

FACULTAD DE INGENIERÍA

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS

**PROYECTO “EDUTEC”**



Autores:

Alvarez Huamaní, Izabo Rosario

Huaman Francia, Yassel Aracely

Asesor:

Coronel Castillo, Eric Gustavo

Lima - Perú

2017

**ÍNDICE**

[I. INTRODUCCION 3](#_Toc500006470)

[II. CUERPO 1](#_Toc500006471)

[2.1 BASE DE DATOS 1](#_Toc500006472)

[2.1.1 MODELO LOGICO 1](#_Toc500006473)

[2.1.2 MODELO FISICO 2](#_Toc500006474)

[2.1.3 MODELO DE LA BASE DE DATOS 3](#_Toc500006475)

[2.2 ANÁLISIS 1](#_Toc500006476)

[2.2.1 CASOS DE USOS 1](#_Toc500006477)

[2.2.2 ANALISIS DE CASOS DE USOS 2](#_Toc500006478)

[CONSULTAR ALUMNO 2](#_Toc500006479)

[REGISTRAR ALUMNO 3](#_Toc500006480)

[REGISTRAR TARIFA 4](#_Toc500006481)

[REGISTRAR PROFESOR 5](#_Toc500006482)

[REGISTRAR EMPLEADOS 6](#_Toc500006483)

[CONSULTAR PROFESOR 7](#_Toc500006484)

[CONSULTAR CICLO 8](#_Toc500006485)

[CONSULTAR TARIFA 9](#_Toc500006486)

[2.2.3 DIAGRAMA DE SECUENCIA 10](#_Toc500006487)

[2.2.4 DIAGRAMA DE ACTIVIDADES 13](#_Toc500006488)

[2.2.5 DIAGRAMA DE CLASES 14](#_Toc500006489)

[2.3 APLICACIÓN 19](#_Toc500006490)

[2.3.1 CODIGO JAVA 19](#_Toc500006491)

[2.3.2 VISTAS EN SQL 48](#_Toc500006492)

# **INTRODUCCION**

EduTec es una institución que ofrece cursos de capacitación en temas específicos sobre temas de ingeniería de software.

Este programa se denomina “Cursos de Extensión Profesional”, ya que cualquier persona puede matricularse en cualquiera de los cursos, las veces que desee, no existe ningún tipo de restricción.

EduTec requiere una aplicación que le permita gestionar su programa de cursos de extensión.

CRITERIOS DEL NEGOCIO

Normalmente, todo el curso tiene una duración de 24 horas, y se dicta en 8 sesiones de 3 horas cada una, dos sesiones por semana. Pueden existir cursos de menor o mayor duración.

Los fines de semana se realizan cursos que tienen sesiones con una duración de 6 horas, por lo tanto, duran 4 fines de semana.

Los cursos se agrupan en ciclos, los cuales se inician normalmente la primera semana de cada mes.

Un profesor no puede ser programado en más de cuatro cursos por ciclo.

La evaluación de los cursos está dada por un examen parcial (40%) y un examen final (60%).

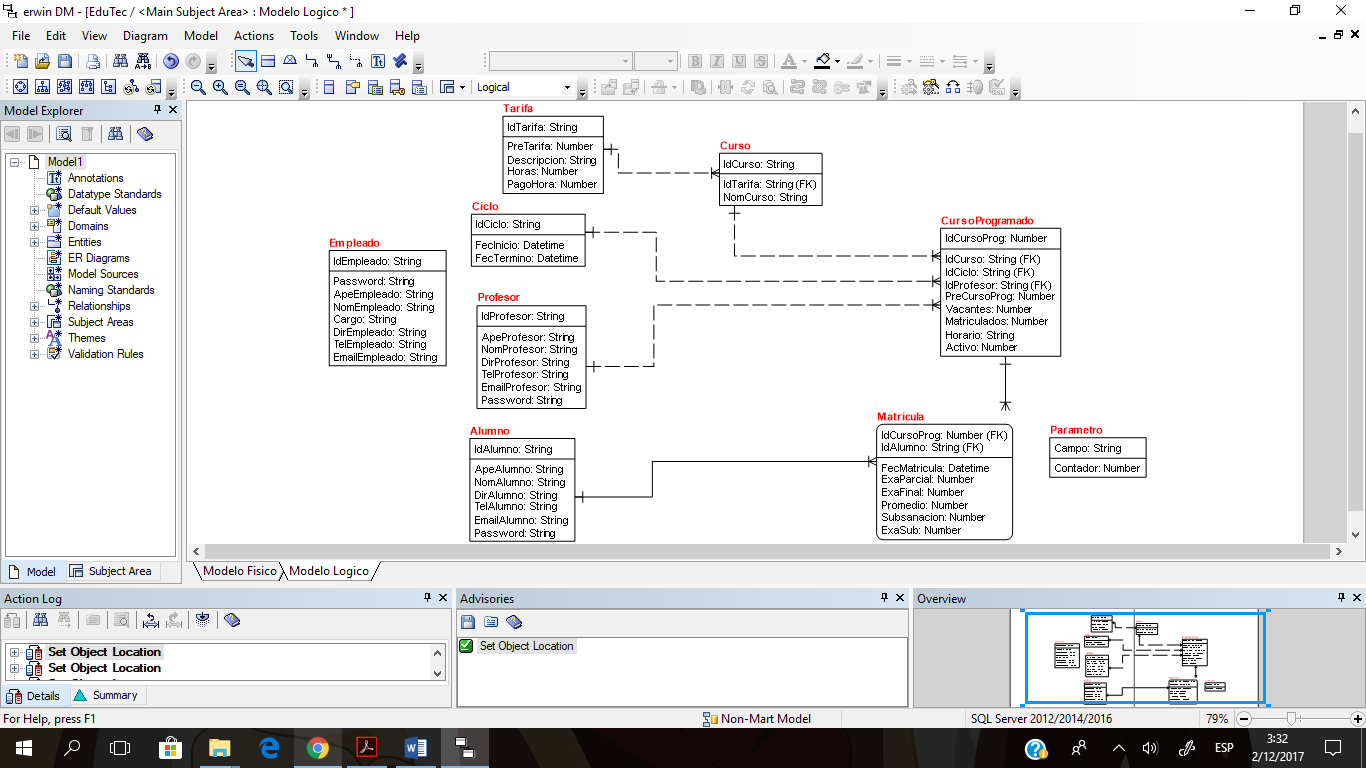
Para que un alumno reciba certificación, debe obtener un promedio mayor o igual a 13.

Si un alumno desaprueba con un promedio mayor o igual que 10, tiene derecho a rendir un examen de subsanación, la nota de este examen reemplaza a la nota que más le conviene.

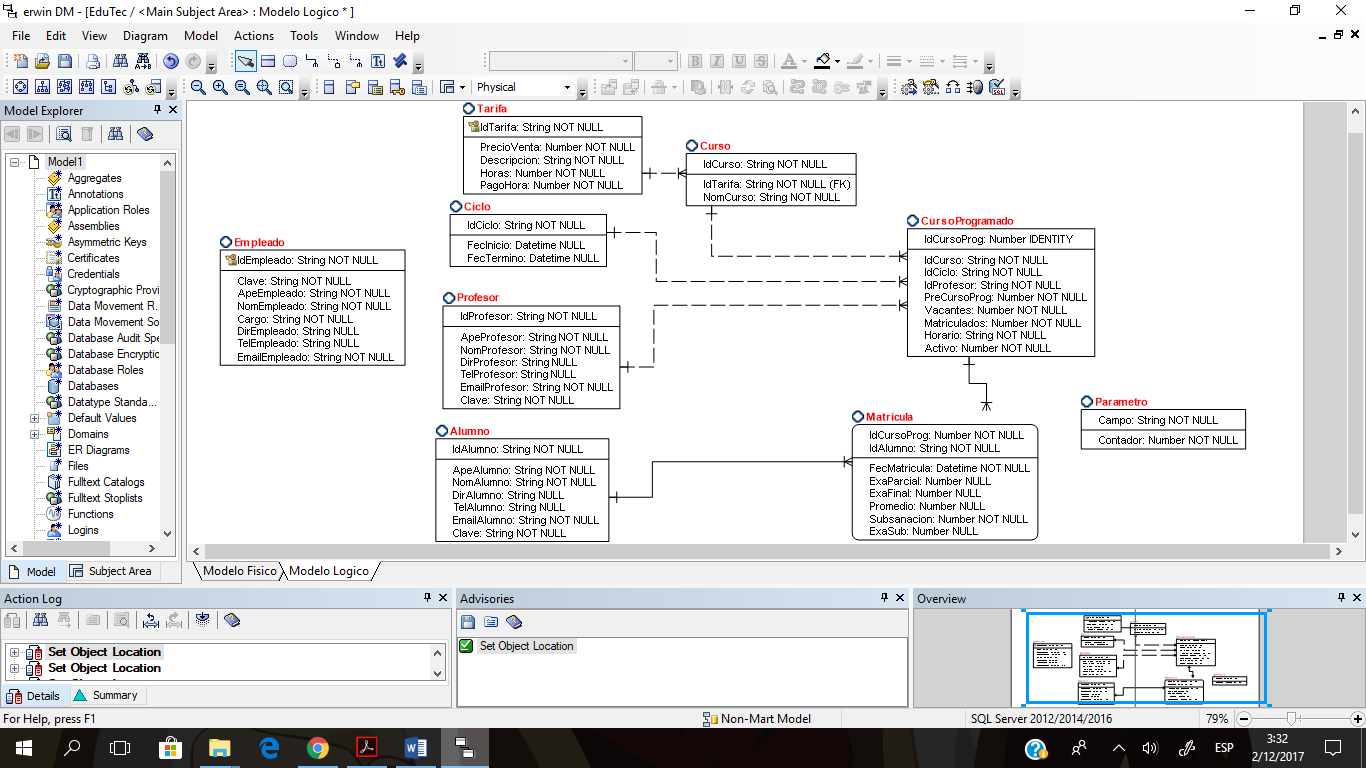
# **CUERPO**

## **BASE DE DATOS**

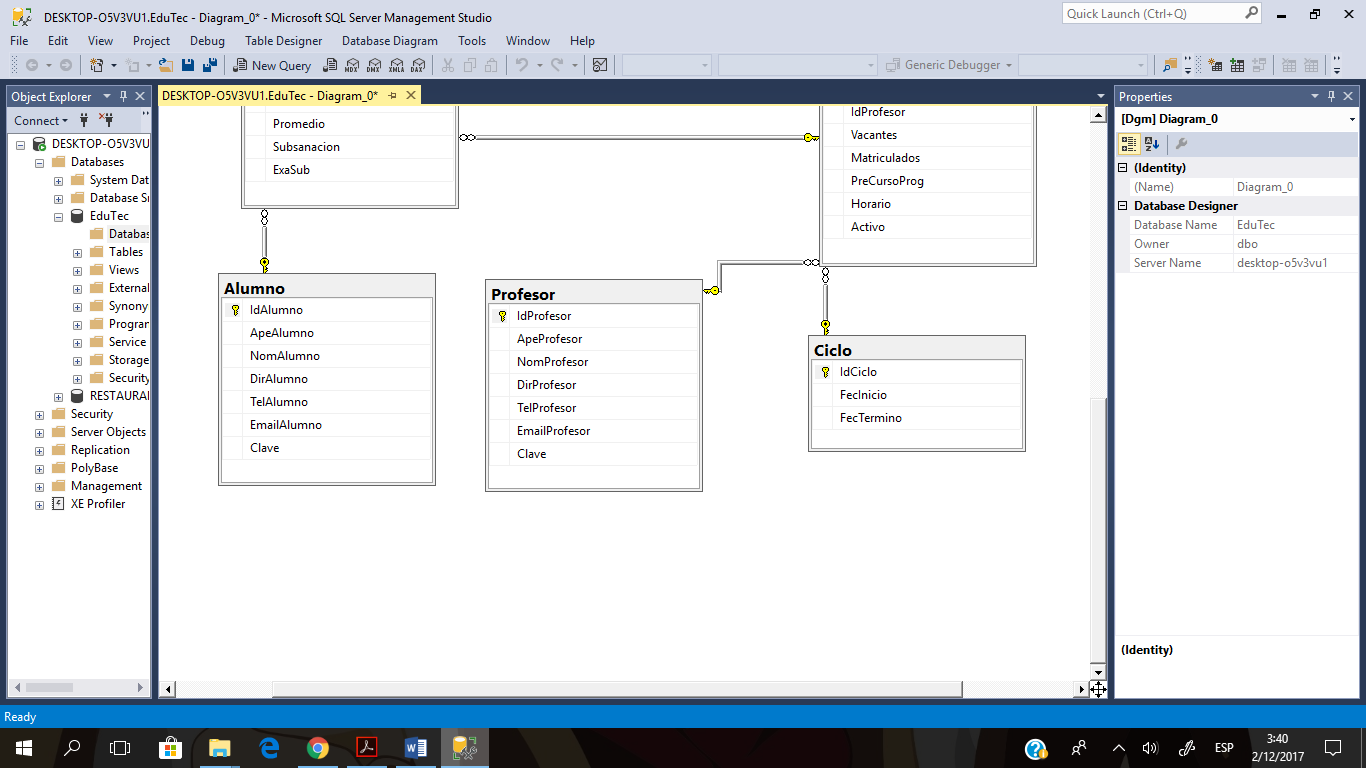
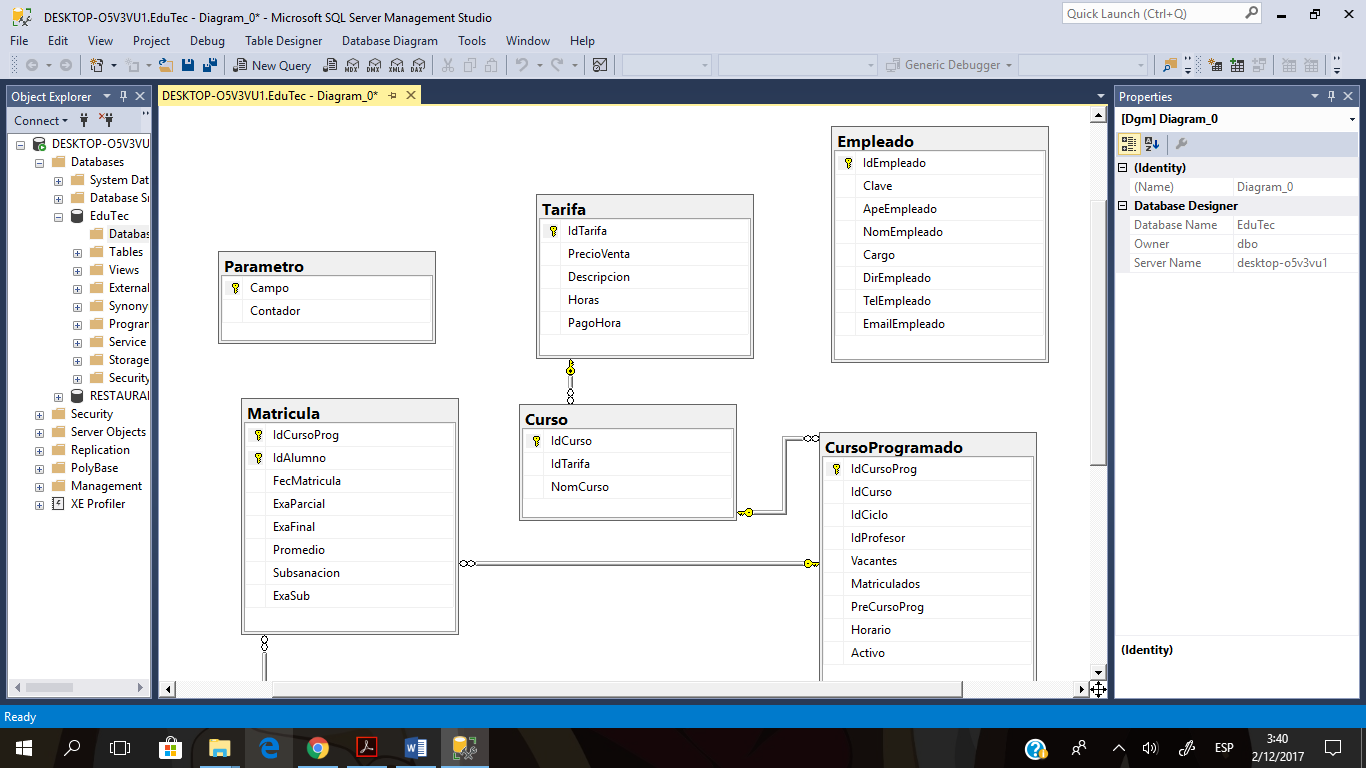
### **MODELO LOGICO**



### **MODELO FISICO**

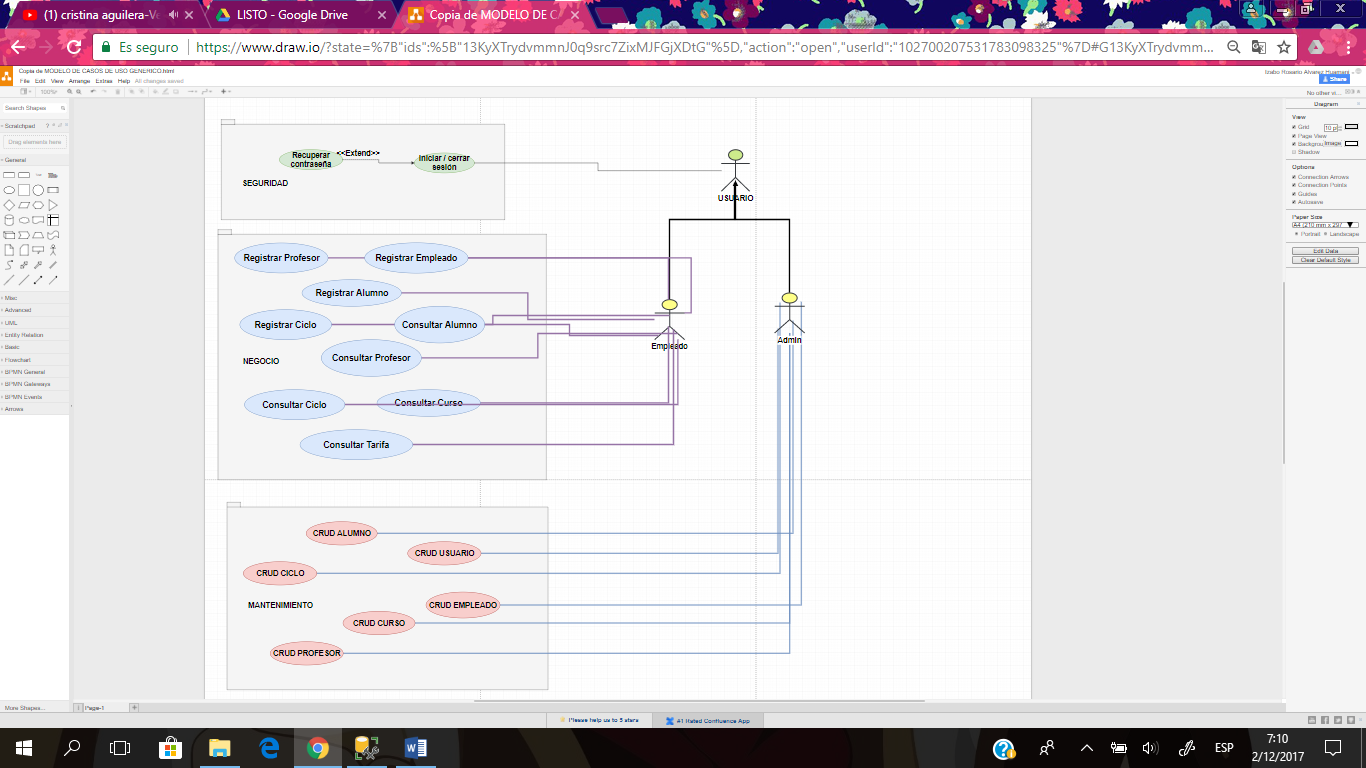


### **MODELO DE LA BASE DE DATOS**



## **ANÁLISIS**

### **CASOS DE USOS**



### **ANALISIS DE CASOS DE USOS**

#### **CONSULTAR ALUMNO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU02 Consultar alumno | | | | | **Nro. de Orden**: 02 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Realizará las consultas de los alumnos | | | | |
| Precondición | Se debe tener los datos de los alumnos | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | Se realizó las consultas correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de consulta de alumno | | | |
| 6 | El sistema le muestra la interfaz de consulta de alumno | | | |
| 7 | El administrador ingresa los datos del nuevo alumno | | | |
| 8 | El sistema le muestra las consultas | | | |

#### **REGISTRAR ALUMNO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU01 Registro alumno | | | | | **Nro. de Orden**: 01 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Se realizará el registro de nuevos alumnos | | | | |
| Precondición | Se debe tener los datos de los alumnos | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | El registro se hizo correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución. | | | |
| 2 | El sistema le muestra el logon. | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de registro de alumno | | | |
| 6 | El sistema le muestra la interfaz de registro de alumno | | | |
| 7 | El administrador ingresa los datos en la interfaz de registro de alumno | | | |
| **Excepciones** | Paso | Acción | | | |
| 7.1 | Cada alumno deberá tener un código distinto | | | |

#### **REGISTRAR TARIFA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU03 Registro de tarifa | | | | | **Nro. de Orden**: 03 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Registrará las tarifas de cada curso | | | | |
| Precondición | Se debe tener todos los precios de cada curso | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | El registro se hizo correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de registro de tarifas | | | |
| 6 | El sistema le muestra la interfaz de registro de tarifas | | | |
| 7 | El administrador ingresa los datos del nuevo alumno | | | |
| 8 | El sistema realiza el registro | | | |
| **Excepciones** | Paso | Acción | | | |
| 1 | Cada persona deberá tener un código distinto | | | |

#### **REGISTRAR PROFESOR**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU05 Registro de profesor | | | | | **Nro. de Orden**: 05 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Registrará a los profesores | | | | |
| Precondición | Se debe tener todos los datos de cada profesor | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | El registro se hizo correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de registro de profesor | | | |
| 6 | El sistema le muestra la interfaz de registro de profesor | | | |
| 7 | El administrador ingresa los datos del nuevo profesor | | | |
| 8 | El sistema realiza el registro | | | |
| **Excepciones** | Paso | Acción | | | |
| 1 | Cada persona deberá tener un código distinto | | | |

#### **REGISTRAR EMPLEADOS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU07 Registro de empleados | | | | | **Nro. de Orden**: 07 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Registrará a los empleados | | | | |
| Precondición | Se debe tener todos los datos de cada empleado | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | El registro se hizo correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Secundario | |  | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3  4 | El administrador ingresa sus datos de usuario en el logon  El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de registro de empleados | | | |
| 6 | El sistema le muestra la interfaz de registro de empleados | | | |
| 7 | El administrador ingresa los datos de los empleados | | | |
| 8 | El sistema registra correctamente | | | |
| **Excepciones** | Paso | Acción | | | |
| 1 | Cada persona deberá tener un código distinto | | | |

#### **CONSULTAR PROFESOR**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU06 Registro de profesor | | | | | **Nro. de Orden**: 06 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Realizará las consultas de los profesores | | | | |
| Precondición | Se debe tener todos los datos de cada profesor | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | La consulta se hizo correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Secundario | |  | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de consulta de profesor | | | |
| 6 | El sistema le muestra la interfaz de consulta de profesor | | | |
| 7 | El administrador ingresa los datos del nuevo profesor | | | |
| 8 | El sistema le muestra las consultas | | | |
| **Excepciones** | Paso | Acción | | | |
| 1 | Cada persona deberá tener un código distinto | | | |

#### **CONSULTAR CICLO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU10 consultar ciclo | | | | | **Nro. de Orden**: 10 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Realizará las consultas de los ciclos | | | | |
| Precondición | Se debe tener los datos almacenados en la base de datos | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | Se realizó la consulta correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Secundario | |  | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de consulta de ciclo | | | |
| 6 | El sistema le muestra la interfaz de consulta de ciclo | | | |
| 7 | El administrador ingresa los datos del ciclo | | | |
| 8 | El sistema le muestra las consultas | | | |

#### **CONSULTAR TARIFA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel del Use Case:** ☐ Negocio ☒ Sistema de Información | | | | | |
| **Nombre del Use Case**: CU04 Consultar tarifa | | | | | **Nro. de Orden**: 04 |
| **Prioridad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Complejidad**: ☒ Alta ☐ Media ☐ Baja | | | | | |
| **Tipo de Use Case: ☒ Concreto ☐ Abstracto** | | | | | |
| Descripción General | Se realizará las consultas de las tarifas | | | | |
| Precondición | Se debe tener las tarifas registradas | | | | |
| **Post Condiciones** | Condición de falla | | | No se ha obtenido todos los datos | |
| Condición de éxito | | | La consulta se hizo correctamente | |
| **Nivel** | Primario | | | | |
| **Actores** | Principal | | administrador | | |
| Trigger | | Cuando el administrador ingresa al sistema | | |
| **Flujo Básico - Descripción Detallada** | Paso | Acción | | | |
| 1 | El administrador ingresa al sistema de la institución | | | |
| 2 | El sistema le muestra el logon | | | |
| 3 | El administrador ingresa sus datos de usuario en el logon | | | |
| 4 | El sistema reconoce al usuario y muestra el menú principal | | | |
| 5 | El administrador selecciona la interfaz de consulta de tarifa | | | |
| 6 | El sistema le muestra la interfaz de consulta de alumno | | | |
|  | 7 | El administrador ingresa los datos del nuevo alumno | | | |
|  | 8 | El sistema le muestra las consultas | | | |

### **DIAGRAMA DE SECUENCIA**

DIAGRAMA DE SECUENCIA DE REGISTRO DE ALUMNO

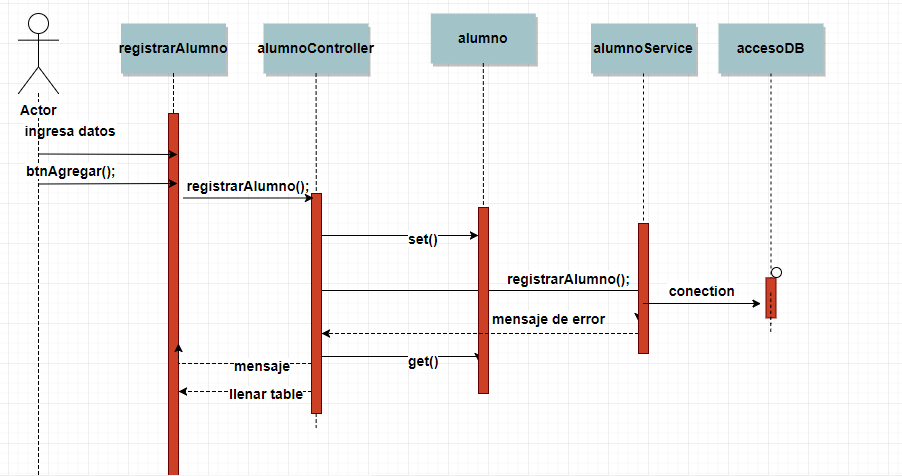


DIAGRAMA DE SECUENCIA DE REGISTRO DE CICLO

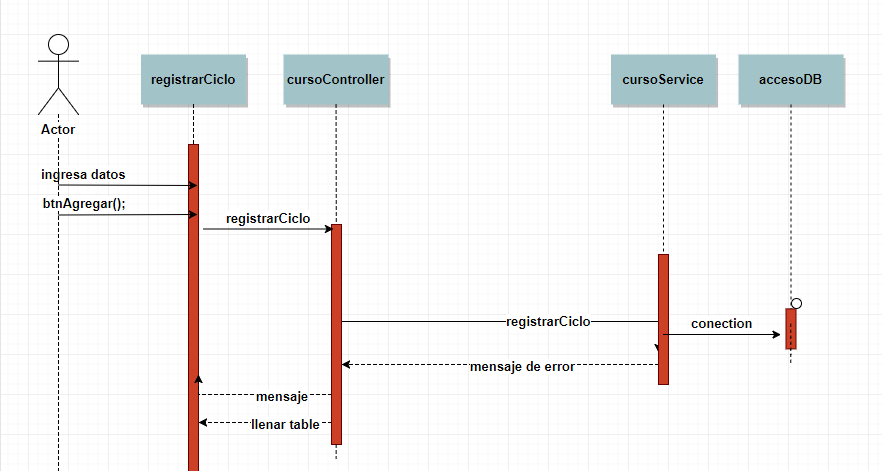


DIAGRAMA DE SECUENCIA DE REGISTRO DE EMPLEADO

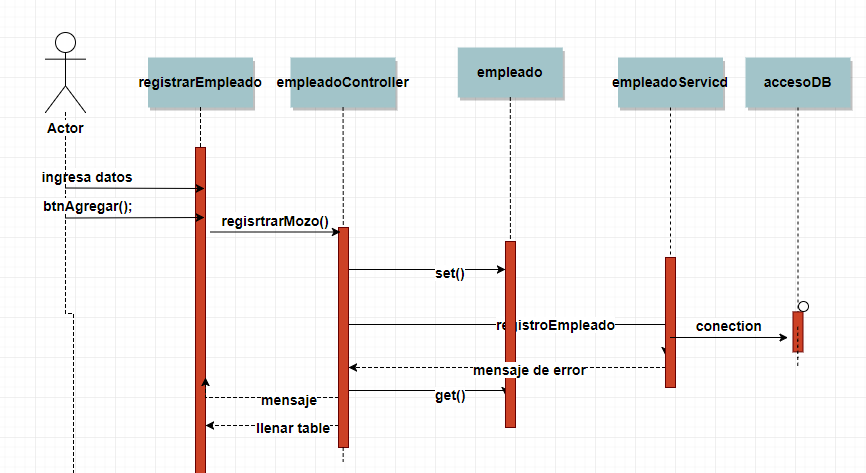


DIAGRAMA DE SECUENCIA DE REGISTRO DE PROFESOR

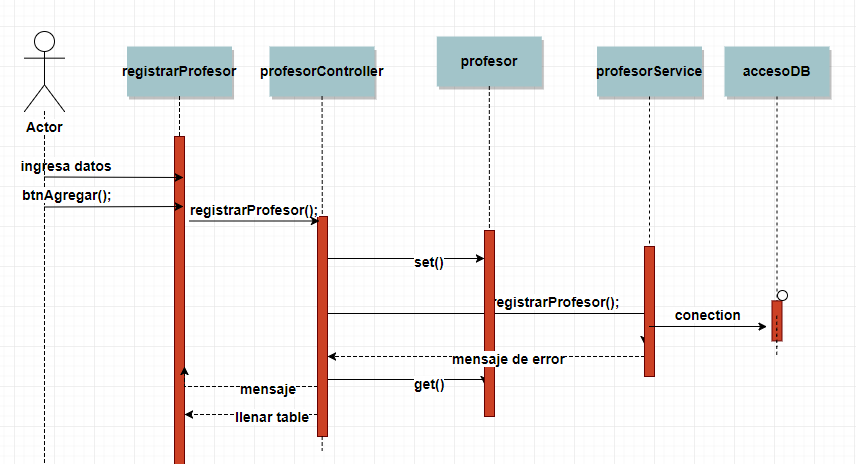
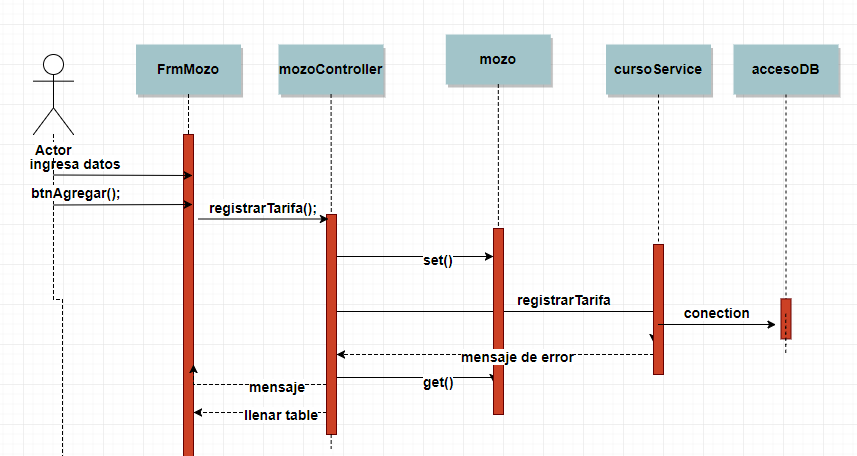
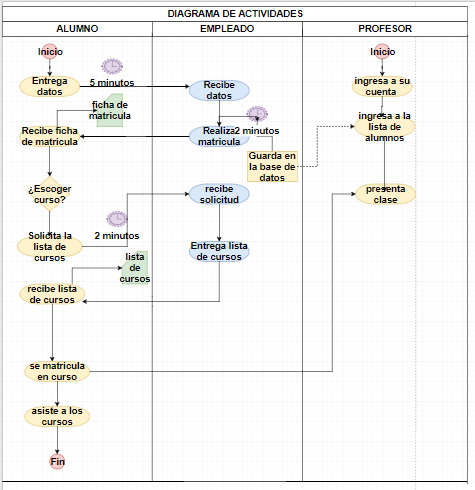


DIAGRAMA DE SECUENCIA DE REGISTRO DE TARIFA



### **DIAGRAMA DE ACTIVIDADES**



### **DIAGRAMA DE CLASES**

DIAGRAMA DE CLASES DE REGISTRO DE ALUMNO

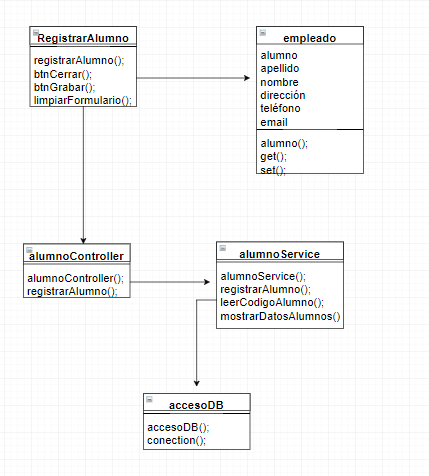


DIAGRAMA DE CLASES DE REGISTRO DE CICLO

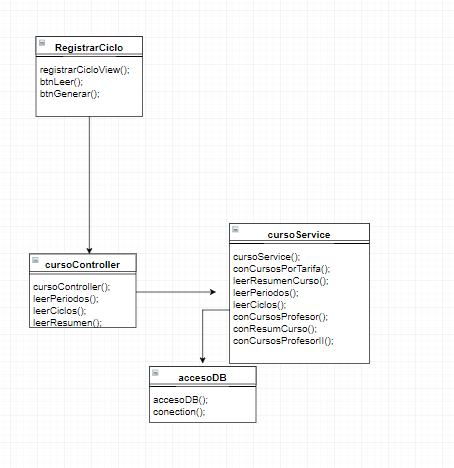


DIAGRAMA DE CLASES DE REGISTRO DE EMPLEADO

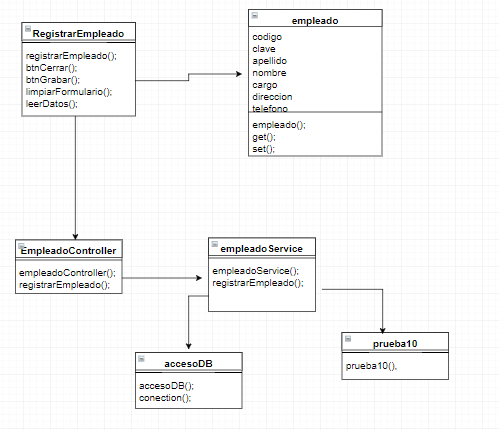


DIAGRAMA DE CLASES DE REGISTRO DE PROFESOR

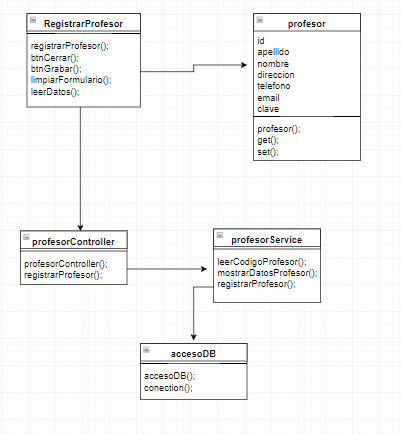
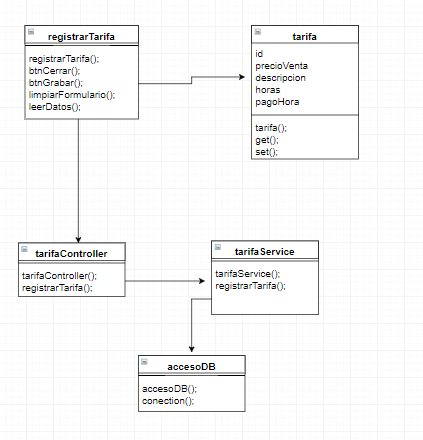
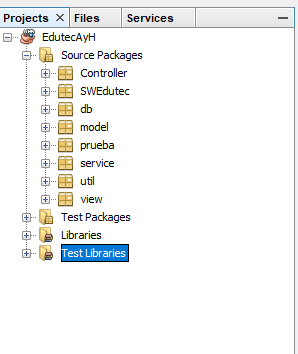


DIAGRAMA DE CLASES DE REGISTRO DE TARIFA

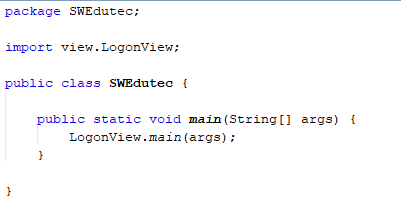


## **APLICACIÓN**

### **CODIGO JAVA**



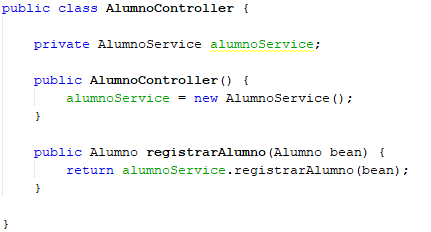
Empezaremos por el paquete “SWEdutec” con la clase MAIN PRINCIPAL “SWEdutec”

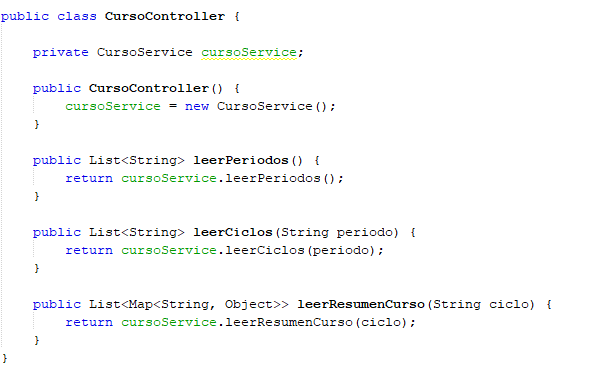
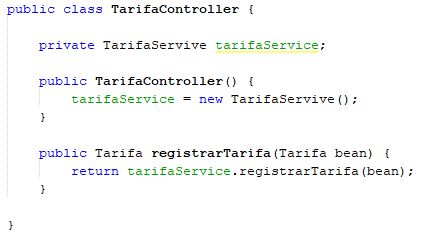


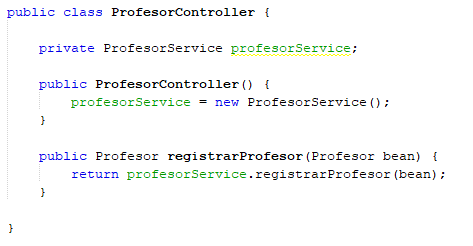
Al compilar, lo primero que ejecutará será el LOGON.

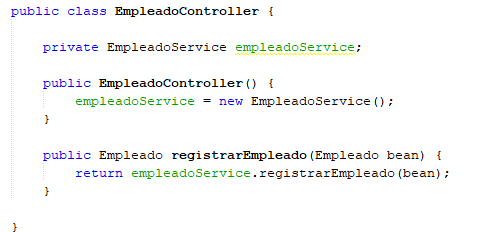
El segundo paquete será “DB” con la clase “AccesoBD”. Esta clase sirve para comunicarnos con la Base de Datos en SQL.

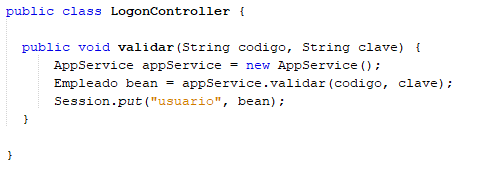
El paquete “CONTROLLER” contiene clases que nos sirven para comunicarnos y controlar los JFRAME con los SERVICIOS.



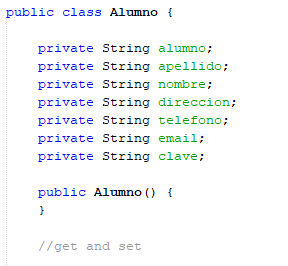


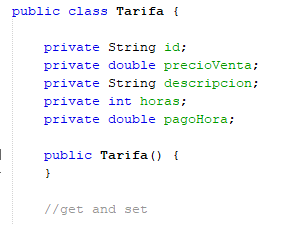


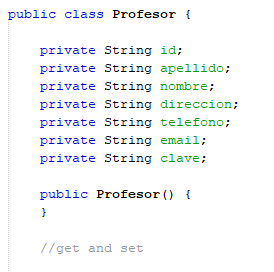


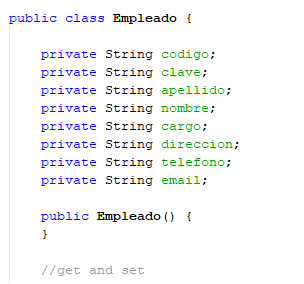


Continuamos con el Paquete Model.

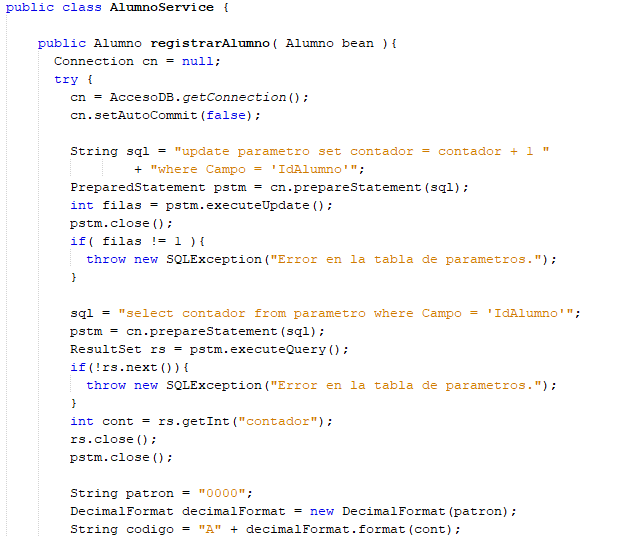


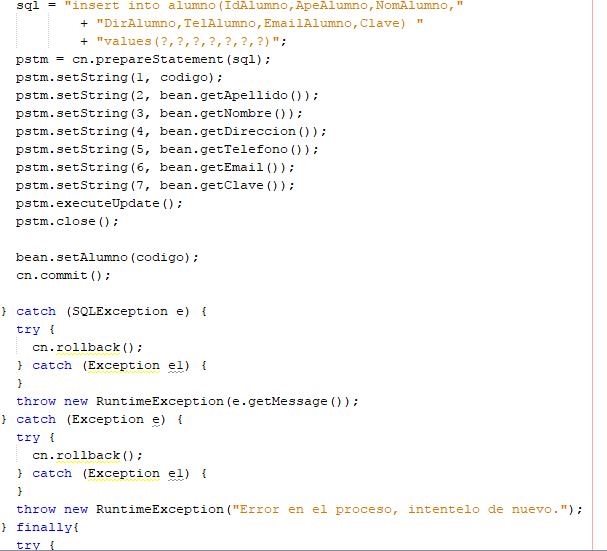


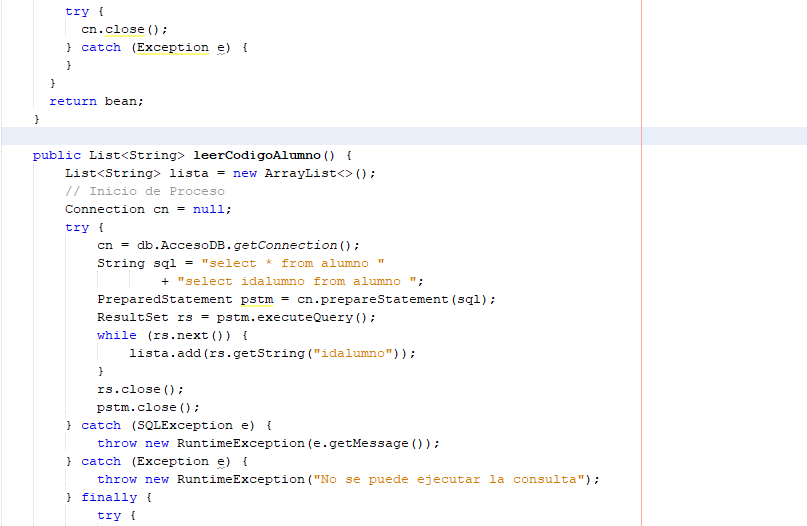


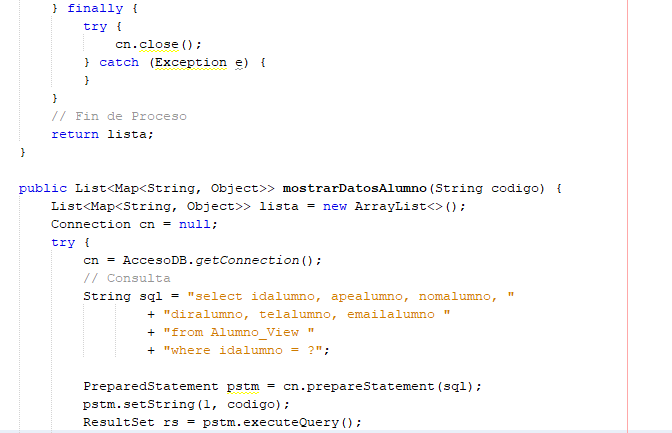


Luego tenemos el Paquete “SERVICE”, aquí se encuentran todas las clases Service.

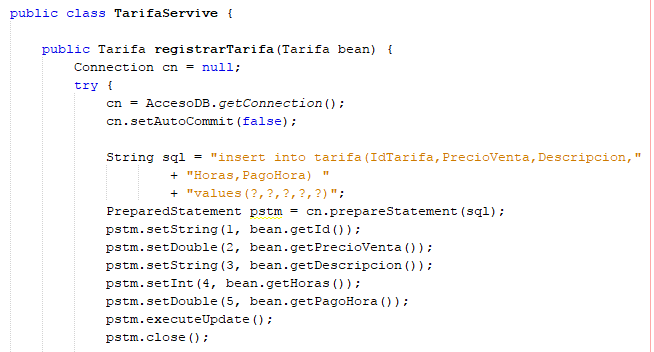


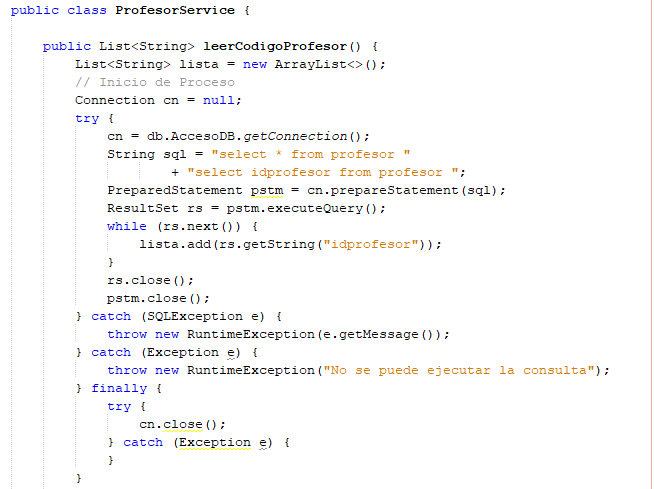


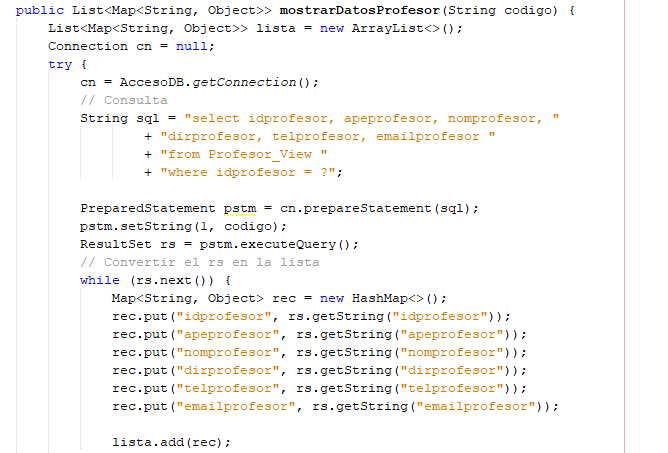


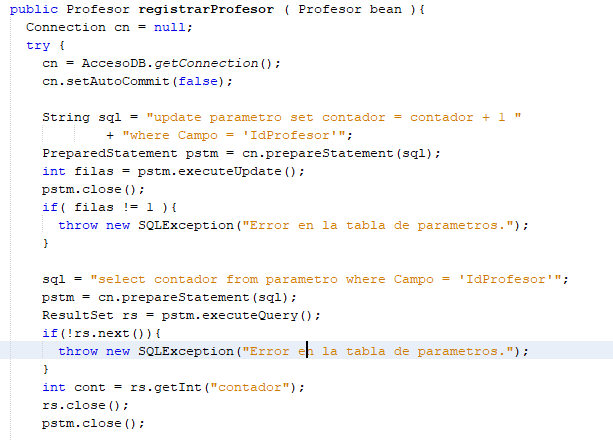


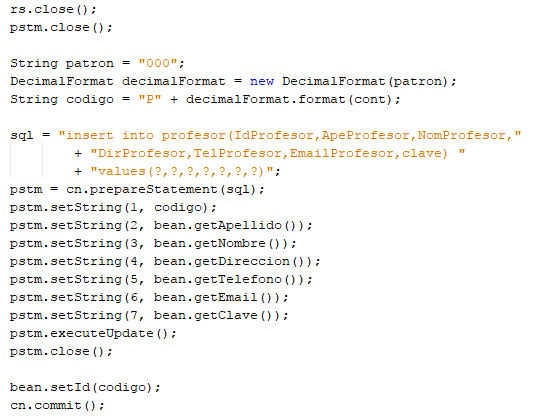




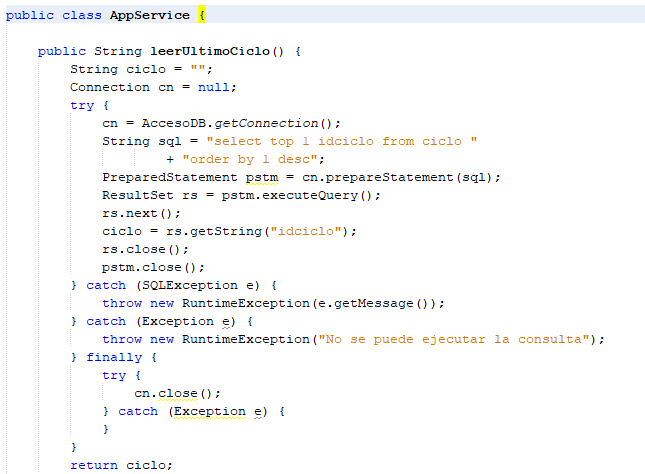


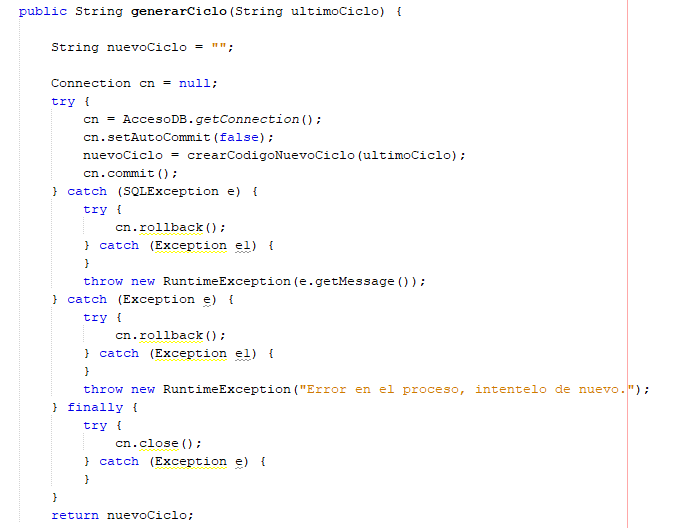


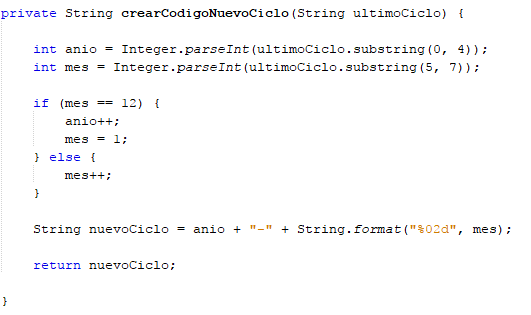


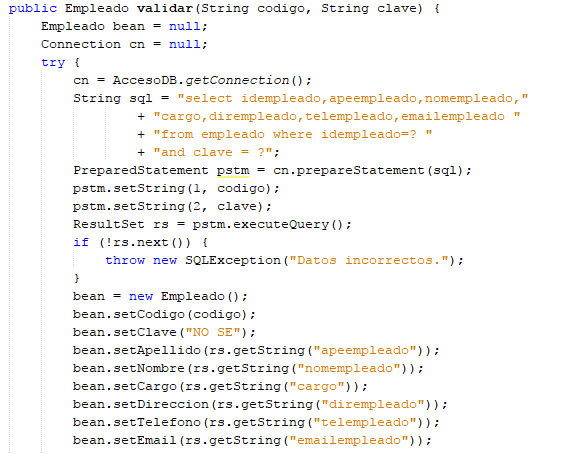




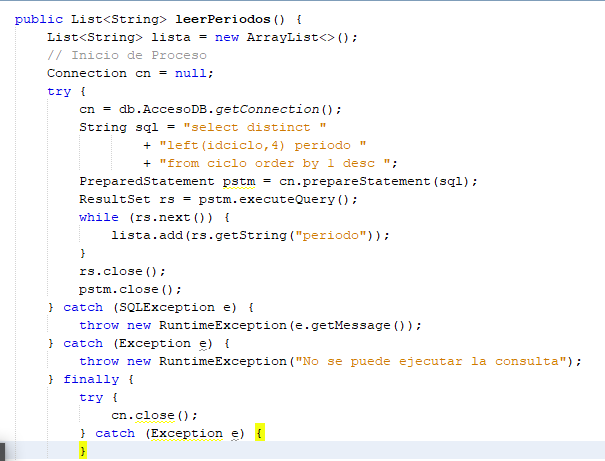
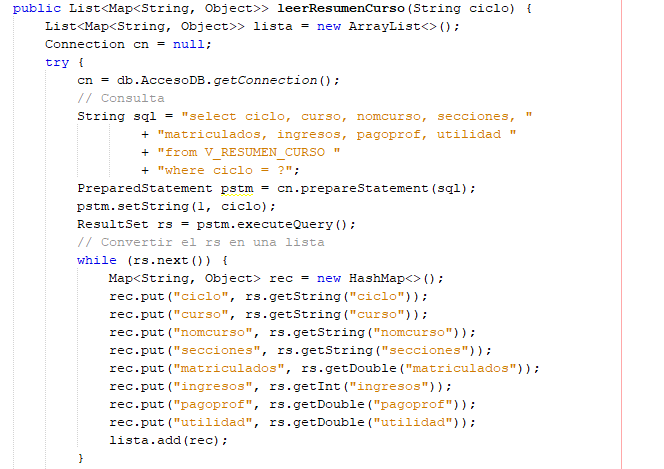


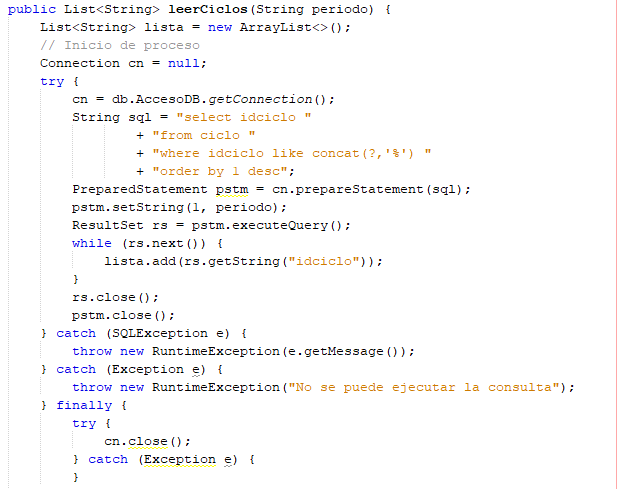


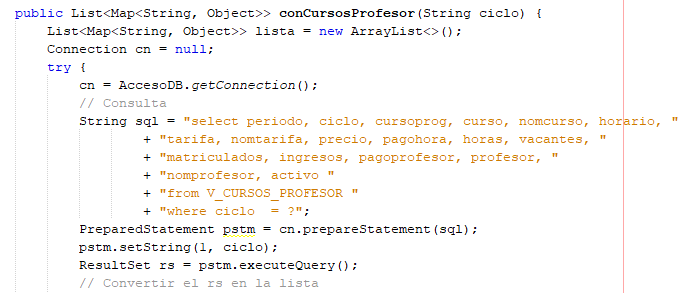


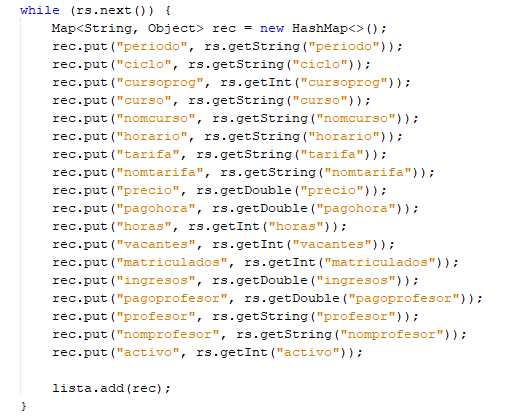


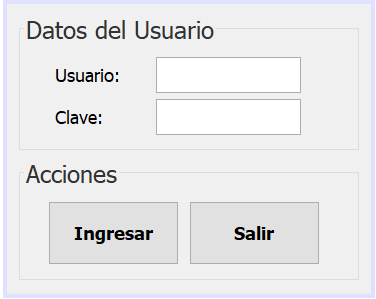




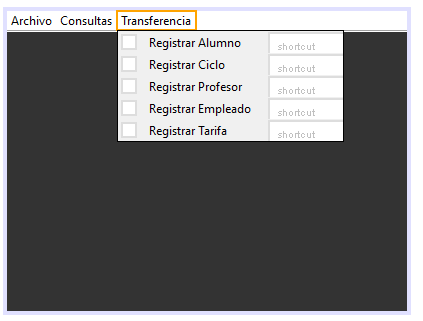
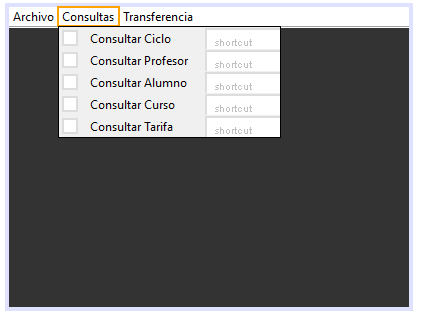


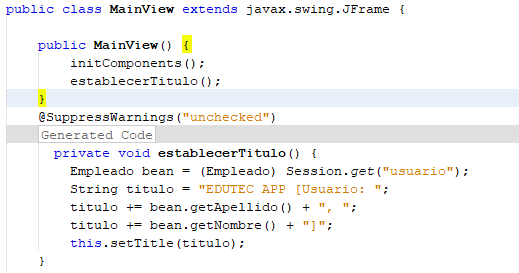


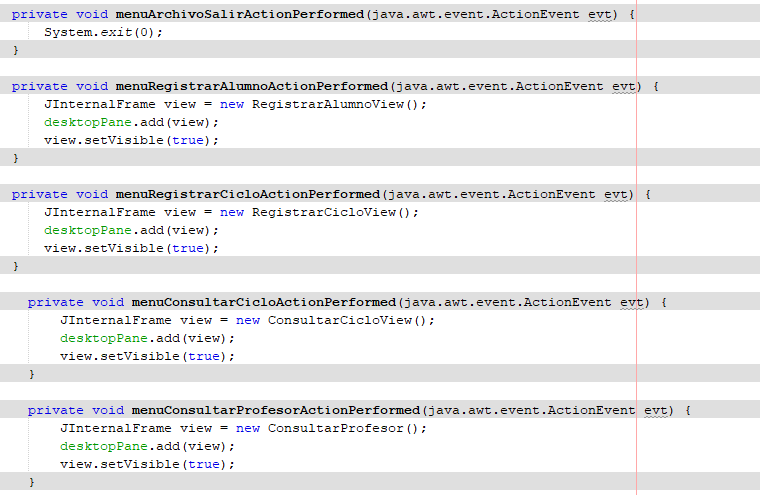


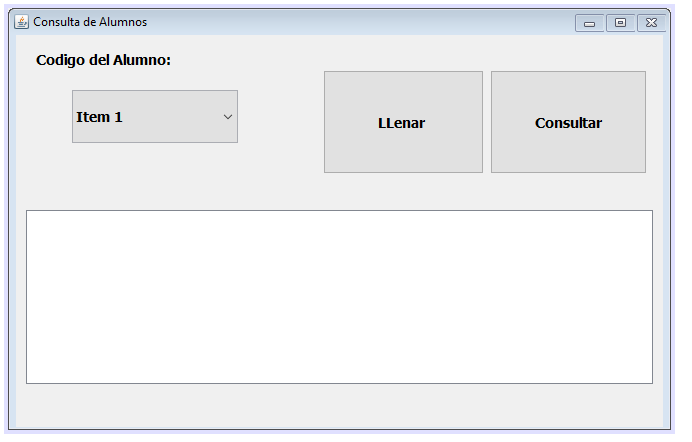


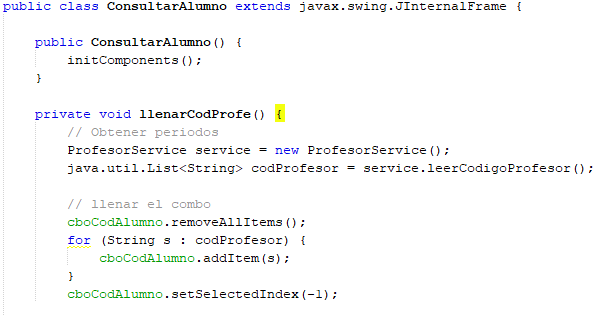


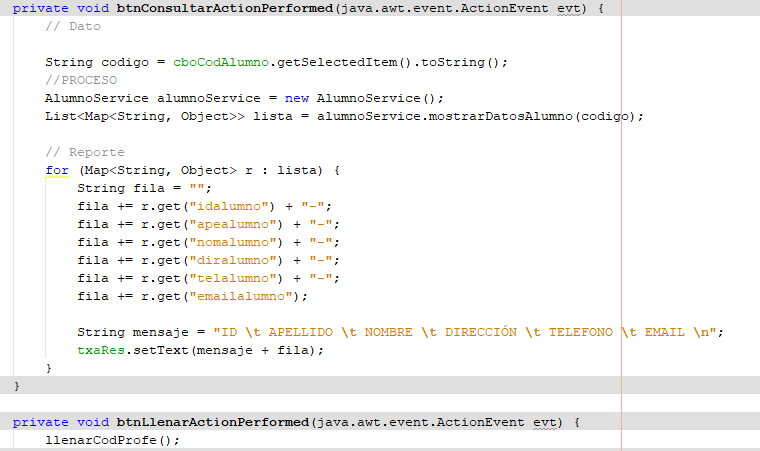


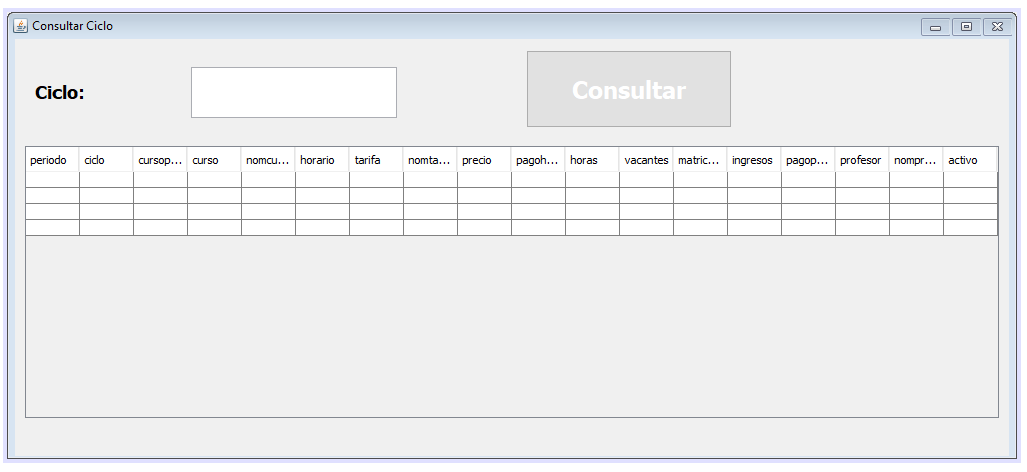




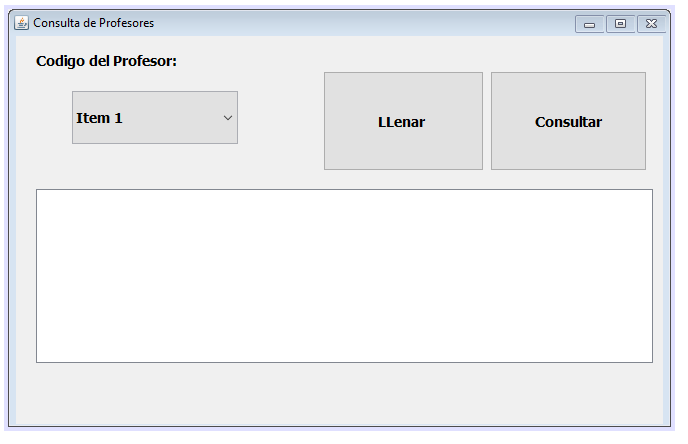




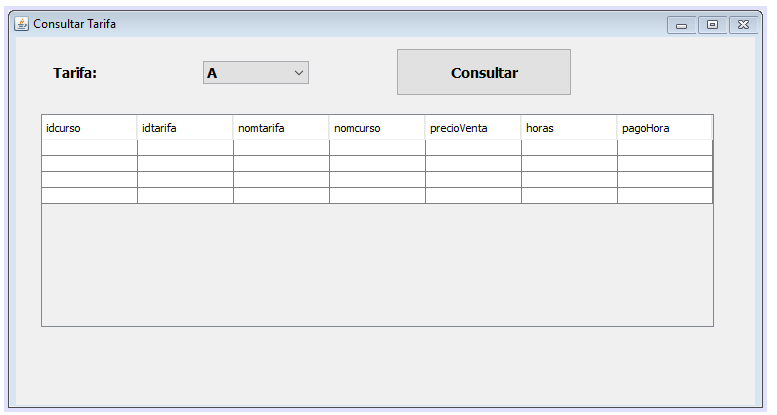




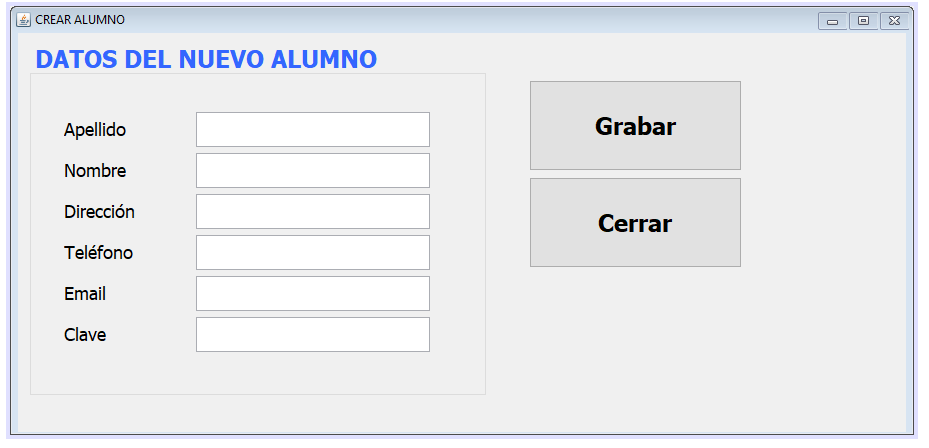


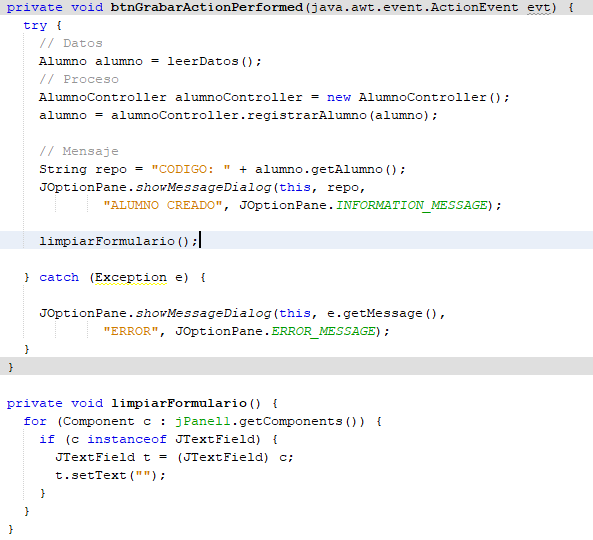


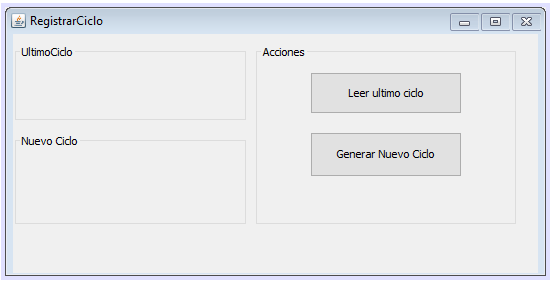




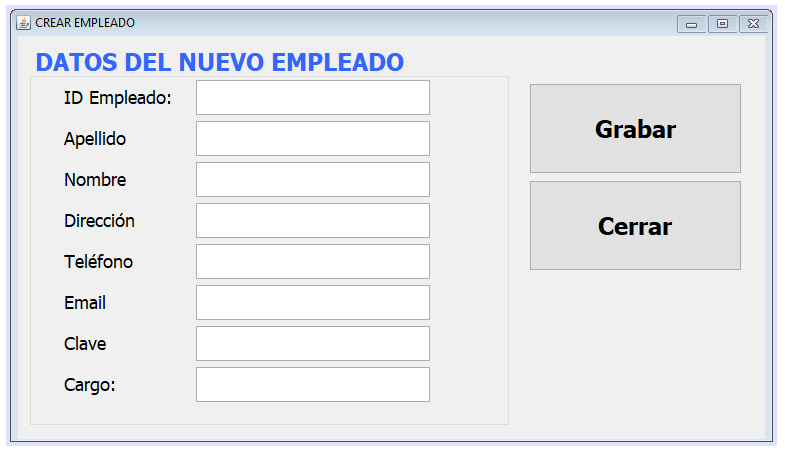




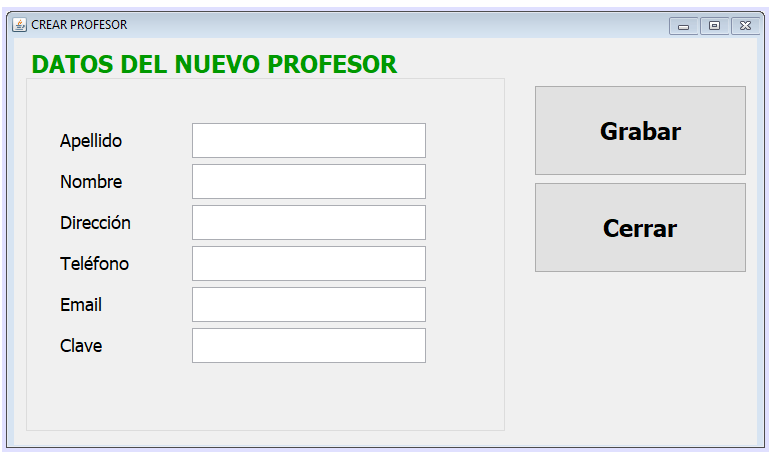


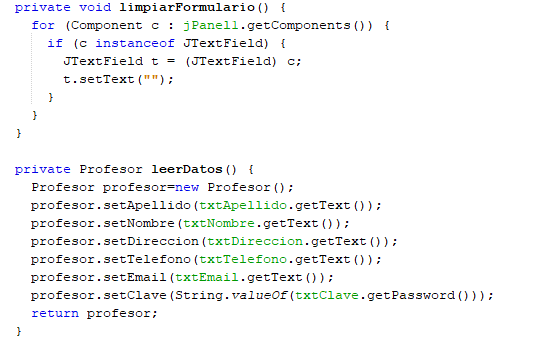


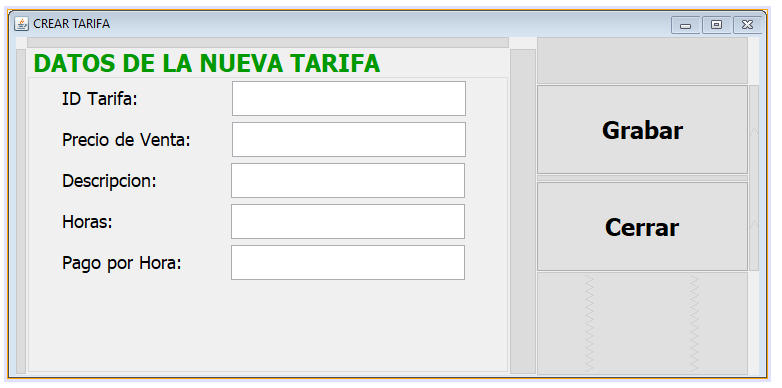


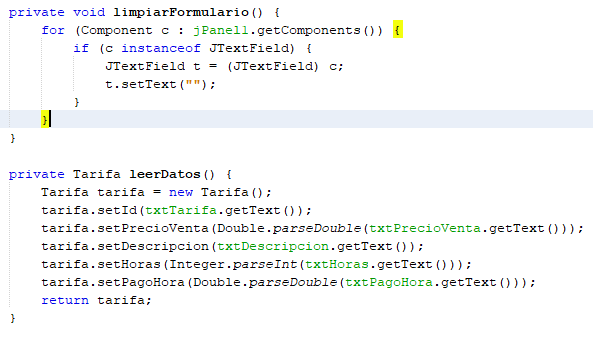












### **VISTAS EN SQL**

CREATE VIEW V\_RESUMEN\_CURSO(

PERIODO, CICLO, IDTARIFA, NOMTARIFA, IDCURSO, NOMCURSO, HORAS, SECCIONES,

VACDISP, VACTOTAL, MATRICULADOS, PRECIO, PAGOHORA, INGRESOS, PAGOPROF, UTILIDAD

) AS

WITH V\_PREVIA AS(

SELECT

LEFT(IDCICLO,4) PERIODO,

IDCICLO, IDCURSO,

COUNT(IDCURSOPROG) SECCIONES,

SUM(VACANTES + MATRICULADOS) VAC\_TOTAL,

SUM(VACANTES) VAC\_DISP,

SUM(MATRICULADOS) MATRICULADOS,

SUM(MATRICULADOS \* PRECURSOPROG) INGRESOS

FROM DBO.CURSOPROGRAMADO

WHERE ACTIVO = 1

GROUP BY IDCICLO, IDCURSO)

SELECT

V.PERIODO,V.IDCICLO, T.IDTARIFA, T.DESCRIPCION, C.IDCURSO, C.NOMCURSO,

T.HORAS, V.SECCIONES, V.VAC\_DISP, V.VAC\_TOTAL, V.MATRICULADOS,

T.PRECIOVENTA, T.PAGOHORA, V.INGRESOS,

(T.HORAS \* T.PAGOHORA \* V.SECCIONES) PAGOPROF,

(V.INGRESOS - (T.HORAS \* T.PAGOHORA \* V.SECCIONES)) UTILIDAD

FROM DBO.TARIFA T

JOIN DBO.CURSO C ON T.IDTARIFA = C.IDTARIFA

JOIN V\_PREVIA V ON C.IDCURSO = V.IDCURSO

JOIN DBO.CICLO CI ON V.IDCICLO = CI.IDCICLO;

GO

CREATE VIEW V\_RESUMEN\_CURSO(

PERIODO, CICLO, TARIFA, NOMTARIFA, CURSO, NOMCURSO,

HORAS, SECCIONES, VACTOTAL, VACDISP, MATRICULADOS,

PRECIO, PAGOHORA, INGRESOS, PAGOPROF, UTILIDAD

) AS

WITH V\_PREVIA AS(

SELECT

LEFT(IDCICLO,4) PERIODO,

IDCICLO, IDCURSO,

COUNT(IDCURSOPROG) SECCIONES,

SUM(VACANTES + MATRICULADOS) VAC\_TOTAL,

SUM(VACANTES) VAC\_DISP,

SUM(MATRICULADOS) MATRICULADOS,

SUM(MATRICULADOS \* PRECURSOPROG) INGRESOS

FROM DBO.CURSOPROGRAMADO

WHERE ACTIVO = 1

GROUP BY IDCICLO, IDCURSO)

SELECT

V.PERIODO,V.IDCICLO, T.IDTARIFA, T.DESCRIPCION, C.IDCURSO, C.NOMCURSO,

T.HORAS, V.SECCIONES, V.VAC\_TOTAL, V.VAC\_DISP, V.MATRICULADOS,

T.PRECIOVENTA, T.PAGOHORA, V.INGRESOS,

(T.HORAS \* T.PAGOHORA \* V.SECCIONES) PAGOPROF,

(V.INGRESOS - (T.HORAS \* T.PAGOHORA \* V.SECCIONES)) UTILIDAD

FROM DBO.TARIFA T

JOIN DBO.CURSO C ON T.IDTARIFA = C.IDTARIFA

JOIN V\_PREVIA V ON C.IDCURSO = V.IDCURSO

JOIN DBO.CICLO CI ON V.IDCICLO = CI.IDCICLO;

GO

CREATE VIEW V\_MEJOR\_CURSO(

PERIODO, CICLO, IDCURSO, NOMCURSO, SECCIONES, MATRICULADOS

)AS

WITH V\_PREVIA AS(

SELECT CICLO, MAX(MATRICULADOS) MATRICULADOS

FROM V\_RESUMEN\_CURSO

GROUP BY CICLO)

SELECT R.PERIODO, R.CICLO, R.IDCURSO, R.NOMCURSO, R.SECCIONES, R.MATRICULADOS

FROM V\_RESUMEN\_CURSO R

JOIN V\_PREVIA P

ON R.CICLO = P.CICLO AND R.MATRICULADOS = P.MATRICULADOS;

SELECT \* FROM V\_MEJOR\_CURSO

WHERE PERIODO = '2017'

ORDER BY CICLO, IDCURSO;

GO

CREATE VIEW V\_CURSOS\_PROFESOR(

PERIODO, CICLO, CURSOPROG, CURSO, NOMCURSO, HORARIO,

TARIFA, NOMTARIFA, PRECIO, PAGOHORA, HORAS, VACANTES,

MATRICULADOS, INGRESOS, PAGOPROFESOR, PROFESOR,

NOMPROFESOR, ACTIVO

) AS

SELECT LEFT(CP.IDCICLO,4), CP.IDCICLO, CP.IDCURSOPROG,

CP.IDCURSO, C.NOMCURSO, CP.HORARIO, C.IDTARIFA,

T.DESCRIPCION, T.PRECIOVENTA, T.PAGOHORA, T.HORAS,

CP.VACANTES, CP.MATRICULADOS, T.PRECIOVENTA \* CP.MATRICULADOS,

T.HORAS \* T.PAGOHORA, ISNULL(CP.IDPROFESOR,''),

ISNULL(P.APEPROFESOR + ', ' + P.NOMPROFESOR,''),

CP.ACTIVO

FROM TARIFA T

JOIN CURSO C ON T.IDTARIFA = C.IDTARIFA

JOIN CURSOPROGRAMADO CP ON C.IDCURSO = CP.IDCURSO

LEFT JOIN PROFESOR P ON CP.IDPROFESOR = P.IDPROFESOR

GO

SELECT COUNT(\*)

FROM CURSOPROGRAMADO

WHERE IDPROFESOR IS NULL;

GO

SELECT \* FROM V\_CURSOS\_PROFESOR

GO

CREATE VIEW V\_APROBADOS(

CICLO,IDCURSOPROG,CURSO,NOMBRE,HORARIO,PROFESOR,APROBADOS,DESAPROBADOS

) AS

WITH PREVIA AS (

SELECT

CP.IDCURSOPROG,

SUM(CASE WHEN M.PROMEDIO>=11 THEN 1 ELSE 0 END) APROBADOS,

SUM(CASE WHEN M.PROMEDIO<11 THEN 1 ELSE 0 END) DESAPROBADOS

FROM CURSOPROGRAMADO CP

JOIN MATRICULA M

ON CP.IDCURSOPROG = M.IDCURSOPROG

GROUP BY CP.IDCURSOPROG

)

SELECT

CP.IDCICLO, CP.IDCURSOPROG, CP.IDCURSO,

C.NOMCURSO, CP.HORARIO,

CASE

WHEN CP.IDPROFESOR IS NULL THEN 'NONE'

ELSE CP.IDPROFESOR + ' - ' + PR.APEPROFESOR + ' - ' + PR.NOMPROFESOR

END,

P.APROBADOS, P.DESAPROBADOS

FROM PREVIA P

JOIN CURSOPROGRAMADO CP ON P.IDCURSOPROG = CP.IDCURSOPROG

JOIN CURSO C ON CP.IDCURSO = C.IDCURSO

LEFT JOIN PROFESOR PR ON CP.IDPROFESOR = PR.IDPROFESOR

GO

SELECT \* FROM V\_APROBADOS WHERE IDCURSOPROG = 8183;

GO

SELECT \* FROM CURSOPROGRAMADO WHERE IDPROFESOR IS NULL;

CREATE VIEW PROFESOR\_VIEW AS

SELECT IDPROFESOR,APEPROFESOR,NOMPROFESOR,TELPROFESOR, DIRPROFESOR FROM DBO.PROFESOR

GO

SELECT \* FROM PROFESOR\_VIEW

SELECT IDPROFESOR FROM PROFESOR\_VIEW

CREATE VIEW V\_ADMIN

DELETE EMPLEADO

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES('PSALCE','PSALCE','SALCEDO BANDERAS','PERCY','GERENTE VENTAS','PSALCEDO@EDUTEC.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES('LCASTR','LCASTR','CASTRO ESCOBAR','LIDIA ROSA','SECRETARIA GENERAL','LCASTRO@EDUTEC.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES('JMANRR','JMANRR','MANRIQUE DIAZ','JOSÉ LUIS','COORDINADOR ACADEMICO','JMANRIQUE@EDUTEC.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES('JRAMIR','JRAMIR','RAMIREZ SALVADOR','JULIO CESAR','ADMINISTRADOR GENERAL','JRAMIREZ@EDUTEC.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES('ACAMPO','ACAMPO','CAMPOS HUAPAYA','ALBERTO','JEFE DE SEGURIDAD','ACAMPOS@EDUTEC.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES('BETETA','BETETA','BETETA BUSTAMANTE','CESAR','GERENTE GENERAL','CBETETA@EDUTEC.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES ('IALVAR','123456','ALVAREZ HUAMANI','IZABO ROSARIO','ADMIN','IALVAREZH@UCVVIRTUAL.COM.PE')

INSERT INTO EMPLEADO(IDEMPLEADO,CLAVE,APEEMPLEADO,NOMEMPLEADO,CARGO,EMAILEMPLEADO) VALUES ('AHUAMA','123456','HUAMAN FRANCIA','YASSEL ARACELY','ADMIN','ARACELYLEO15@GMAIL.COM')

GO

SELECT \* FROM EMPLEADO

CREATE VIEW ALUMNO\_VIEW AS

SELECT IDALUMNO,APEALUMNO,NOMALUMNO,DIRALUMNO,TELALUMNO,EMAILALUMNO FROM DBO.ALUMNO

GO

SELECT \* FROM ALUMNO\_VIEW

SELECT IDALUMNO FROM ALUMNO\_VIEW