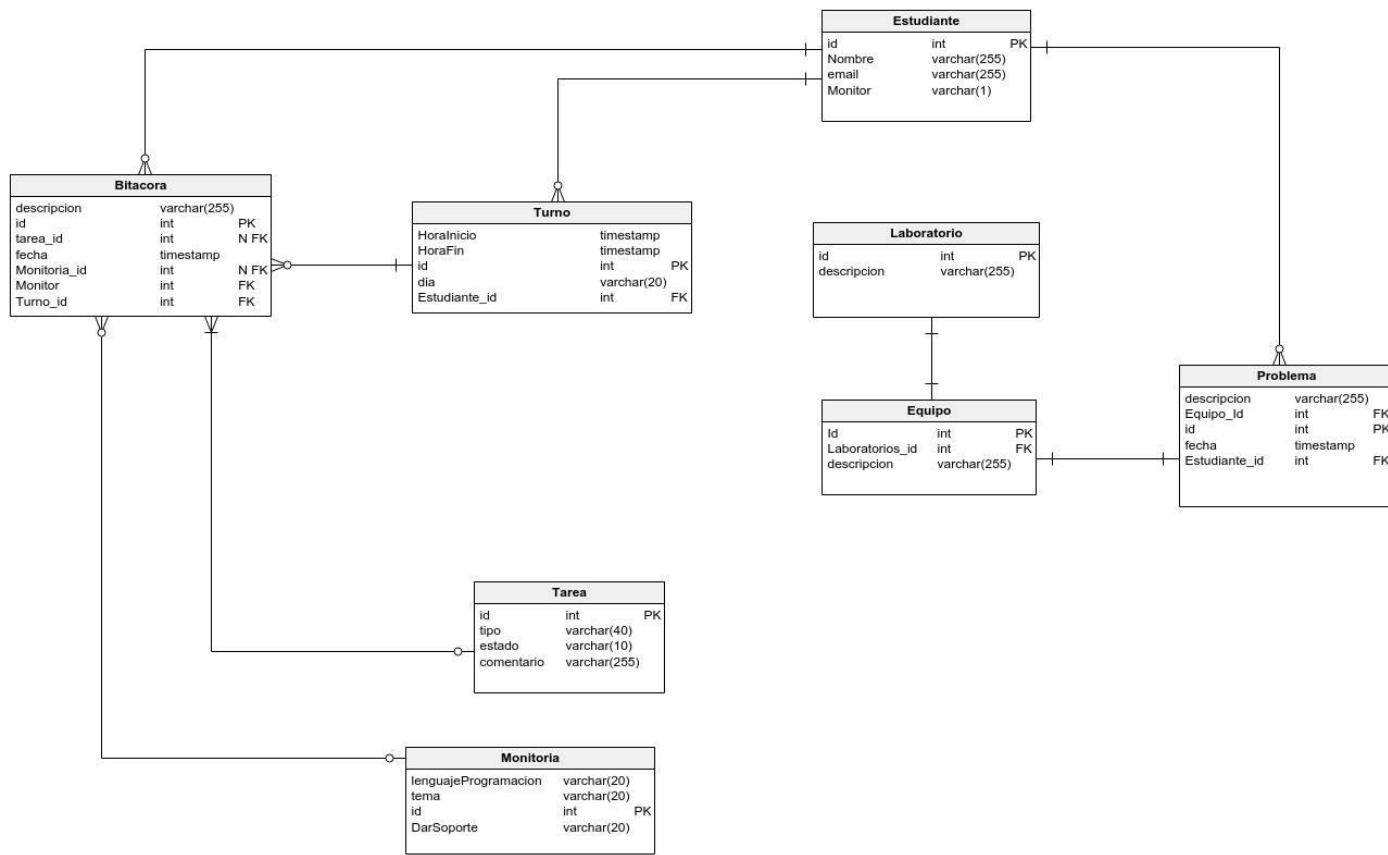


## 6. Modelo relacional

El siguiente es el modelo de datos de la aplicación que cuenta con ocho (8) entidades que contienen información relevante para las aplicaciones.

Entidades:

- Estudiante: a través de esta entidad se obtiene el código, nombre e email de una un usuario (estudiante) y a su vez su es estudiante un monitor de laboratorio.
- Bitácora: genera un reporte hecho por un monitor del laboratorio con información relacionada de una tarea o un soporte realizado
- Turno: tiempo (hora inicio – fin) que un monitor trabaja en el laboratorio en un día determinado.
- Laboratorio
- Equipo: recurso que hace parte del laboratorio de informática y es usado por los estudiantes.
- Problema
- Tarea: labor que desarrolla un monitor del laboratorio.
- Monitoria

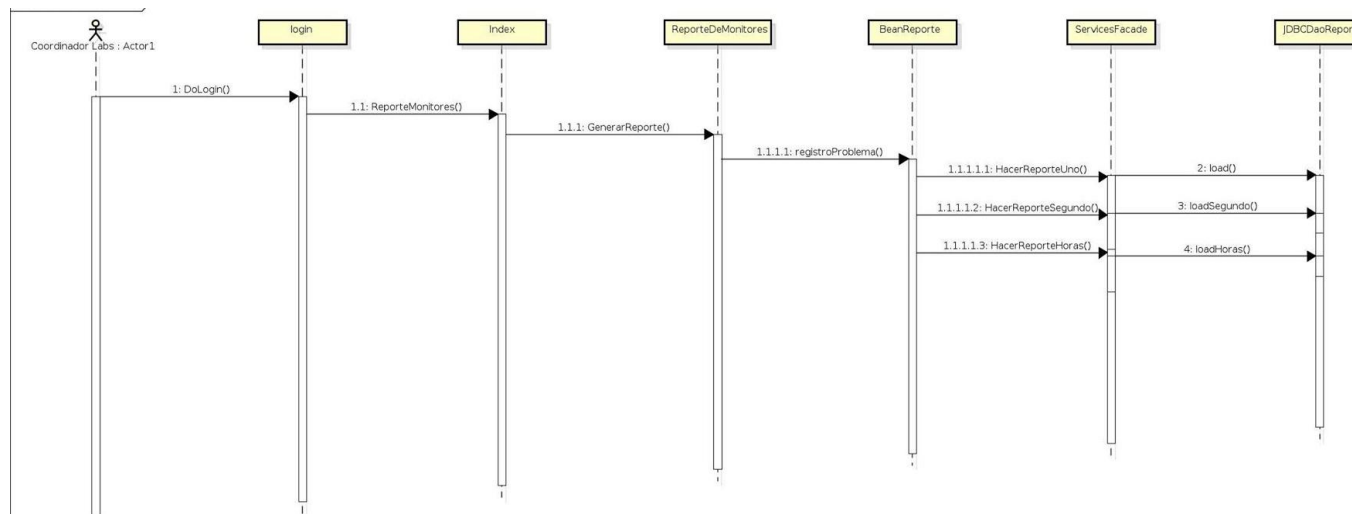


## 7. Diagramas de secuencia

### 7.1. Planeador Monitores

El diagrama de secuencia contiene los un actor que es el coordinador del laboratorio y los siguientes componentes:

- login
- Index
- Reporte de monitores
- beanReporte
- ServiceFacade
- JDBCDaoReport



## 7.2. Gestión de la Bitácora

El diagrama de secuencia contiene los un actor que es el coordinador del laboratorio y los siguientes componentes:

- login
- Index
- registroDeBitacora
- backingBean
- ServiceFacade
- JDBCDaoBitacora



Sequence Diagram0

