**TRABAJO FINAL**

**SISTEMA DE MANTENIMIENTO DE PERMISOS DE ACCESO APLICACIONES**

**INTEGRANTES:**

* Andy Cotillo Meneses
* Alberto Joaquin Carranza Varias
* Arturo Eduardo Camayo Córdova
* Joel Carhuaricra Colqui
* Jair Ladislao Caycho Vega

**DOCENTE:**

Eric Gustavo Coronel Castillo

**CURSO:**

Java 6

**San Isidro, 17 de Julio de 2018**

**ACERCA DE ESTE PROYECTO**

Una de las tareas más recurrentes es el mantenimiento de los perfiles de usuario, tarea que conlleva el 80% de las labores diarias del personal encargado.

Lamentablemente, el desarrollo individual de las aplicaciones no ha contemplado un módulo unificado que administre de manera centralizada la asignación de perfiles y permisos de acceso a las distintas aplicaciones.

Este módulo busca palear esta carencia y centralizar la administración de la asignación de perfiles y accesos, reduciendo el tiempo de atención de estos requerimientos.

Para el desarrollo de este proyecto se han tenido en cuenta todos los conceptos y técnicas de programación aprendidos en el desarrollo del presente curso.

**OBJETIVO GENERAL**

Implementar una aplicación en Web, con conexión a base de datos, que permita gestionar los perfiles y permisos de acceso del personal de la institución.

**OBJETIVOS ESPECÍFICOS**

* Desarrollar el diseño de la base de datos.
* Definir los procesos.
* Desarrollar los requerimientos para la implementación.
* Desarrollar el software.

**REQUERIMIENTOS**

* Control de acceso a la aplicación Web.
* Listar a todos los trabajadores.
* Filtrar, de acuerdo a la siguiente regla: Apellidos y/o Nombres.
* Gestión de cambio de perfiles.
* Activar o inactivar los usuarios.

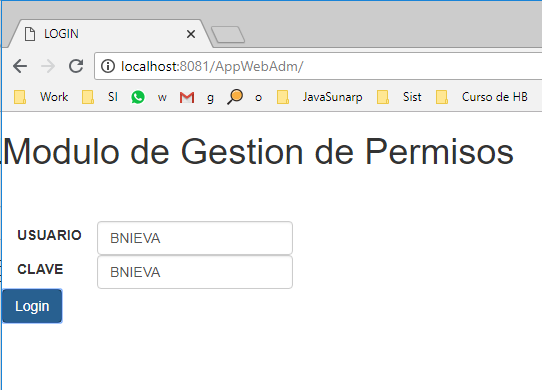
**FUNCIONALIDAD**

Como primera etapa, dos de las principales aplicaciones existentes basan su comportamiento en el cargo asignado a un trabajador, es por ello que, para esta oportunidad, el módulo sólo contempla las siguientes funcionalidades:

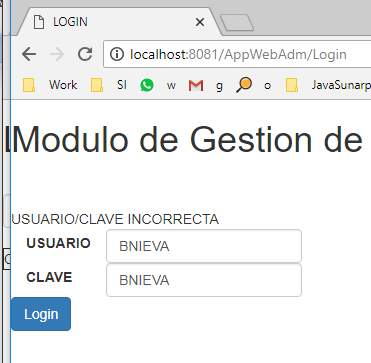
* Login: validando las credenciales de acceso del personal encargado.
* Búsqueda y filtro: submódulo que permite ubicar a uno o varios trabajadores y mostrar en una tabla información de los mismos., permitiendo acceder a funcionalidades rápidas como cambio de estado y editar los datos.
* Mantenimiento de empleado: permite la actualización de los datos de un empleado, así como si su estado y cargo actual que ocupa.

**CAPTURAS DE PANTALLA**

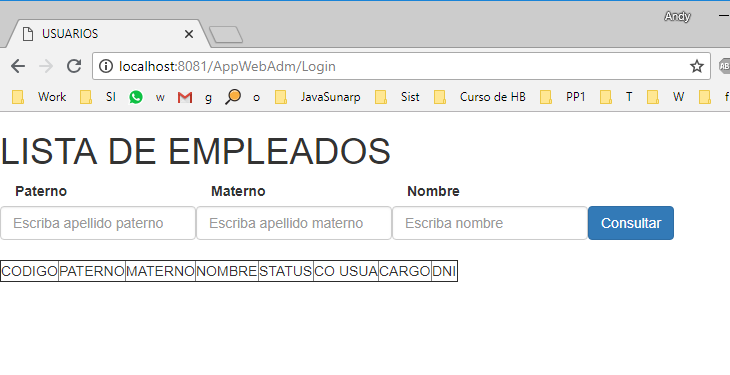
1. **Login**



Validación de credenciales:

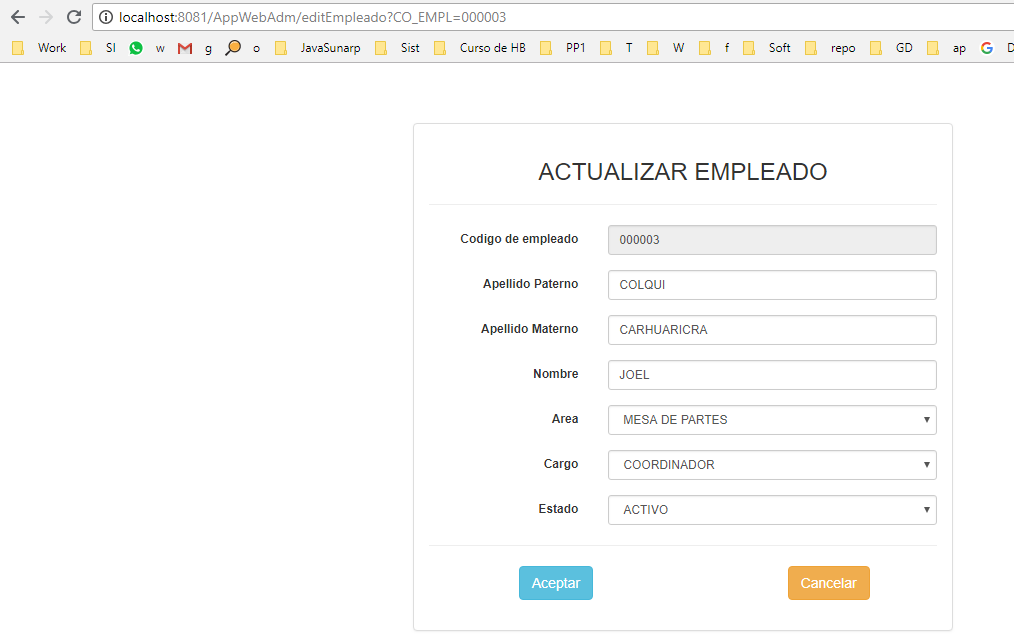


1. **Lista Empleado**





1. **Editar Empleado**



**CONCLUSIONES**

El trabajo desarrollado brinda cumple con los objetivos del proyecto, toda vez que permite resolver en el menor tiempo posible las solicitudes de cambio de perfiles y permisos de los usuarios, así como las altas y bajas.

**RECOMENDACIONES**

Ampliar la solución para Integrar las otras aplicaciones existentes en la institución.

**Anexos**

**CREACION DE LA BASE DE DATOS**

create table AREA(

CO\_AREA CHAR(2),*--PK*

DESC\_AREA VARCHAR2(50) NOT NULL,

DESC\_CORT VARCHAR2(20) NOT NULL,

EST\_REGI CHAR(1) DEFAULT 'A'

);

create table CARGO(

CO\_CARGO CHAR(2),*--PK*

DESC\_CARGO VARCHAR2(100) not null,

EST\_REGI CHAR(1) DEFAULT 'A' not null

);

create table CARGO\_DET(

CO\_CARGO CHAR(2),*--PK-FK*

DET\_CARGO VARCHAR2(10) not null

);

create table EST\_EMPLEADO(

EST\_EMPL CHAR(1),*--PK*

ESTADO VARCHAR2(15) not null

);

create table EMPLEADO(

CO\_EMPL CHAR(6),*--PK*

CO\_AREA CHAR(2) not null,*--FK*

USU\_SIST VARCHAR2(15) not null,

CO\_CARGO CHAR(2) not null,*--FK*

EST\_EMPL CHAR(1) not null ,*--FK*

CLAVE VARCHAR2(15) not null,

APE\_PAT VARCHAR2(30) not null,

APE\_MAT VARCHAR2(30) not null,

NOMBRE VARCHAR2(30) not null,

CO\_EMPL\_REG CHAR(6),

CO\_EMPL\_MOD VARCHAR2(6),

FEC\_REG DATE,

FEC\_MOD DATE

);

create table PERSONA(

CO\_EMPL CHAR(6),*--PK --FK*

APE\_PAT VARCHAR2(30) not null,

APE\_MAT VARCHAR2(30) not null,

NOMBRE VARCHAR2(30) not null,

DNI CHAR(8) not null,

DIRECC VARCHAR2(50) not null,

TELF VARCHAR2(7),

CEL VARCHAR2(9),

EMAIL VARCHAR2(30),

CO\_EMPL\_REG CHAR(6),

CO\_EMPL\_MOD VARCHAR2(6),

FEC\_REG DATE,

FEC\_MOD DATE

);

*---------------- LLAVES ----------------*

ALTER TABLE AREA ADD CONSTRAINT PK\_AREA PRIMARY KEY (CO\_AREA);

ALTER TABLE CARGO ADD CONSTRAINT CARGO\_PK\_CO\_CARGO PRIMARY KEY (CO\_CARGO);

ALTER TABLE CARGO\_DET ADD CONSTRAINT PK\_CARGO\_DET PRIMARY KEY (CO\_CARGO);

ALTER TABLE CARGO\_DET ADD CONSTRAINT FK\_CARGO\_DET\_CO\_CARGO

FOREIGN KEY (CO\_CARGO) REFERENCES CARGO(CO\_CARGO);

ALTER TABLE EST\_EMPLEADO ADD CONSTRAINT EST\_EMPLEADO\_PK\_EST\_EMPL PRIMARY KEY (EST\_EMPL);

*/\*\*/*

ALTER TABLE EMPLEADO ADD CONSTRAINT EMPLEADO\_PK\_CO\_EMPL PRIMARY KEY (CO\_EMPL);

*/\*ALTER TABLE EMPLEADO*

*ADD CONSTRAINT EMPLEADO\_UNIQ\_USU\_SIST UNIQUE (USU\_SIST);\*/*

ALTER TABLE EMPLEADO ADD CONSTRAINT FK\_EMPLEADO\_CO\_AREA

FOREIGN KEY (CO\_AREA) REFERENCES AREA(CO\_AREA);

ALTER TABLE EMPLEADO ADD CONSTRAINT FK\_EMPLEADO\_CO\_CARGO

FOREIGN KEY (CO\_CARGO) REFERENCES CARGO(CO\_CARGO);

ALTER TABLE EMPLEADO ADD CONSTRAINT FK\_EMPLEADO\_EST\_EMPL

FOREIGN KEY (EST\_EMPL) REFERENCES EST\_EMPLEADO(EST\_EMPL);

ALTER TABLE PERSONA ADD CONSTRAINT PERSONA\_PK\_CO\_EMPL PRIMARY KEY (CO\_EMPL);

ALTER TABLE PERSONA ADD CONSTRAINT FK\_PERSONA\_CO\_EMPL

FOREIGN KEY (CO\_EMPL) REFERENCES EMPLEADO(CO\_EMPL);

*---*

* TABLAS PARA EL CONTROL DE ACCESO A LA APLICACION

**-** EMPLEADO

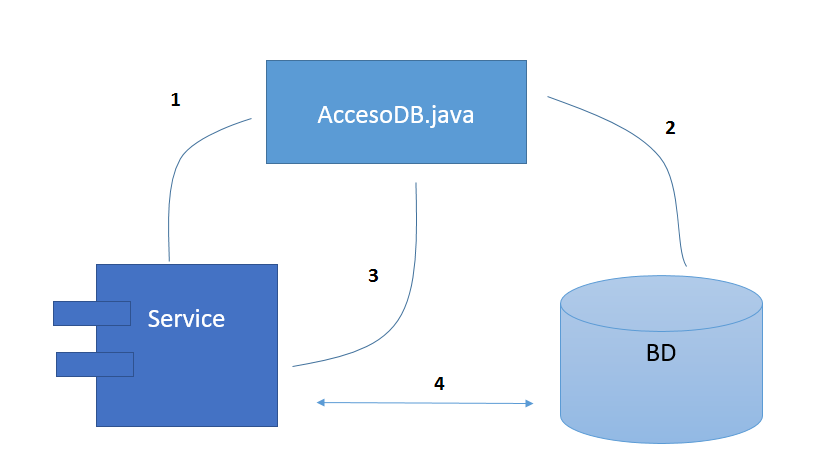
* TABLAS PARA LA LISTA DE TRABAJADORES DE LA SUNARP

**-** EMPLEADO, CARGO , AREA , EST\_EMPLEADO

* TABLAS PARA EL CONTROL Y CAMBIO DE PERFILES

**-** EMPLEADO, CARGO , AREA , EST\_EMPLEADO

**CONECCION DE LA BASE DE DATOS**



**ARQUITECTURA DE LA APLICACION**

