

Aguascalientes, Ags. 16 de diciembre de 2019

Universidad Tecnológica de Aguascalientes
Tecnologías de la Información y Comunicación

Ingeniería de Software I

Prof. Roberto Correa Bravo

Documento final proyecto **CajeroUT**

Compilación de documentación generada



Karen Monserrat Rodríguez Fierro | Matrícula: **170785**

Eduardo Michael Rodríguez Medina | Matrícula: **171679**

Salvador Daniel Castillo Guerra | Matrícula: **162016**

Andrés Emanuel Reyes Jines | Matrícula: **140854**

Miguel Ángel Esparza soto | Matrícula: **170818**

Alejandro Salas Hernández | Matrícula: **170541**

Luis Alberto González | Matrícula: **171578**

José Luis Terán Fernández | Matrícula: **151447**

Grupo: 7E

Ficha del documento

Fecha	Revisión	Autores
22/10/2019	23/10/2019	Karen Montserrat Rodríguez Fierro
21/11/2019	22/11/2019	Eduardo Michael Rodríguez Medina
13/12/2019	16/12/2019	Luis Alberto González
		Andrés Emmanuel Reyes Jines
		Miguel Ángel Esparza Soto
		Alejandro Salas Hernández
		José Luis Terán Fernández
		Salvador Daniel Castillo Guerra

Documento validado por las partes en la fecha:

Por el profesor	Por la universidad
Roberto Correa Bravo	Gerardo Orduña Álvarez

Índice de Contenido

.....	1
Ficha del documento	2
Introducción al proyecto	7
Introducción al Estudio de Factibilidad	7
Propósito del Estudio de Factibilidad	7
Alcance del documento	7
Ámbito del sistema	8
Personal Involucrado	9
Definiciones, Acrónimos y abreviaturas	10
Referencias	12
Resumen	12
Estudio de Factibilidad	13
Análisis de la situación	13
Sistemas e infraestructura.	14
Aplicaciones y entornos de software	14
Factibilidad operativa	15
Factibilidad técnica	16
Herramientas de desarrollo	16
Conexión a internet	16
Servidor	16
Navegador web	17
Factibilidad económica	17
Factibilidad Legal	18
Ley Orgánica de Protección de Datos (LOPD)	19
Factibilidad Alternativa:	20
Conclusión:	20
Descripción General del Proyecto	21
Perspectiva del producto	21
Funcionalidad del producto	22

Características de los usuarios	29
Restricciones	29
Suposiciones y dependencias.....	29
Requisitos específicos	30
Requerimientos funcionales	30
Requerimientos no funcionales	38
Requisitos comunes de las interfaces	40
Interfaces de usuario.....	40
Interfaces de hardware	40
Interfaces de software	41
Interfaces de comunicación	41
Requerimientos funcionales	41
Rcon1: Calificaciones.....	41
Rcon2: Horario	41
Rcon3: Mapa	41
Rcon4: Estado de cuenta.....	41
Rtra1: Pre-Trámite de becas.....	42
Rtra2: Solicitud de beca.....	42
Rtra3: Solicitud de Constancia de estudios.....	42
Rtra4: Impresión de constancia de estudios.....	42
Rtra5: Solicitud de constancia de no adeudos	42
Rtra6: Impresión de constancia de no adeudos.....	42
Rserv1: Abono Crédito UT	42
Rserv2: Pago colegiatura	42
Rserv3: Pago examen	43
Rserv4: Pago Inscripción	43
Rserv5: Pago Titulación	43
Rserv6: Turno Ventanilla	43
Rserv7: Lo boletos	43
Rserv8: Servicios: “Evaluación servicios y docentes”	43
Rserv9: Servicios: “Buzón”	43

Rlob1: Email.....	44
Rlob2: Navegar	44
Rlob3: Contacto Soporte	44
Requerimientos no funcionales	44
Requisitos de rendimiento	44
Seguridad.....	44
Fiabilidad	45
Disponibilidad.....	45
Mantenibilidad	45
Portabilidad	45
Modelado de software	46
Diagramas de secuencia.....	46
Servicios: Caja (Pagos).....	46
Servicios: CreditoUT	46
Servicios: Loboletos.....	47
Servicios: Evaluación de servicios	47
Servicios: Buzón.....	47
Consultas: Calificaciones	48
Trámites: Becas	48
Trámites: Constancias	49
LobosConect:.....	49
Diagramas de colaboración.....	50
Servicios: Caja (Pagos).....	50
Servicios: CreditoUT	50
Servicios: Loboletos.....	51
Servicios: Evaluación de servicios	51
Servicios: Buzón.....	52
Consultas: Calificaciones	52
Trámites: Constancias	53
LobosConect:.....	53
Diagramas de máquina de estados	54

Servicios: Caja (Pagos)	54
Servicios: CreditoUT	54
Servicios: Loboletos.....	54
Servicios: Evaluación de servicios	55
Servicios: Buzón.....	55
Consultas: Calificaciones	55
Trámites: Constancias	55
LobosConect: Soporte	56
Código del sistema	56
Modelos.....	56
Vistas	85
Controladores.....	201

Introducción al proyecto

Este documento tiene el fin de ser una un reporte de la justificación y planeación de proyecto de desarrollo de software “**CajeroUT**”, este incluye un ensayo de la justificación, de la metodología que se eligió para el proyecto (*SCRUM*), El estudio de factibilidad, la Especificación de Requerimientos Software (ERS), toda la documentación generada durante el modelado del software, así como el código del proyecto. La parte de la especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para la especificación de requerimientos Software *ANSI/IEEE 830, 1998*.

Introducción al Estudio de Factibilidad

A continuación se presenta el texto íntegro del estudio de factibilidad realizado previo al inicio a la aceptación y desarrollo del proyecto.

Propósito del Estudio de Factibilidad

Mediante las siguientes líneas se busca establecer la viabilidad del proyecto, así como definir los requerimientos funcionales y no funcionales para el desarrollo de un sistema de información que permitirá gestionar distintos trámites administrativos y servicios académicos por parte de las y los alumnos de la Universidad Tecnológica de Aguascalientes, esto mediante al montaje del sistema en módulos parecidos a cajeros bancarios que se encontrarán dispersos en varios puntos estratégicos del campus.

Alcance del documento

El presente documento está dirigido a cualquier posible usuario-desarrollador del sistema, con el fin de dejar una base de conocimiento operativo y descriptivo del sistema CajeroUT, facilitando así que cualquiera pueda tomar lo desarrollado y continuarlo, de igual forma poder llevar acabo el mantenimiento del mismo.

Ámbito del sistema

Bautizado como **CajeroUT**; este sistema buscará ser una herramienta más a disposición de todas y todos los alumnos de la universidad con el fin de facilitar diferentes procesos estudiantiles y administrativos propios de la universidad, así como el representar una solución factible a la saturación de diferentes áreas administrativas en tiempos de mucha afluencia de estudiantes y la necesidad del alumnado de contar con un equipo que se conecte a internet para realizar diferentes consultas.

En los propósitos generales del sistema se estipula que este deberá cubrir varias de las actividades de consulta básicas que se dan entre alumnado y la propia universidad, así como el brindar herramientas que ayuden tanto a administrativos como estudiantes realizar acciones de consulta sobre diferentes temas.

Los diferentes Cajeros serán los módulos que facilitaran el acceso al sistema se plantea, funcione mediante una conexión Wireless (Wifi) a una intranet propia de la universidad y contando también con salida a internet.

La información y datos que presentará el sistema se obtendrán de bases de datos ya existentes como las usadas en cajas o para el Sistema de Información Táctico Operativo (*SITO*), por lo que su manejo administrativo será desde estas aplicaciones y directamente desde las bases de datos (Actualización y manipulación de datos).

Se plantea que la distribución de los Cajeros sea de uno al interior de cada aula, para que así las y los alumnos no estén a más de unos pasos desde sus salones a los mismos.

Personal Involucrado

Nombre	Eduardo Michael Rodríguez Medina
Rol	<u>Líder de proyecto</u> , Scrum master, analista y programador Backend (Modelos)
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Control de actividades Cruz, dirección del equipo, analista de procesos y coordinar desarrollo del modelo
Información de contacto	171679@utags.edu.mx

Nombre	Karen Montserrat Rodríguez Fierro
Rol	<u>Líder de equipo y gerente</u> , Documentación, analista y programadora Frontend (Vistas)
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Coordinación del equipo, desarrollo y resguardo de documentación y entregables, control de bitácoras, analista de procesos y coordinar desarrollo de vistas
Información de contacto	170785@utags.edu.mx

Nombre	Andrés Emmanuel Reyes Jines
Rol	Analista y programador Backend (Controlador)
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Analista de procesos y coordinar desarrollo de controlador
Información de contacto	140854@utags.edu.mx

Nombre	Luis Alberto González
Rol	Administrador de base de datos (DBA)
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Diseño, desarrollo y administración de bases de datos
Información de contacto	171578@utags.edu.mx

Nombre	Miguel Ángel Esparza Soto
Rol	Programador y tester
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Programación por módulos y pruebas a programas compilados (versiones beta)
Información de contacto	170818@utags.edu.mx

Nombre	Alejandro Salas Hernández
Rol	Programador y tester
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Programación por módulos y pruebas a programas compilados (versiones beta)
Información de contacto	170541@utags.edu.mx

Nombre	José Luis Terán Fernández
Rol	Programador
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Programación por módulos
Información de contacto	151447@utags.edu.mx

Nombre	Salvador Daniel Castillo Guerra
Rol	Programador
Categoría profesional	Pasante de TSU en TICS
Responsabilidad	Programación por módulos
Información de contacto	162016@utags.edu.mx

Definiciones, Acrónimos y abreviaturas

Nombre	Descripción
Cajeros	Módulos físicos.
CajeroUT	Sistema instalado de forma local en los cajeros.
Usuario	Persona que hará uso del sistema.
Administrativos	Personal propio de la universidad que podrán realizar diversas consultas en el sistema CajeroUT .
Base de datos	Conjunto de datos digitales almacenados en tablas dentro de un servidor.
Motor de base de datos	Sistema que permite la gestión de las bases de datos.
Intranet	Servicio de red privada ala interior de la universidad.

Lenguaje de programación	Lenguaje formal que proporcionan un conjunto de órdenes y comandos capaces de ser interpretados por las máquinas.
GUI	Siglas en ingles de Graphic Usar Interface, se refiere a las pantallas con las que interactúa el usuario
Modulo	Sección o apartado del sistema
Redituó	Crédito digital para estudiantes que permite abonos y adeudos a ser liquidados al fin de cada cuatrimestre.
Lo boletos	boletos digitales que permiten hacer uso de servicios de la universidad cómo impresiones, copias, autobús, etc.
Wireless	Conexiones a internet sin cables (Wifi).

Referencias

IEEE Standards Association. (1998). IEEE 830-1998 - IEEE Recommended Practice for Software Requirements Specifications. Octubre 19 2019, de IEEE Standards Association. Sitio web: <https://standards.ieee.org/standard/830-1998.html>

Manual de identidad. Universidad Tecnológica de Aguascalientes. Sitio Web: <http://www.utags.edu.mx/>

Resumen

El presente apartado relativo al Estudio de factibilidad consta de cuatro secciones. En la primera sección se realiza una introducción al mismo y se proporciona una visión general de lo que concierne al sistema.

En la segunda sección se realiza una descripción del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo, sin entrar en excesivos detalles.

En la tercera se establece la factibilidad del desarrollo del sistema y por último, la cuarta sección del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema.

Estudio de Factibilidad

Título del proyecto: Plataforma en módulos para consultas y pagos de la Universidad Tecnológica de Aguascalientes (CajeroUT).

Problemática: En la Universidad Tecnológica de Aguascalientes es muy difícil realizar ciertas consultas y pagos, así como empezar trámites, ya que la única forma es presencial, se tiene que pasar a cajas y hacer fila solo en determinadas horas. En estos procesos, los estudiantes pierden tiempo y clases para realizarlos, además de no tener un horario que permita ser aprovechado para esto. De igual forma se identifica que las personas requieren de conectarse vía internet a la página de la universidad o bien a sistemas web de la misma como los es el *SITO* para realizar consultas o gestionar trámites, por lo que es necesario que las y los interesados cuenten con el servicio de conexión a internet y un dispositivo o computadora. Debido a esas dificultades surgió la idea de realizar una plataforma que permita optimizar diferentes procesos propios de la universidad, así como realizar algunos pagos de servicios internos de la institución.

Justificación: Los alumnos necesitan hacer uso de los servicios proporcionados por la universidad. Este proyecto se enfoca en: Información general de la Universidad (Mapa y calendario), información propia de las y los alumnos (Calificaciones, estados de cuenta y horario), compra de boletos para transporte, comida, impresiones y fotocopiado.

La plataforma agilizará el consumo de dichos servicios, lo que resultará en un ahorro de tiempo significativo para los alumnos. Esto permitirá la mejorar la eficiencia en el aprovechamiento de estos servicios y evitará que los alumnos desatiendan sus labores estudiantiles, así mismo, permitirá el acceso permanente al pago de transporte y diferentes servicios.

La plataforma se desarrollará pensando en su ejecución en un entorno local, haciendo uso de diferentes tecnologías como servicios de base de datos (procedimientos almacenados), el lenguaje de programación será java basados en un desarrollo Modelo, Vista y Controlador (MVC).

Análisis de la situación

Objetivo: Desarrollar una sistema que permita a los alumnos que por alguna razón no tengan acceso las diferentes plataformas y/o espacios para la realización de consultas, trámites y pagos acceder a los mismos a través de cómodos módulos tipo cajero multi-trámite que deberían estar a la mano en las áreas comunes de cada aula.

Sistemas e infraestructura.

Infraestructura:

Se requiere que cada módulo multi-trámite con los siguientes requerimientos mínimos:

Procesador de 2.5 GHz mínimo

16 GB de memoria RAM.

128 GB de almacenamiento.

Sistema operativo Windows 7 o superior.

JDK de java instalado.

Aplicaciones y entornos de software

A continuación, se enlista el software a utilizar separado por uso y seguido de su costo por periodo:

Desarrollo

Licencias JDK Java librerías + IDE (Gratis)

MySQL Workbench (Gratis)

Sublime Text 3 (Gratis)

PayPal API (comisiones variables)

Planeación y organización

Slack (Gratis)

Github (Gratis)

Google Hangouts (Gratis)

WhatsApp (Gratis)

Factibilidad operativa

Se ha visto mediante la matriz de estudio de problemas/oportunidades que las dificultades que se van a remediar mediante el sistema son de gran prioridad y urgencia, debido a que los procesos son realizados diariamente y de manera no equitativa.

Por este motivo, la creación de un sistema que automatice en gran medida estos procesos entre alumnos, administrativos y la propia institución, así mismo esperamos que este sistema resuelva los actuales problemas detectados, de nuestra parte creemos que se ha obtenido la información necesaria mediante la investigación realizada, con lo que se creará un sistema que logre llenar las expectativas y necesidades de los posibles usuarios y de la universidad.

En cuanto a información, control y su eficacia se tiene como objetivo trabajar mutuamente con las áreas administrativas y operativas de la universidad, así como con el propio alumnado, para que ellos vayan describiendo cuáles serán los requerimientos en el sistema y de esta manera desarrollar un sistema en donde se tendrá la completa satisfacción de los que lo van a utilizar.

Con la poca sociabilización que hemos hecho del proyecto, vemos que la idea tiene gran aceptación entre los que lo van a utilizar, quienes piensan que el sistema es ideal para sus mejor las diferentes áreas de oportunidad que existen en diferentes procesos y que el sistema resulta propositivo y verdaderamente funcional.

En este punto, vemos la existencia de la constante necesidad de agilizar los trámites internos que realizamos como cuerpo estudiantil y creemos que el desarrollo de esta plataforma y posible implementación en módulos podría resultar beneficioso al resolver problemas de filas y pérdida de tiempo en procesos de pago, consultas, impresión, etc. Así como ofrecer soluciones y herramientas que consideramos resultaran útiles.

Factibilidad técnica

A continuación, se enlistan los elementos necesarios para el desarrollo y funcionamiento de la plataforma.

Herramientas de desarrollo

Para desarrollar: Se necesita del JDK de java, NETBeans como IDE instalado y funcionando en todas las máquinas, así como de MySQL y PHP MyAdmin, todas son herramientas gratuitas con las que ya se cuentan.

Para funcionar: JDK de java instalado y Windows 7 o superior.

Conexión a internet

Para desarrollar: Se necesita de una conexión estable de al menos 25MB.

Para funcionar: Se necesita conexión Wireless a la intranet de la universidad de alta velocidad (25MB mínimo).

Servidor

Para desarrollar: No es necesaria la renta de un servidor web para desarrollar la plataforma.

Para funcionar: Se requiere acceso a las bases de datos y tablas de diferentes sistemas ya en uso para poder sacar de ahí la información y datos con los que la aplicación trabajará.

Navegador web

Para desarrollar y funcionar: Se necesita un navegador moderno (motores Gecko, V8, EdgeHTML, Blink) en su última versión. Se recomienda uso del navegador Google Chrome o Firefox.

Para funcionar: Se requiere que los módulos **CajerosUT** cuenten con un navegador instalado, se recomienda Chrome o Firefox.

En términos generales podríamos decir que si bien ya se tienen todas las herramientas identificadas cómo necesarias, vemos también que por el momento no se cuentan con los conocimientos necesarios lo suficientemente desarrollados que se requieren para este proyecto. Sin embargo, en el tiempo estimado para su desarrollo, pruebas e implementación, consideró que estamos a tiempo para afinar este punto.

Factibilidad económica

El análisis económico nos permitió ver los gastos generales del proyecto y cuáles de estos caen sobre el equipo de desarrollo.

Hardware

Comprar módulos tipo cajero multi-trámite (touchscreen) **\$14,000**

Servidores de base de datos **Gratis (Facilitados por la universidad)**

Equipo propio **Gratis**

Recursos materiales

Gastos de papelería \$20,000

Gastos en computación \$16,000

Gasto único total: \$50,000

Salarios (Gasto mensual)

Líder del Proyecto \$18,000

Analista/Diseñador \$16,000

DBA \$16,000

Testers \$14,000 x2

Programador \$12,000 x3

Total mensual: \$114,000

Total de gastos para el equipo desarrollador en los 3 meses propuestos para el desarrollo del sistema: **\$342,000 pesos.**

Total de inversión durante los 3 meses de desarrollo: \$392,000

Factibilidad Legal

Nuestro acuerdo de licencia será, en esencia, el mismo que usa Atlassian, Inc. con su servicio web de almacenamiento de repositorios BitBucket: <https://es.atlassian.com/legal/product-specific-terms>

De igual forma se recomienda adquirir las licencias para el software a emplearse en la implantación de un sistema informático de manera auténtica, con la finalidad de no tener inconvenientes legales a futuro.

Ley Orgánica de Protección de Datos (LOPD)

Es una ley que tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor, intimidad y privacidad personal y familiar.

Su objetivo principal es regular el tratamiento de los datos y ficheros, de carácter personal, independientemente del soporte en el cual sean tratados, los derechos de los ciudadanos sobre ellos y las obligaciones de aquellos que los crean o tratan.

Estructura

La Ley comprende un total de 49 artículos divididos en 7 Títulos y finaliza con una serie de disposiciones. Su estructura es la siguiente:

Título I. Disposiciones Generales.

Título II. Principios de la Protección de Datos.

Título III. Derechos de las Personas.

Título IV. Disposiciones Sectoriales.

Capítulo I. Ficheros de Titularidad Pública.

Capítulo II. Ficheros de Titularidad Privada.

Título V. Movimiento Internacional de Datos.

Título VI. Agencia Española de Protección de Datos.

Título VII. Infracciones y Sanciones.

6 Disposiciones Adicionales.

3 Disposiciones Transitorias.

3 Disposiciones Finales.

Factibilidad Alternativa:

De momento no se encuentran otras posibles barreras que impidan la factibilidad del proyecto.

Conclusión:

Conscientes de las necesidades actuales y de que estas, si recibimos el apoyo esperado de la universidad serán fácilmente cubiertas, **consideramos que el proyecto es factible y su desarrollo es viable**, puesto que las necesidades identificadas sobretodo en la factibilidad técnica se pueden adquirir durante el tiempo previsto de tres meses para el desarrollo del propio proyecto y al menos por el momento tras exponer costos y necesidades hemos recibido aprobación de la institución para empezar el desarrollo.

Descripción General del Proyecto

Perspectiva del producto

CajeroUT será un sistema local montado sobre módulos tipo cajero multi-trámite, los cuales estarán dispersos por todo el campus de la Universidad Tecnológica de Aguascalientes. Funcionaran como una interfaz sencilla y práctica dónde alumnos y visitantes podrán realizar consultas relativas a la propia institución y solo para el caso del alumnado, se podrán realizar diversos pagos mediante distintas formas.

El sistema se alimentará de otros ya existentes cómo el sistema de cajas, el de calificaciones y el Sistema de Información Táctico Operativo (SITO), así como de las bases de datos en producción de las cuales la universidad es dueña. Por ello se entiende a CajeroUT solo cómo una interfaz cuyo fin es mostrar de forma sencilla la información capturada y administrada desde otra plataforma y buscando la practicidad para la gestión de diferentes necesidades de los usuarios.

Una de las propuestas de valor que se ponen sobre la mesa es la de agregar el concepto de pago denominado “Redituó”, el cual planteamos será una cuenta de prepago para funcionar en la compra de servicios de la institución: Transporte, Tienda, Cafetería, Servicio de impresión y fotocopiado y pagos de colegiatura, exámenes, adeudos, etc. Así como la asignación de becas o recursos económicos asignados por viáticos, etc. En caso de que el alumno genere adeudos y este no llegue a liquidar durante el cuatrimestre se realizará un cobro directo a esta cuenta.

Redituó será un servicio exclusivo para estudiantes de la Universidad Tecnológica de Aguascalientes, se asignará una cuenta de prepago por alumno.

Funcionalidad del producto

Diagrama de casos de uso: Alumno

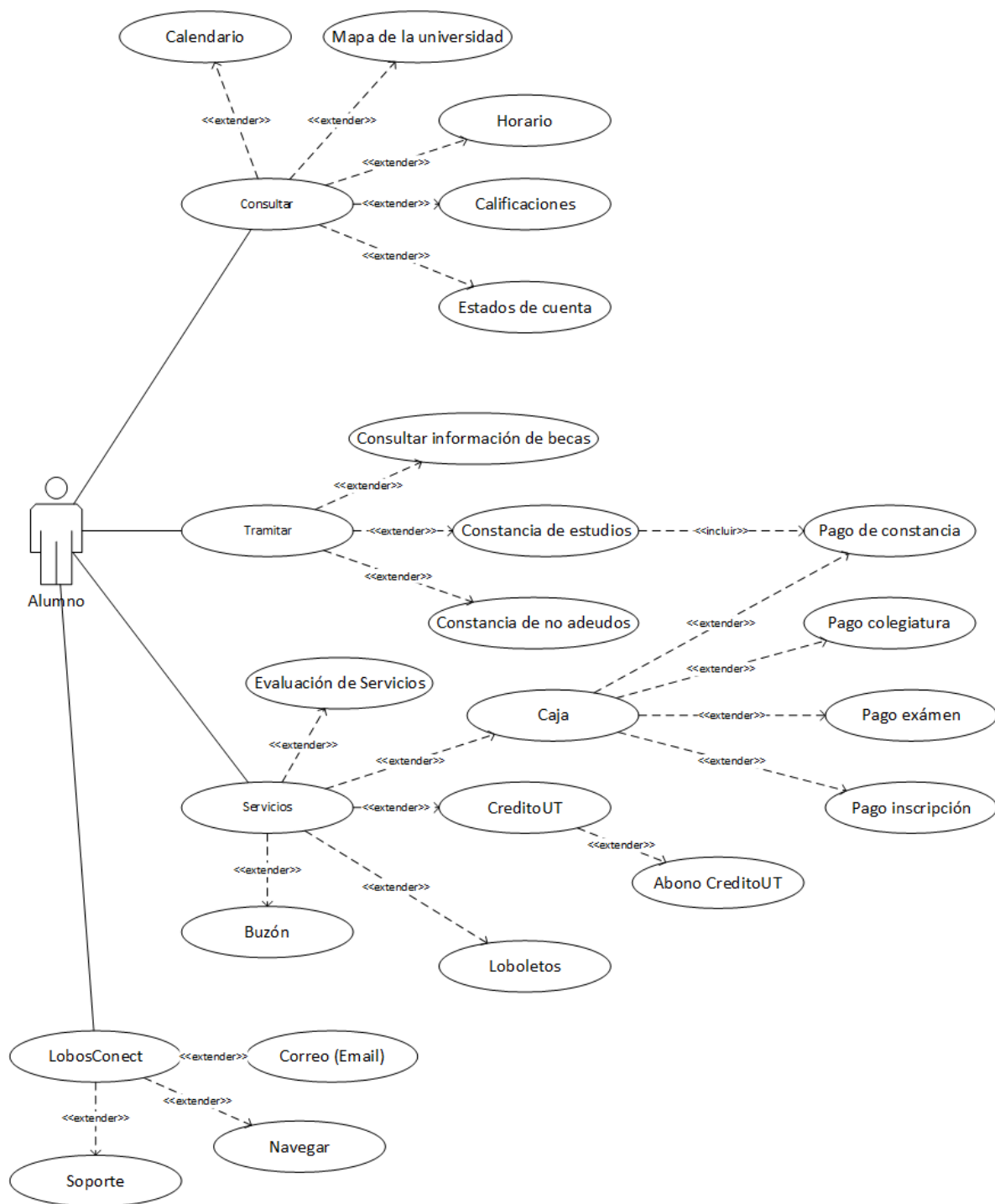
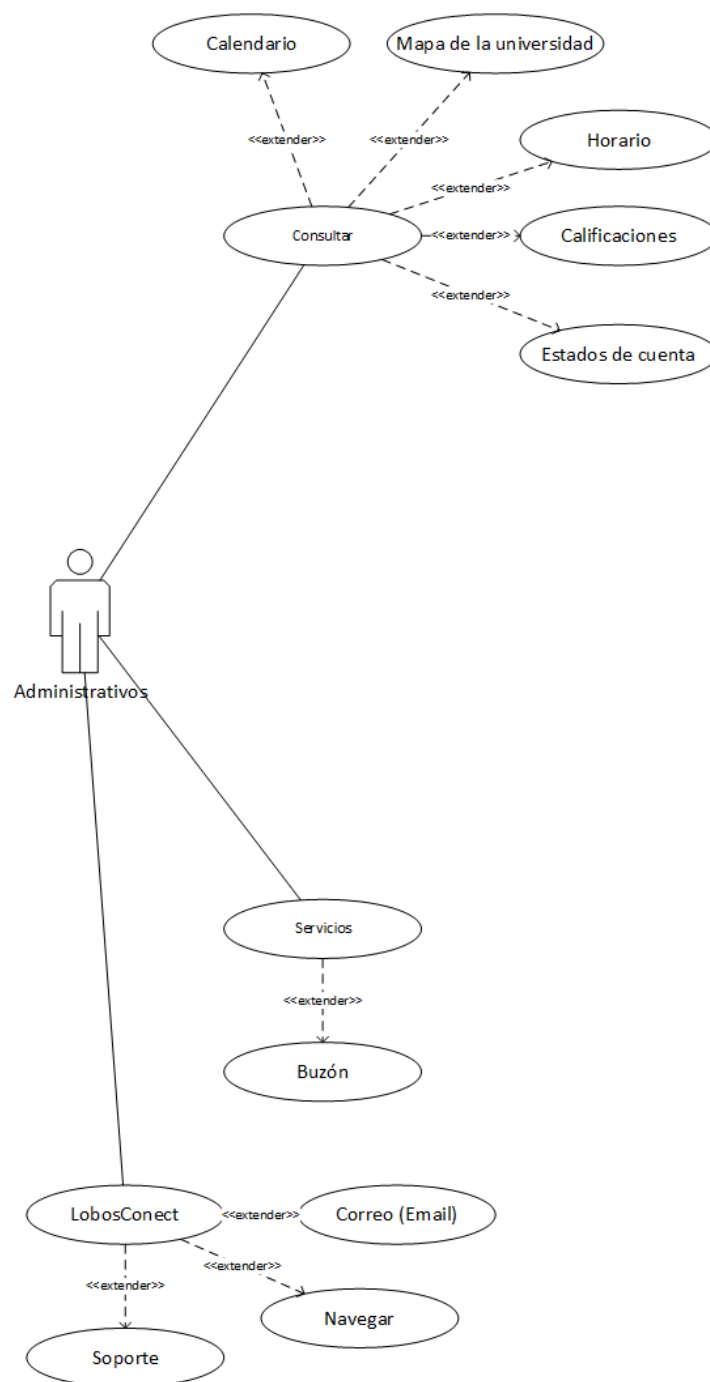


Diagrama de casos de uso: Administrador



Descripción de casos de uso

Consultas

Caso de uso:	Calificaciones
Actores:	Alumnos
Descripción: Este caso se representa cuando un alumno requiere generar un informe de sus calificaciones del cuatrimestre actual.	

Caso de uso:	Horario
Actores:	Alumnos y Administrativos
Descripción: Este caso se representa cuando un alumno requiere ver el horario en el que se impartirá la materia ya que es conveniente verificar horas de salida y horarios muertos, esto para poder hacer una mejor horario que sea corrido, también debe generarse con nombre de la materia, horario, maestro y derecho a faltas.	

Caso de uso:	Mapa de la Universidad
Actores:	Alumnos y Administrativos
Descripción: Este caso se representa cuando un alumno o Administrativo requiere saber dónde está ubicado un aula, cafetería, estacionamiento, edificios administrativos etc.	

Caso de uso:	Edo. Cuentas
Actores:	Alumnos y Administración UT
Descripción: Este caso se representa cuando un alumno requiere saber su estado de cuenta en cuanto a saldo, número de cuenta, retiros, monto anterior, fechas, movimientos de cualquier tipo, etc.	

Caso de uso:	Calendario de la Universidad
Actores:	Alumnos y Administrativos
Descripción: Este caso se representa cuando un alumno o Administrativo requiere consultar el calendario vigente de la universidad.	

Servicios

Caso de Uso	Abono Redituó
Actores:	Alumno
Descripción: En el caso de uso de Abono Redituó este le permitirá al alumno, elegir cantidades o ingresar la cantidad que se desea abonar, misma que al seleccionar el botón de abonar, nos mostrara o se generará un documento (ticket) cómo comprobante de pago.	

Caso de Uso	Pago Colegiatura
Actores:	Alumno
Descripción: En el caso de uso de Pago de Colegiatura, Es una de las opciones que el usuario (Alumno) podrá seleccionar de un ComboBox, Al ser Seleccionada Habilitará las opciones de pago que se tendrá que elegir mediante RadioButton (Redituó, Tarjeta, Deposito) y un TextArea con información del costo o de la cantidad a pagar .En el caso de ser Seleccionada la Tarjeta, nos habilitará el Textfield donde se podrá ingresar el número de tarjeta, y también se habilitará mediante Combo_box los datos para elegir fecha de vencimiento y CVE. Para todos los tipos de pago al momento de dar clic en boton Pagar en el TextArea nos aparecerá información sobre las cantidades que se pagaran (Ticket) el cual el alumno elegirá si quiere imprimirlo o no.	

Caso de Uso	Pago Examen
Actores:	Alumno
Descripción: En el caso de uso de Pago de Examen, Es una de las opciones que el usuario (Alumno) podrá seleccionar de un ComboBox, Al ser Seleccionada Habilitará las opciones de pago que se tendrá que elegir mediante RadioButton (Redituó, Tarjeta, Deposito) y un TextArea con información del costo o de la cantidad a pagar .En el caso de ser Seleccionada la Tarjeta, nos	

habilitará el Textfield donde se podrá ingresar el número de tarjeta, y también se habilitará mediante Combo_box los datos para elegir fecha de vencimiento y CVE. Para todos los tipos de pago al momento de dar clic en boton Pagar en el TextArea nos aparecerá información sobre las cantidades que se pagaran (Ticket) el cual el alumno elegirá si quiere imprimirlo o no.

Caso de Uso	Pago Inscripción
Actores:	Alumno
<p>Descripción: En el caso de uso de Pago de Inscripción Es una de las opciones que el usuario (Alumno) podrá seleccionar de un ComboBox, Al ser Seleccionada Habilitará las opciones de pago que se tendrá que elegir mediante RadioButton (Redituó, Tarjeta, Deposito) y un TextArea con información del costo o de la cantidad a pagar .En el caso de ser Seleccionada la Tarjeta, nos habilitará el Textfield donde se podrá ingresar el número de tarjeta, y también se habilitará mediante Combo_box los datos para elegir fecha de vencimiento y CVE. Para todos los tipos de pago al momento de dar clic en boton Pagar en el TextArea nos aparecerá información sobre las cantidades que se pagaran (Ticket) el cual el alumno elegirá si quiere imprimirlo o no.</p>	

Caso de Uso	Pago Titulación
Actores:	Alumno
<p>Descripción: En el caso de uso de Pago de Titulación Es una de las opciones que el usuario (Alumno) podrá seleccionar de un ComboBox, Al ser Seleccionada Habilitará las opciones de pago que se tendrá que elegir mediante RadioButton (Redituó, Tarjeta, Deposito) y un TextArea con información del costo o de la cantidad a pagar. En el caso de ser Seleccionada la Tarjeta, nos habilitará el Textfield donde se podrá ingresar el número de tarjeta, y también se habilitará mediante Combo_box los datos para elegir fecha de vencimiento y CVE. Para todos los tipos de pago al momento de dar clic en boton Pagar en el TextArea nos aparecerá información sobre las cantidades que se pagaran (Ticket) el cual el alumno elegirá si quiere imprimirlo o no.</p>	

Caso de Uso	Lo boletos
Actores:	Alumno
<p>Descripción: En el caso de uso de loboletos este apartado permitirá seleccionar el servicio que se desea pagar mediante la selección de la opción de un combo box , y una vez seleccionado en un text área aparecerá información en forma de ticket con información del costo del servicio , mismo que el alumno podrá decidir si imprimir o no.</p>	

Caso de uso	Evaluación Servicios y Docentes
Actores	Alumnos
Descripción	<p>Conjunto de actividades y mecanismos para el Ingreso, la Promoción, el Reconocimiento y la Permanencia en el servicio público educativo y el impulso a la formación continua, con la finalidad de garantizar la idoneidad de los conocimientos y capacidades del Personal Docente y del Personal con Funciones de Dirección y de Supervisión en la Educación Básica y Media Superior que imparta el Estado y sus Organismos Descentralizados.</p> <p>La encuesta de los trabajadores se puede realizar, basada en cuestionarios estandarizados, entrevistas personales estructuradas, o en el marco de un diálogo grupal.</p>

Caso de uso	Buzón
Actores	Alumnos y administrativos
Descripción	<p>Es una herramienta, en línea, que nos permite conocer su opinión sobre los servicios y trámites disponibles en la Universidad Tecnológica de Aguascalientes.</p> <p><u>Usuario:</u> cualquier persona relacionada con la Universidad Tecnológica de Aguascalientes, a través de cualquiera de sus servicios, académicos y de gestión, sea o no miembro de la Comunidad Universitaria.</p>

LobosConect

Caso de Uso	Email
Actores	Usuario
Descripción: El caso de uso se efectúa al momento en el que el usuario registrado y logueado tiene la necesidad de consultar su correo electrónico institucional. El sistema debe permitir al usuario acceder al correo electrónico institucional mediante la plataforma outlook a través de navegador chrome, todo esto con el fin de efectuar las tareas típicas del uso de un correo electrónico (consulta, envío y borrado de correos).	

Caso de Uso	Navegar
Actores	Usuario
Descripción: La opción de navegar es el caso que permite al usuario registrado y logueado navegar por internet desde el cajero mediante el navegador chrome de google. Permitirá al usuario realizar una navegación libre por internet salvo la restricción interna impuesta por la institución en cumplimiento con las normas de restricción de proxy de navegación.	

Caso de Uso	Contacto Soporte Técnico
Actores	Usuario
Descripción: El caso de uso "Contacto Soporte Técnico" permite realizar un mensaje mediante la misma aplicación, donde se podrá especificar libremente la problemática presentada por el sistema. El correo una vez redactado se enviará directamente al encargado de soporte técnico con los datos del usuario que emitió el mensaje (los datos deben ser: nombre, matrícula, correo electrónico), esto con el fin de que el soporte técnico se ponga en contacto con el usuario afectado y dar pronta solución al problema.	

Características de los usuarios

Tipo de usuario	Estudiantes
Formación	Bachillerato o superior
Actividades	Consultas y realización de servicios y trámites

Tipo de usuario	Administrativos
Formación	Bachillerato o superior
Actividades	Consultas

Restricciones

- Requiere estar conectada a la red interna de la universidad.
- El sistema deberá poder conectar con las bases de datos solicitadas de la universidad.
- Lenguaje y herramientas: Java, MVC y MySQL.
- Servidor de base de datos robusto para atender múltiples solicitudes al tiempo.
- El sistema se diseñará según un modelo cliente/servidor.
- El sistema deberá tener un diseño e implementación sencilla.

Suposiciones y dependencias

- Se asume que los requisitos aquí descritos son estables
- Se le permitirá a la aplicación acceso a las bases de datos que el mismo necesita para su correcto funcionamiento.
- Los módulos (Cajeros) en los que se vaya a ejecutar el sistema deben cumplir los requisitos antes indicados para garantizar una ejecución correcta de la misma.
- Los datos que alimentan al sistema provienen de servicios ya existentes en la universidad, por lo que su actualización y soporte dependerá de los administradores asignados a los mismos.

Requisitos específicos

Requerimientos funcionales

Módulo: CONSULTAS

Identificación del requerimiento:	Rcon1
Nombre del Requerimiento:	Calificaciones
Características:	Pide al usuario datos para ingresar a su cuenta
Descripción del requerimiento:	El sistema podrá ser consultado para visualizar las calificaciones de cada alumno así como su promedio, materias, unidades, cuatrimestre y promedio general.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rcon2
Nombre del Requerimiento:	Horario
Características:	Pide al usuario ingresar a su cuenta
Descripción del requerimiento:	El sistema podrá ser consultado para visualizar el horario que tiene cada materia dentro de su cuatrimestre con hora, fecha y materia. Faltas, maestro asignado, y aula correspondiente
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rcon3
Nombre del Requerimiento:	Mapa
Características:	Pide al usuario lugar de preferencia para visualizar

Descripción del requerimiento:	El sistema podrá ser consultado para visualizar el lugar de preferencia y sus alrededores para verificar donde está ubicado.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rcon4
Nombre del Requerimiento:	Estado de cuenta
Características:	Pide al usuario ingresar a su cuenta
Descripción del requerimiento:	El sistema podrá ser consultado para visualizar el estado de cuenta y el cual tendrá movimientos realizados, número de cuenta, nombre, dirección, fechas, montos, etc.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta

Módulo: TRÁMITES

Identificación del requerimiento:	Rtra1
Nombre del Requerimiento:	Pre-Trámite de becas
Características:	El alumno consultará información de las becas
Descripción del requerimiento:	El estudiante deberá realizar un estudio socioeconómico antes de poder solicitar cualquier beca, esto para realizar un filtrado de las becas que puede solicitar y sea más fácil su aplicación.
Requerimiento NO funcional:	
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rtra2
Nombre del Requerimiento:	Solicitud de beca
Características:	El alumno podrá de una lista solicitar la beca que le pueda aplicar en el portal de la universidad.

Descripción del requerimiento:	El estudiante deberá realizar un estudio socioeconómico antes de solicitar una beca, esto para realizar un filtrado de las becas que puede solicitar y sea más fácil su aplicación.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rtra3
Nombre del Requerimiento:	Solicitud de Constancia
Características:	El alumno podrá pre-solicitar una constancia el sistema le generará un ticket con el que podrá recojerla.
Descripción del requerimiento:	Si es un alumno regular, este le permitirá solicitar una constancia de estudios, donde pedirá si va dirigida para alguien o si deberá tener un formato específico. Una vez llenado los campos, nos arroja la referencia de pago en Banco o pagar en este mismo cajero. De cumplir el pago podrá ahora imprimir su constancia.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rtra4
Nombre del Requerimiento:	Pago de constancia de estudios
Características:	El alumno podrá visualizar su constancia digital válida
Descripción del requerimiento:	Una vez pagada la constancia, este campo nos permite imprimirla de inmediato.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8

Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rtra5
Nombre del Requerimiento:	Pago de constancia de no adeudos
Características:	El alumno podrá solicitar una constancia de no adeudos
Descripción del requerimiento:	Nos permite verificar si tenemos posibles adeudos de nuestro ciclo escolar, si los tenemos nos liga a la parte de cuenta para poder pagarlos.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rtra6
Nombre del Requerimiento:	Pago de constancia de no adeudos
Características:	El alumno podrá generar una constancia de no adeudos
Descripción del requerimiento:	Si el alumno no tiene ningún adeudo, nos permite realizar la impresión de una constancia donde no tengamos adeudos, esto por si tenemos inconveniencias de algún tipo.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta

Módulo: SERVICIOS

Identificación del requerimiento:	Rserv1
Nombre del Requerimiento:	Abono Crédito UT
Características:	El usuario deberá Iniciar sesión o estar registrado para acceder a la información correspondiente de su crédito

Descripción del requerimiento:	El alumno podrá depositar ("abonar ") a un tipo de cuenta que le servirá como un tipo de crédito con el cual podrá realizar pagos relacionados con la universidad (Pagos colegiaturas, pagos de exámenes, etc.).
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rserv2
Nombre del Requerimiento:	Pago colegiatura
Características:	El usuario deberá Iniciar sesión o estar registrado para acceder a la información correspondiente a sus colegiaturas
Descripción del requerimiento:	Servicio que el Estudiante podrá utilizar para poder realizar los pagos de las colegiaturas correspondientes (4 en cada cuatrimestre).
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rserv3
Nombre del Requerimiento:	Pago examen
Características:	El usuario deberá Iniciar sesión o estar registrado para acceder a la información correspondiente de los exámenes que se pueden pagar
Descripción del requerimiento:	Este apartado servirá para que el Estudiante pueda realizar el pago de algún examen (intercuatrimestral) de manera ágil y rápida.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	Rserv4
Nombre del Requerimiento:	Pago Inscripción
Características:	El usuario deberá Iniciar sesión o estar registrado para acceder a la información correspondiente a la reinscripción
Descripción del requerimiento:	El estudiante podrá utilizar este servicio como otra forma para realizar su pago correspondiente a la inscripción o reinscripción de cuatrimestre a cursar.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rserv5
Nombre del Requerimiento:	Pago Titulación
Características:	El usuario deberá Iniciar sesión o estar registrado para acceder a la información correspondiente al trámite de titulación
Descripción del requerimiento:	El estudiante podrá utilizar este servicio como otra forma para realizar su pago correspondiente al trámite de titulación.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rserv6
Nombre del Requerimiento:	Lo boletos
Características:	El usuario deberá Iniciar sesión o estar registrado para acceder a la información para la obtención de boletos de copias, impresiones y Lobobus
Descripción del requerimiento:	El estudiante podrá utilizar este servicio para obtener pases de autobús nocturno, así como boletos para poder imprimir y sacar copias dentro de la institución

Requerimiento NO funcional:	
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rserv7
Nombre del Requerimiento:	Servicios: “Evaluación servicios y docentes”.
Características:	Los servicios de evaluaciones deberán identificarse para acceder a cualquier parte del sistema.
Descripción del requerimiento:	Los sistemas educativos necesitan desarrollar mecanismos confiables y evaluaciones rigurosas para identificar las fortalezas y debilidades de sus profesores. (servicio y docente)
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rserv8
Nombre del Requerimiento:	Servicios: “Buzón”.
Características:	Los servicios de buzón deberán avisarse en el sistema para acceder a cualquier parte del sistema.
Descripción del requerimiento:	Lista todas las vías de acceso de buzón que el usuario tiene permiso para utilizar. Para los buzones tradicionales, si se configura una raíz virtual para el usuario, las vías de acceso de buzón se devuelven como vías de acceso relativas. Para los buzones distribuidos, la raíz virtual debe estar configurada para el usuario. (Matricula, Nombre, Correo, Fecha del incidente, Asunto y Mensaje)
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	Rlob1
Nombre del Requerimiento:	Email
Características:	Sistema de consulta de correo electrónico institucional
Descripción del requerimiento:	El sistema permitirá a los usuarios enviar, consultar y eliminar correos electrónicos desde el sistema mediante su usuario y dirección de correo electrónico institucional
Requerimiento NO funcional:	
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rlob2
Nombre del Requerimiento:	Navegar
Características:	Los usuarios tendrán la posibilidad de navegar en internet desde un navegador de internet.
Descripción del requerimiento:	Los usuarios deben de seleccionar la opción de navegar, acto seguido el sistema abrirá el navegador de internet para realizar alguna actividad en para la que requiera consultar en internet.
Requerimiento NO funcional:	RNF1 RNF2 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	Rlob3
Nombre del Requerimiento:	Contacto Soporte
Características:	Información general y completa para contacto con soporte técnico del CajeroUT

Descripción del requerimiento:	Pantalla en la que se debe de mostrar todo lo necesario para ponerse de acuerdo con el soporte técnico por presuntas fallas de cualquier índole en el sistema, dicha información debe ser el número de teléfono, correo electrónico, dirección física del departamento de soporte técnico, teléfono celular del técnico (estos datos serán proporcionados por el cliente al líder de equipo).
Requerimiento NO funcional:	RNF1 RNF2 RNF3 RNF4 RNF5 RNF6 RNF7 RNF8
Prioridad del requerimiento:	Alta

Requerimientos no funcionales

Identificación del requerimiento:	RNF1
Nombre del Requerimiento:	Interfaz del sistema
Características:	El sistema presentara una interfaz de usuario sencilla para que sea de fácil manejo a los usuarios del sistema
Descripción del requerimiento:	El sistema debe tener una interfaz de uso intuitiva y sencilla.
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	RNF2
Nombre del Requerimiento:	Compatibilidad touchscreen
Características:	La interfaz del usuario deberá de ser funcional con la característica touchscreen que se planea tengan los módulos (CajerosUT).
Descripción del requerimiento:	La interfaz debe estar desarrollada de modo que su uso sea a fin al uso de tecnologías touchpad, es decir evitar la necesidad de clics en la aplicación.
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	RNF3
Nombre del Requerimiento:	Ayuda en el uso del sistema

Características:	La interfaz del usuario deberá de presentar un sistema de ayuda para que los mismos usuarios del sistema se les faciliten el trabajo en cuanto al manejo del sistema.
Descripción del requerimiento:	La interfaz debe estar complementada con un buen sistema de ayuda (la administración puede recaer en personal con poca experiencia en el uso de aplicaciones informáticas)
Prioridad del requerimiento:	Media
Identificación del requerimiento:	RNF4
Nombre del Requerimiento:	Mantenimiento
Características:	El sistema deberá de tener un manual de instalación y manual de usuario para facilitarlos mantenimientos que serán realizados por el administrador.
Descripción del requerimiento:	El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	RNF5
Nombre del Requerimiento:	Desempeño
Características:	El sistema garantizara a los usuarios un desempeñen cuanto a los datos almacenado en el sistema ofreciéndole una confiabilidad a esta misma.
Descripción del requerimiento:	Garantizar el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	RNF6
Nombre del Requerimiento:	Nivel de Usuario
Características:	Garantizara al usuario el acceso de información de acuerdo al nivel que posee.
Descripción del requerimiento:	Facilidades y controles para permitir el acceso a la información al personal autorizado a través de Internet, con la intención de consultar y subir información pertinente para cada una de ellas.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RNF7
Nombre del Requerimiento:	Confiabilidad continua del sistema
Características:	El sistema tendrá que estar en funcionamiento las 24 horas los 7 días de la semana. Ya que es una página web diseñada para la carga de datos y comunicación entre usuarios.
Descripción del requerimiento:	La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 6 días (lunes a sábado) y al menos en el intervalo de tiempo en que la universidad se encuentra abierta (7:00 a 22:00 horas), garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes, contar con una contingencia, generación de alarmas.
Prioridad del requerimiento:	Alta
Identificación del requerimiento:	RNF8
Nombre del Requerimiento:	Seguridad en información
Características:	El sistema garantizara a los usuarios una seguridad en cuanto a la información que se procede en el sistema.
Descripción del requerimiento:	Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.
Prioridad del requerimiento:	Alta

Requisitos comunes de las interfaces

Interfaces de usuario

La interfaz con el usuario consistirá en un conjunto de ventanas con botones, listas y campos de textos. Ésta deberá ser construida específicamente para el sistema propuesto y, será visualizada desde un navegador de internet

Interfaces de hardware

Será necesario disponer de equipos de cómputos en perfecto estado con las siguientes características:

- Adaptadores de red.
- Procesador de 1.66GHz o superior.
- Memoria mínima de 256Mb.

Interfaces de software

- Sistema Operativo: Windows XP o superior.
- Explorador: Mozilla o Chrome.

Interfaces de comunicación

Los servidores, clientes y aplicaciones se comunicarán entre sí, mediante protocolos estándares en internet, siempre que sea posible. Por ejemplo, para la operación con datos almacenados en bases de datos se realizará por medio de la red Ethernet de la universidad y por sus servicios de intranet.

Requerimientos funcionales

Rcon1: Calificaciones

Solo los alumnos podrán acceder a generar un reporte de su historial de calificaciones, el cual se desplegará en la pantalla y les permitirá tenerlo en formato digital o impreso.

Rcon2: Horario

Solo los alumnos podrán visualizar su horario de clases, el cual se desplegará en la pantalla.

Solo se mostrará el horario que tenga asignado actualmente.

Rcon3: Mapa

Cualquier usuario podrá ver en pantalla el mapa de la universidad, al dar clic a la opción la pantalla mostrará el mapa como imagen en la pantalla.

Rcon4: Estado de cuenta

Solo los alumnos podrán realizar consultas de su estado de cuenta.

El sistema les permitirá revisar sus adeudos en cajas y también sus adeudos cargados a través de creditoUT.

La información se desplegará en la pantalla mostrando el concepto y el monto de cada adeudo y el total.

Rtra1: Pre-Trámite de becas

Solo los alumnos podrán empezar el trámite de beca, el sistema detectará si ya hizo o no su estudio socio-económico, de no haberlo hecho el sistema lo llevará al formulario para que complete este requisito.

Rtra2: Solicitud de beca

Solo los alumnos que ya hayan hecho su estudio socio económico, podrán acceder a elegir la beca a la que se quieren postular, esto trabajará mediante el sistema de becas de la universidad y su proceso será seguido por el área de vinculación.

Rtra3: Solicitud de Constancia de estudios

Solo los alumnos podrán solicitar una constancia de estudios, este proceso requiere un previo pago o cargo a Redituó, por lo que primero se le mostrará la pantalla para hacer el pago.

Rtra4: Impresión de constancia de estudios

Solo los alumnos que hayan concretado un pago o adeudo para constancia de estudios podrán generarla mediante el sistema. La constancia se mostrará en pantalla y permitirá al alumno tenerla en digital o imprimirla, siendo un documento con folio válido.

Rtra5: Solicitud de constancia de no adeudos

Solo los alumnos podrán solicitar una constancia de no adeudo. De existir algún adeudo el alumno verá en pantalla el listado de los mismos y no podrá generar constancia.

Rtra6: Impresión de constancia de no adeudos

Solo los alumnos que no presenten adeudos al momento podrán generar su constancia y verla en pantalla para después guardarla en digital o imprimirla.

Rserv1: Abono Crédito UT

Solo los alumnos podrán abonar o liquidar adeudos cargados a su créditoUT.

El sistema les mostrará el total de adeudo y el alumno podrá elegir si abonar a la deuda o liquidarla por completo.

Rserv2: Pago colegiatura

Los alumnos podrán elegir si quieren generar ficha para hacer el pago en banco o hacer el pago vía tarjeta en el mismo módulo.

El sistema permitirá generar fichas de pago para que el alumno pueda guardarla en digital o imprimirla para posteriormente ir a un banco a pagar.

Por otro lado de elegir pagar en el módulo, el alumno solo podrá hacer el pago utilizando una tarjeta de crédito o débito.

Rserv3: Pago examen

El sistema permitirá generar fichas de pago para que el alumno pueda guardarla en digital o imprimirla para posteriormente ir a un banco a pagar.

Por otro lado de elegir pagar en el módulo, el alumno solo podrá hacer el pago utilizando una tarjeta de crédito o débito.

Rserv4: Pago Inscripción

El sistema permitirá generar fichas de pago para que el alumno pueda guardarla en digital o imprimirla para posteriormente ir a un banco a pagar.

Por otro lado de elegir pagar en el módulo, el alumno solo podrá hacer el pago utilizando una tarjeta de crédito o débito.

Rserv5: Pago Titulación

El sistema permitirá generar fichas de pago para que el alumno pueda guardarla en digital o imprimirla para posteriormente ir a un banco a pagar.

Por otro lado de elegir pagar en el módulo, el alumno solo podrá hacer el pago utilizando una tarjeta de crédito o débito.

Rserv6: Turno Ventanilla

De preferirlo, los alumnos podrán generar un turno para pasar a ventanilla, este le dará una hora aproximada para que se presente a cajas a hacer el pago que necesite.

En cajas solo se le atenderá alumnos con turno y de perder su hora asignada, el alumno deberá generar un nuevo turno.

Rserv7: Lo boletos

Solo los alumnos podrán generar boletos vía cago a su Redituó. El alumno elegirá el concepto entre coipas, impresiones o autobús.

Rserv8: Servicios: “Evaluación servicios y docentes”

Solo los alumnos podrán realizar sus evaluaciones a servicios y/o docentes de la misma forma en la que lo hacen en el Sistema de Información Táctico Operativo. CajeroUT utilizará las mismas tablas y bases de datos.

Rserv9: Servicios: “Buzón”

Los alumnos podrán emitir mensajes de quejas o sugerencias con la seguridad de que estos se almacenan de forma anónima.

Rlob1: Email

Alumnos, administrativos y docentes podrán hacer uso de esta opción para revisar su correo institucional vía web.

El sistema hará uso del explorador instalado en el propio módulo para conectarse con la herramienta de Outlook de Microsoft.

Rlob2: Navegar

Alumnos, administrativos y docentes podrán utilizar el navegador de internet instalado en el módulo para hacer consultas en internet.

Rlob3: Contacto Soporte

Todos los usuarios podrán contactar vía mensaje a soporte técnico para dudas, quejas o reportes.

El sistema pondrá en pantalla un formulario para que el usuario escriba y mande su mensaje.

El sistema mostrará también información de contacto directa al área de soporte encargada de los módulos.

Requerimientos no funcionales

Requisitos de rendimiento

Garantizar que el diseño de las consultas u otro proceso no afecte el desempeño de la base de datos, ni considerablemente el tráfico de la red.

Seguridad

Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.

Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.

Facilidades y controles para permitir el acceso a la información al personal autorizado a través de Internet, con la intención de consultar y subir información pertinente para cada una de ellas.

Fiabilidad

El sistema debe tener una interfaz de uso intuitiva y sencilla

La interfaz de usuario debe ajustarse al manual de identidad de la institución.

Disponibilidad

La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas, garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes, contar con una contingencia, generación de alarmas.

Mantenibilidad

El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible.

La interfaz debe estar complementada con un buen sistema de ayuda (la administración puede recaer en personal con poca experiencia en el uso de aplicaciones informáticas).

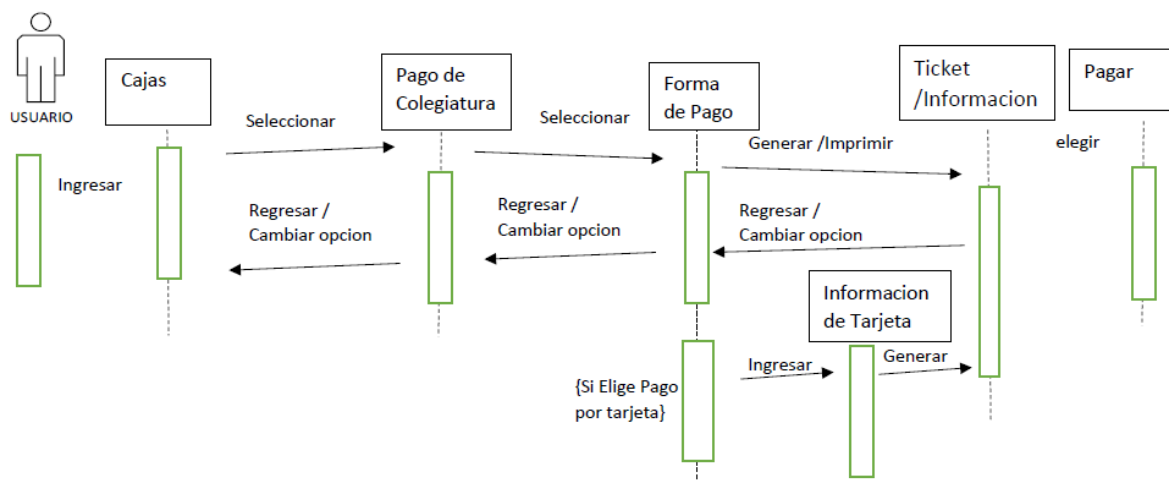
Portabilidad

El sistema será implantado bajo la plataforma de Windows y con herramientas actualizadas del JDK de Java y MySQL.

Modelado de software

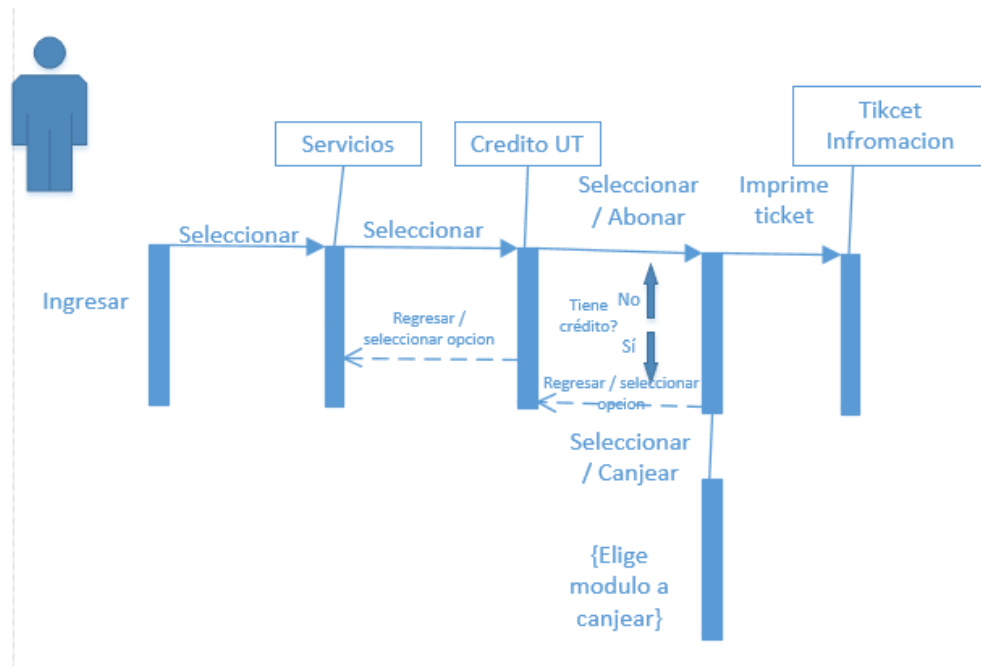
Diagramas de secuencia

Servicios: Caja (Pagos)

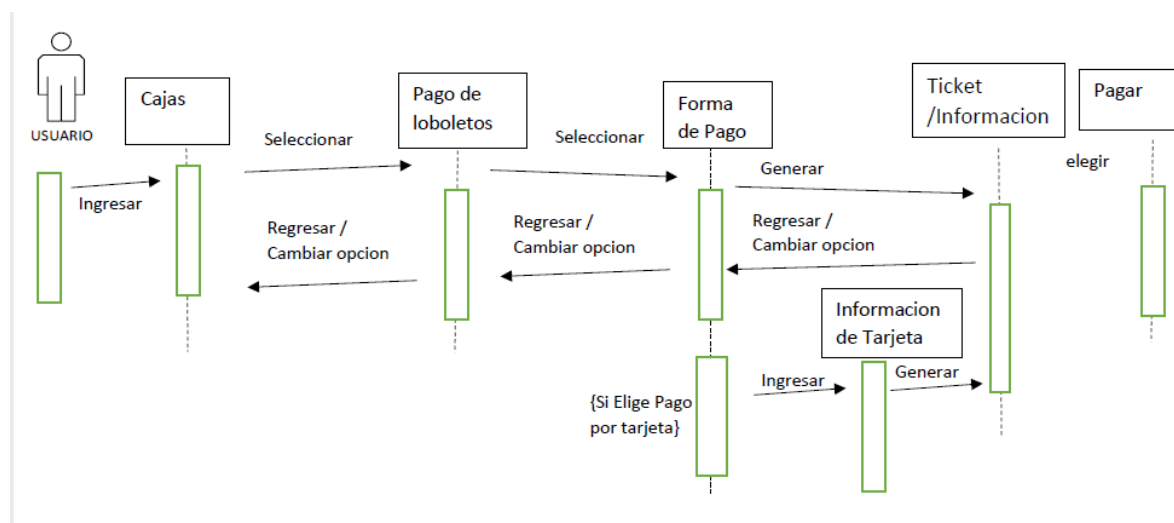


* El proceso es el mismo para todos los tipos de pago habilitados en el sistema

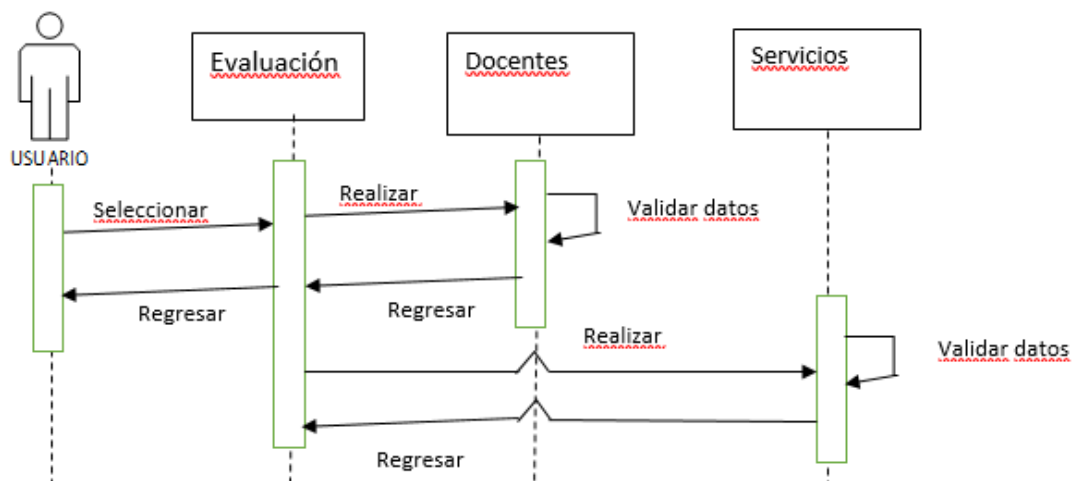
Servicios: CreditoUT



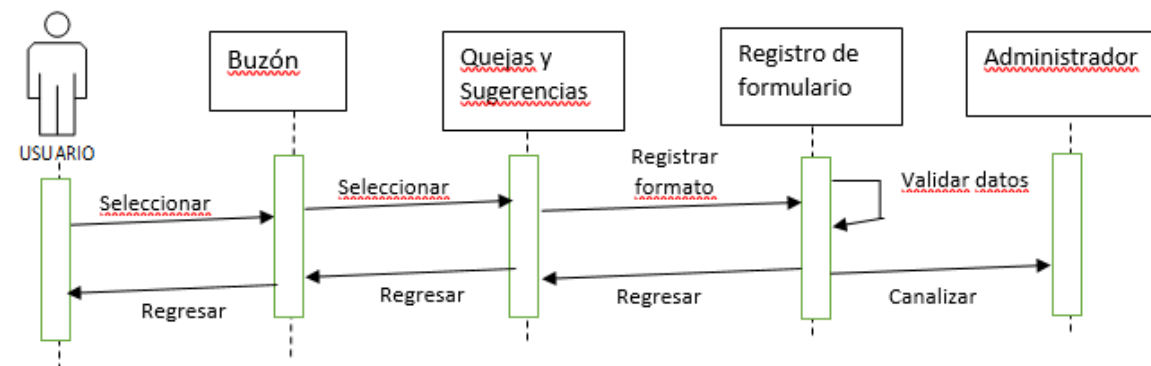
Servicios: Loboletos



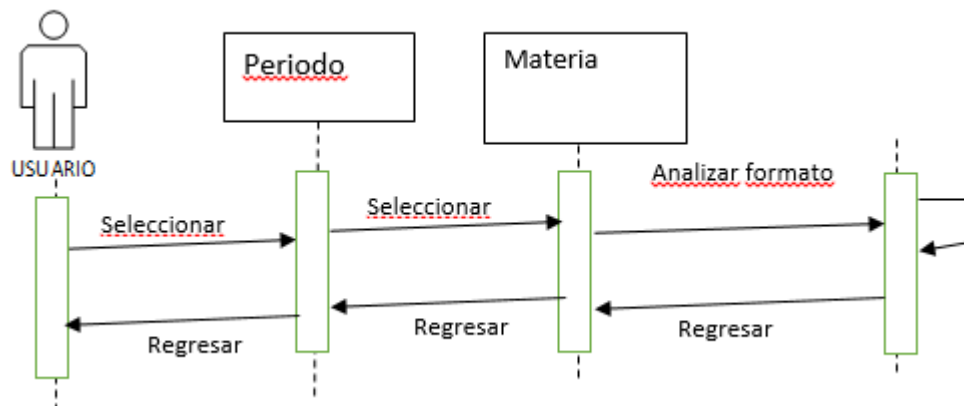
Servicios: Evaluación de servicios



Servicios: Buzón

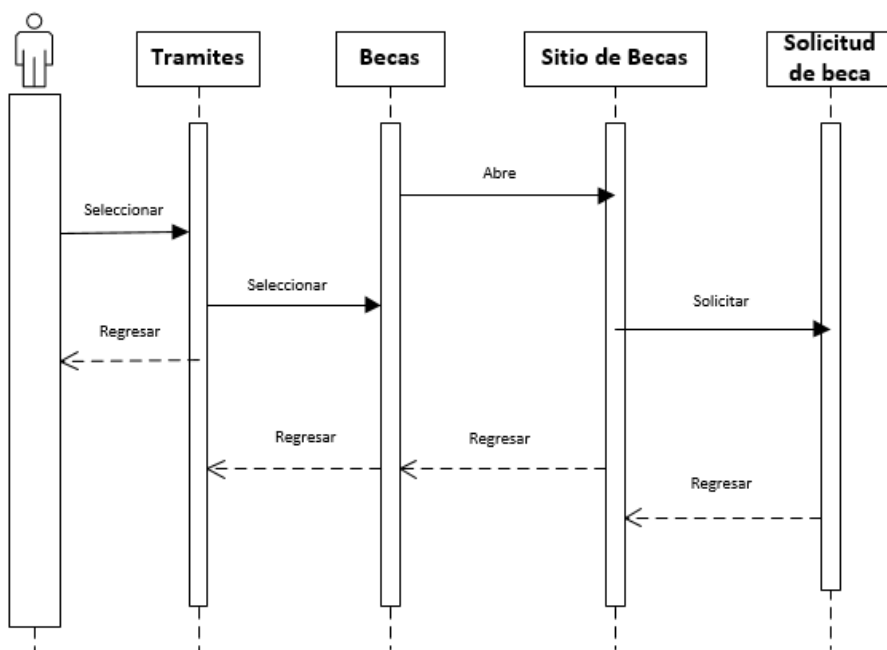


Consultas: Calificaciones

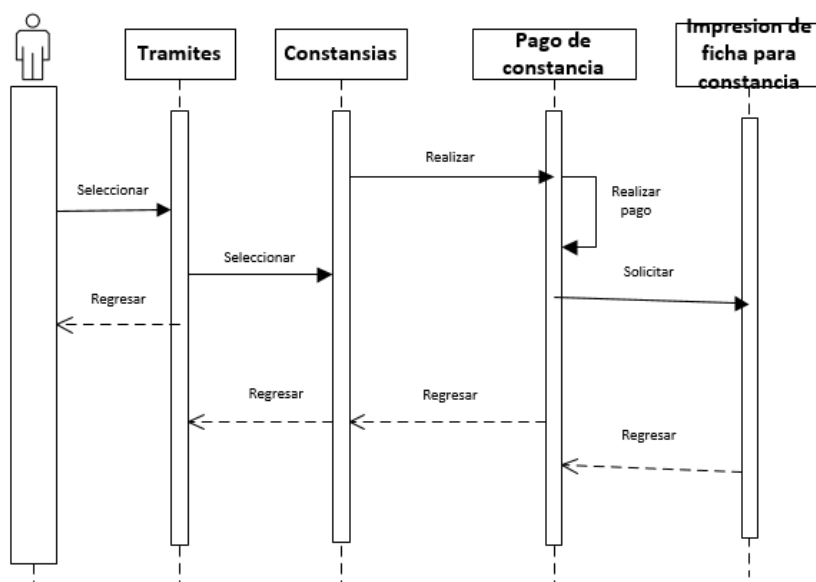


* Los módulos de horario, mapa y estado de cuenta al abrir otra plataforma no tienen funcionalidad en nuestro sistema

Trámites: Becas

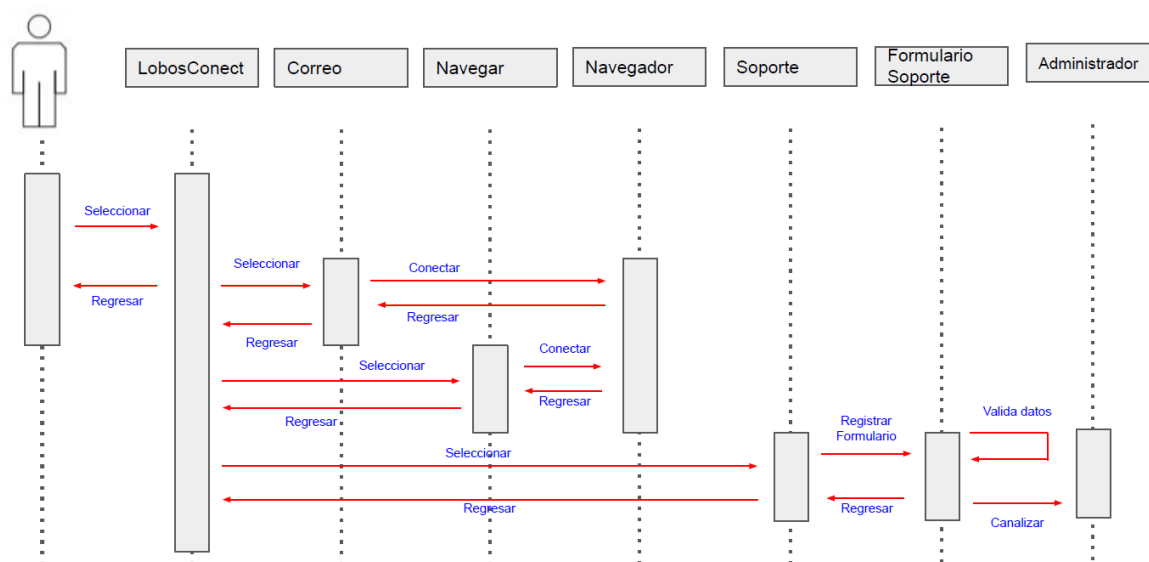


Trámites: Constancias



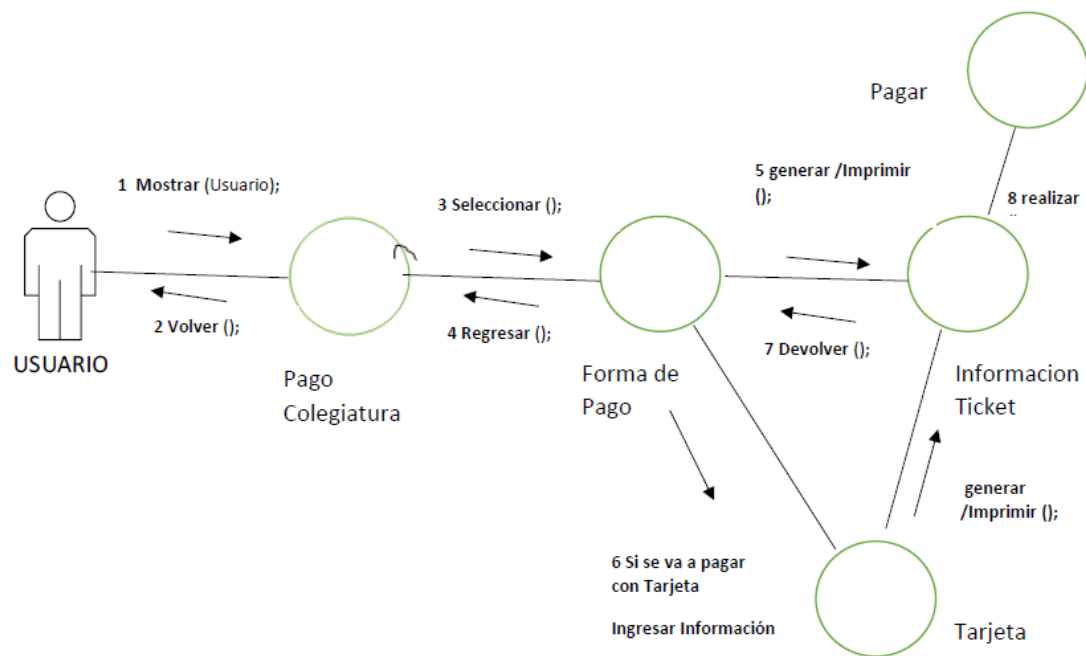
LobosConect:

Diagramas de Secuencias (LobosConect)



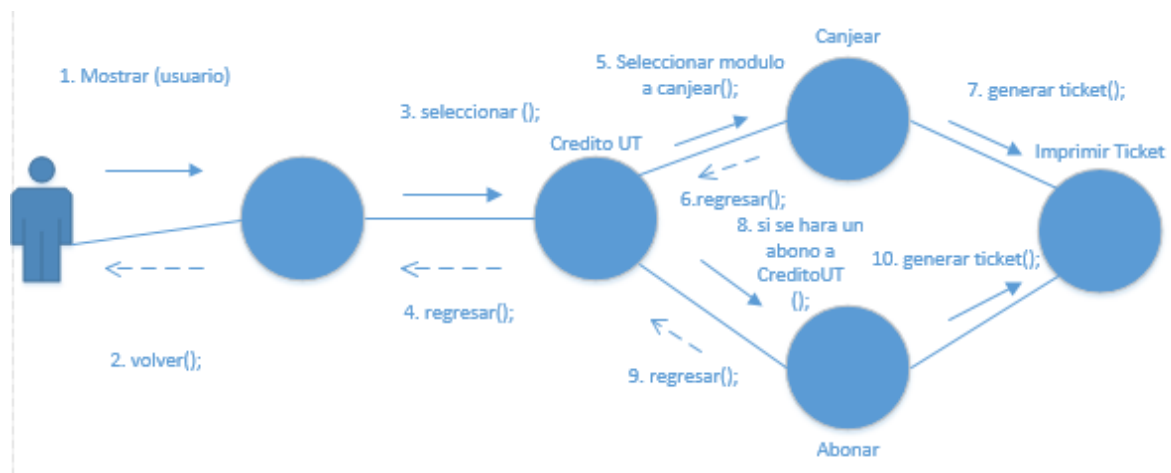
Diagramas de colaboración

Servicios: Caja (Pagos)

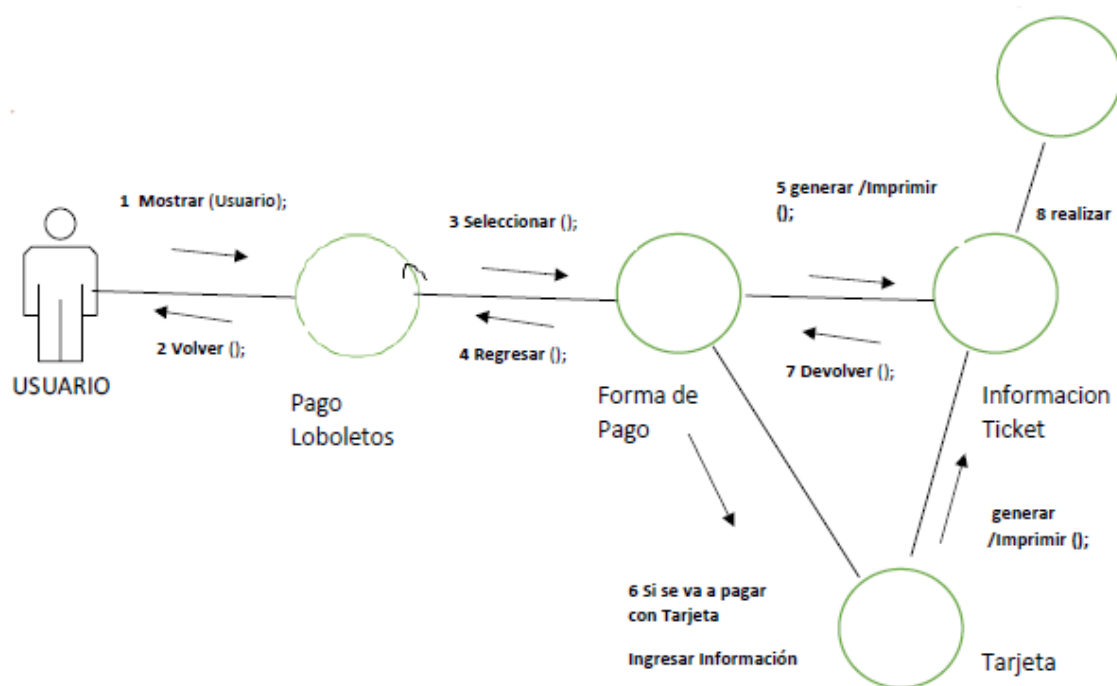


* El proceso es el mismo para todos los tipos de pago habilitados en el sistema

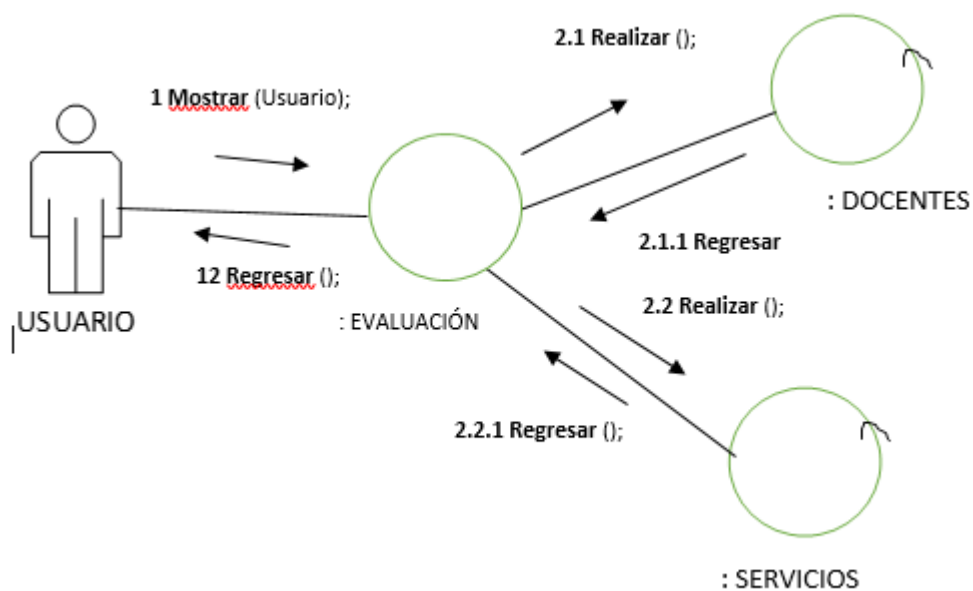
Servicios: CreditoUT



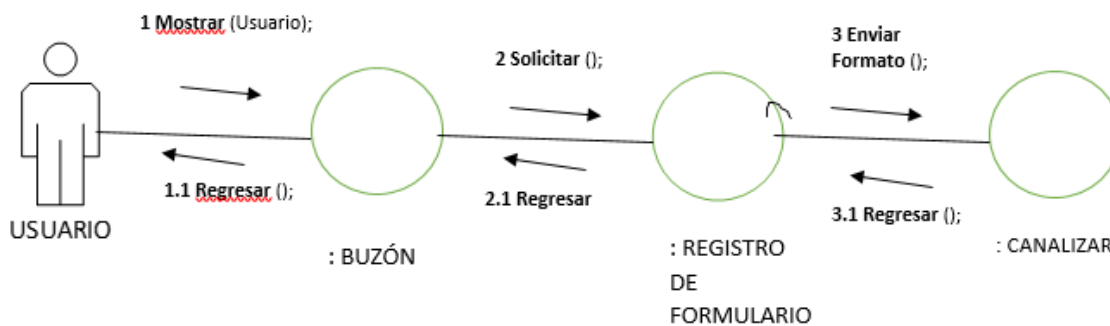
Servicios: Loboletos



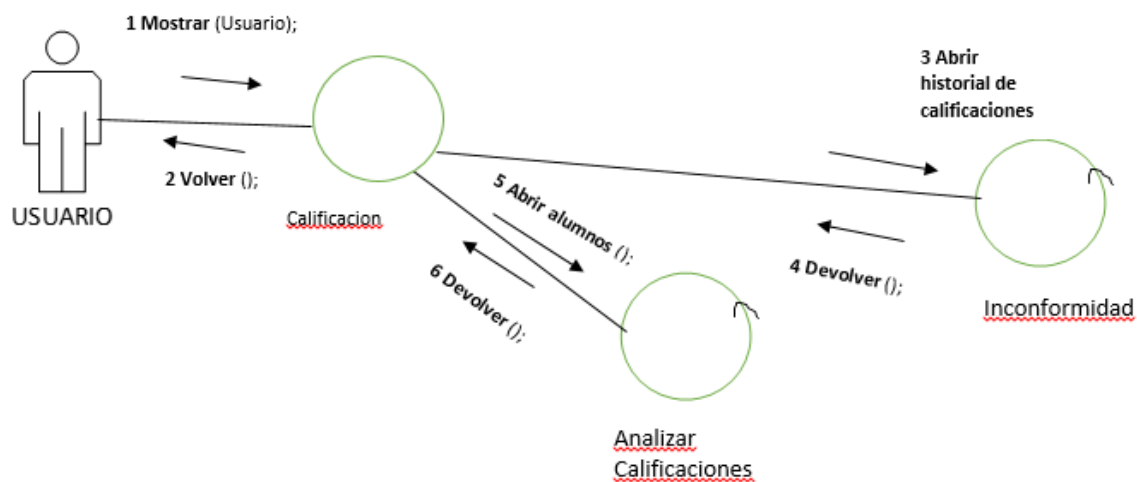
Servicios: Evaluación de servicios



Servicios: Buzón

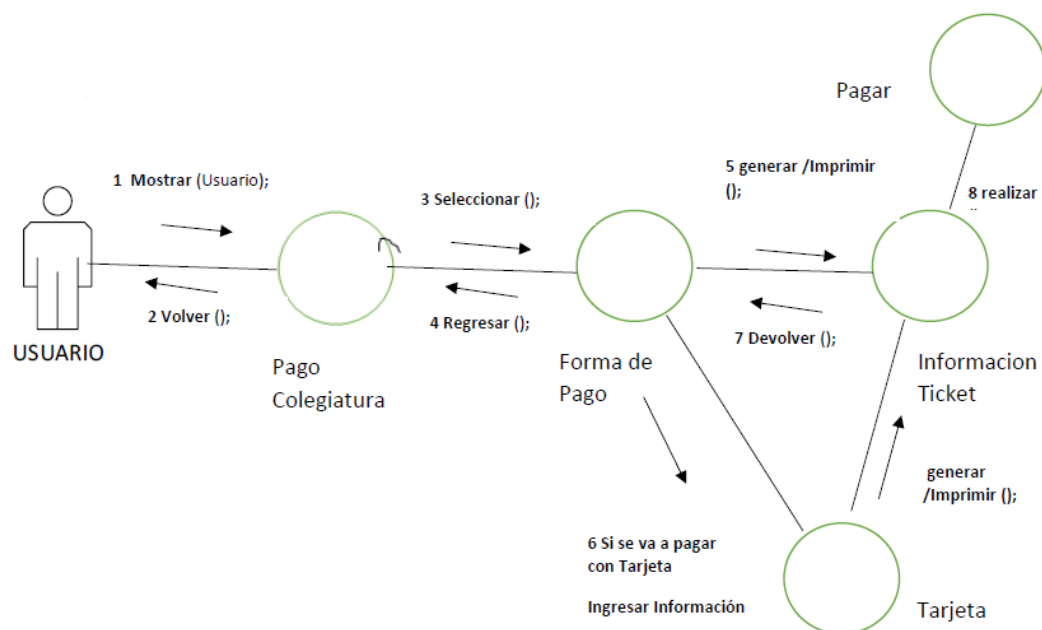


Consultas: Calificaciones



* Los módulos de horario, mapa y estado de cuenta al abrir otra plataforma no tienen funcionalidad en nuestro sistema

Trámites: Constancias

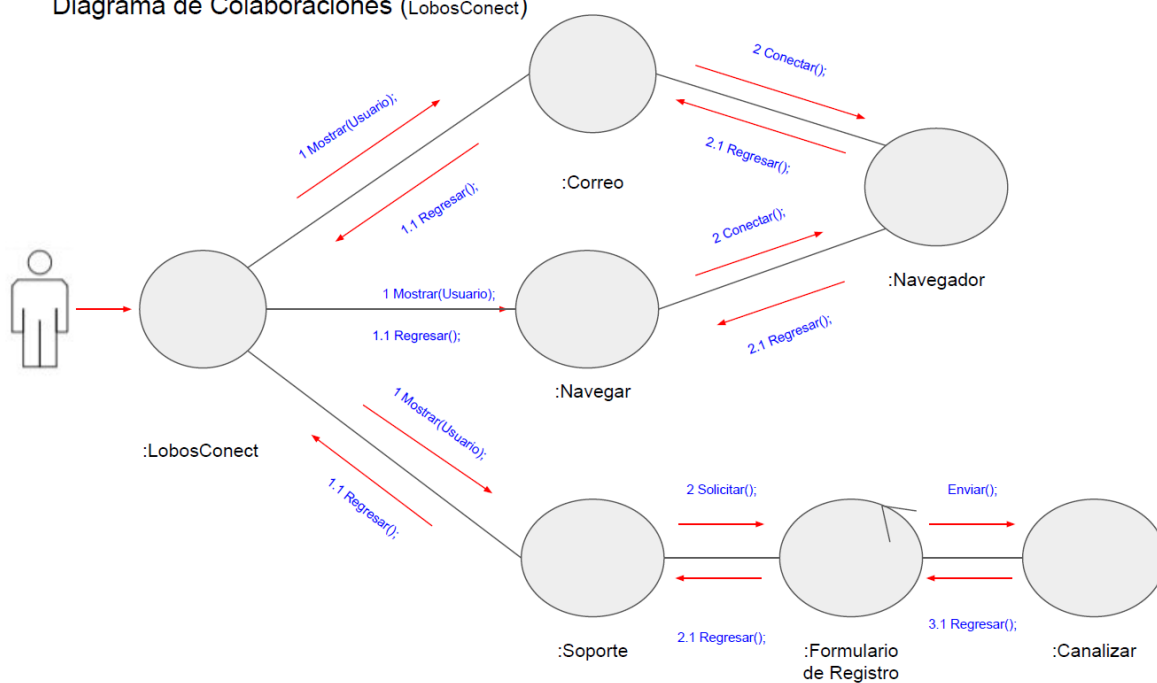


* El pre-trámite de constancia al solo ser un pago es igual al módulo de caja.

* El módulo becas al abrir otra plataforma no tienen funcionabilidad en nuestro sistema

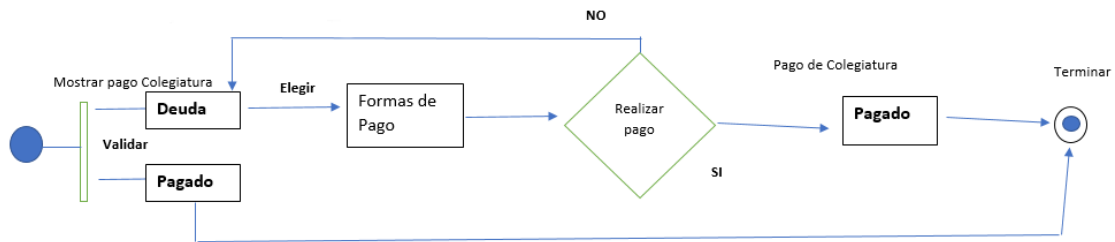
LobosConect:

Diagrama de Colaboraciones (LobosConect)

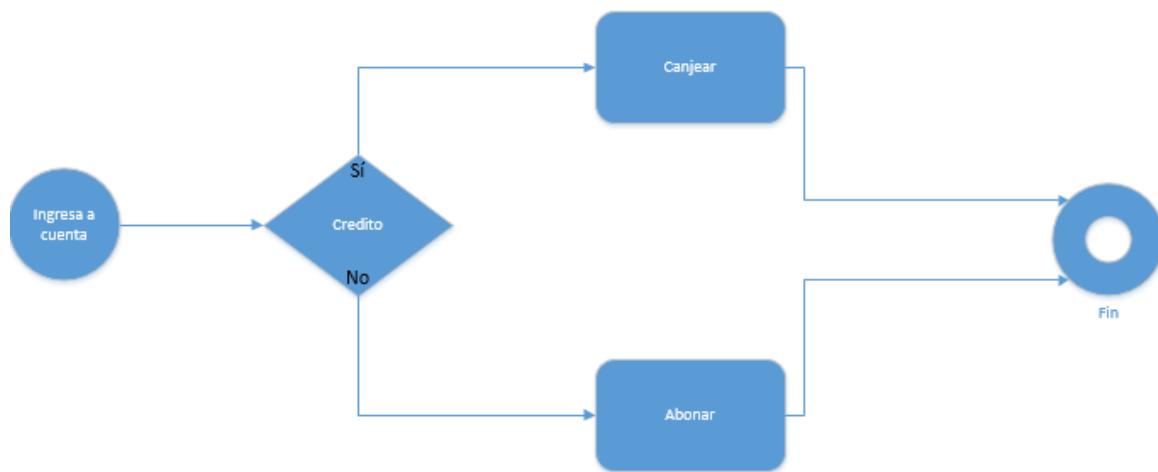


Diagramas de máquina de estados

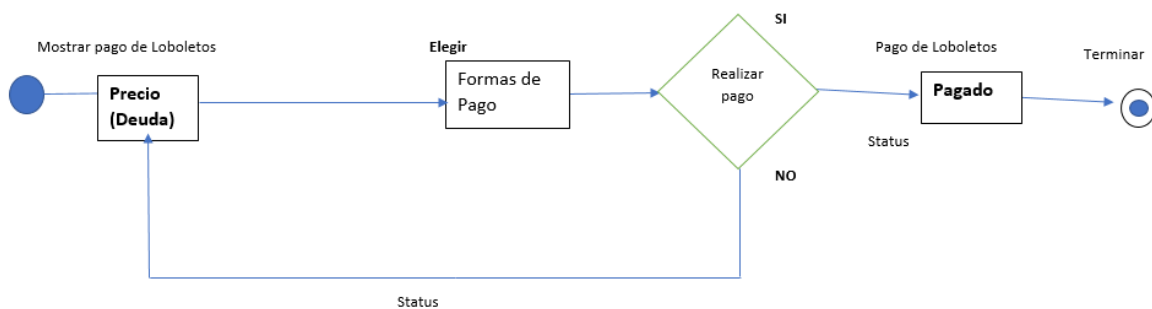
Servicios: Caja (Pagos)



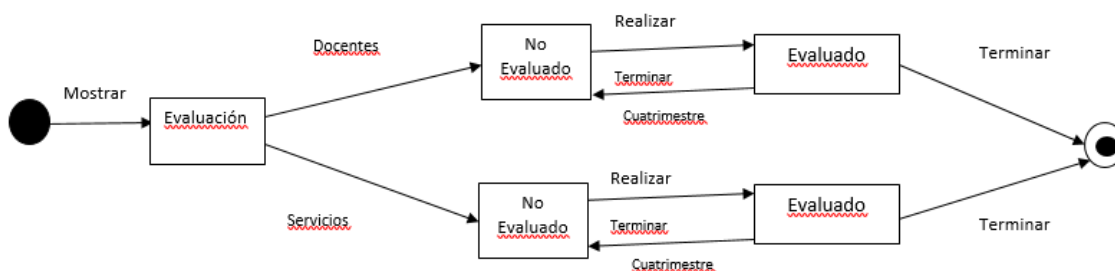
Servicios: CreditoUT



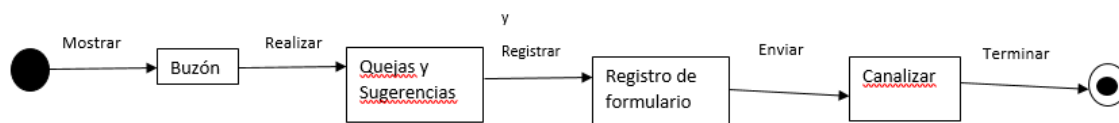
Servicios: Loboletos



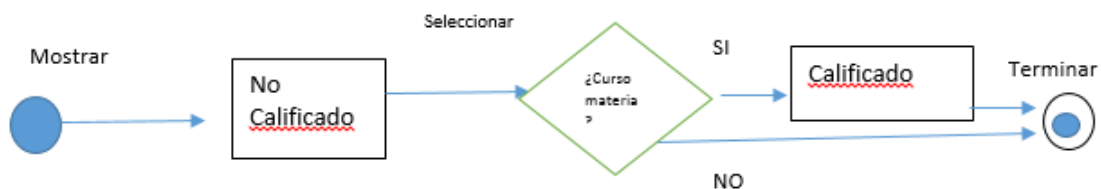
Servicios: Evaluación de servicios



Servicios: Buzón

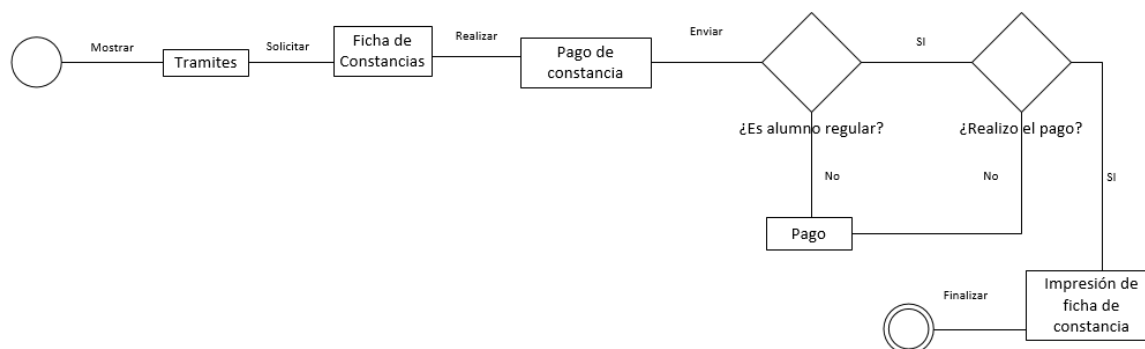


Consultas: Calificaciones



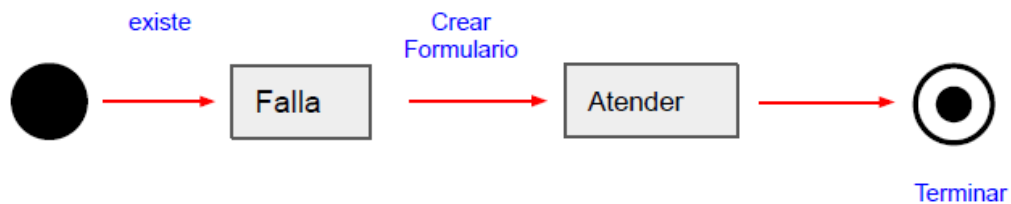
* Los módulos de horario, mapa y estado de cuenta al abrir otra plataforma no tienen funcionalidad en nuestro sistema

Trámites: Constancias



* El módulo becas al abrir otra plataforma no tienen funcionalidad en nuestro sistema

LobosConect: Soporte



* Los módulos de correo y navega al abrir otra plataforma no tienen funcionalidad en nuestro sistema

Código del sistema

Todo el código tanto desarrollado como reciclado está escrito en **lenguaje Java**, utilizando la versión más reciente de su librería JDK (Versión 12).

El aproximado de líneas de código es de: **6250**.

Modelos

ConexionMySQL

```
package Modelo;

import java.sql.*;

import javax.swing.JOptionPane;

public class conexionMySQL {

    // Conexión...

    public Connection conn;

    // Librería de MySQL, SQL SEVER, ORACLE, etc...
```



```

public String driver = "com.mysql.jdbc.Driver";

// Host
public String hostname = "localhost";

// Puerto
public String port = "3306";

// Nombre de la base de datos
public String database = "javacnn";

// Ruta de nuestra base de datos
public String url = "jdbc:mysql://" + hostname + ":" + port + "/" + database;

//public String url = "jdbc:sqlserver://" + hostname + ":" + port + "; database=" + database;

// Nombre de usuario
public String username = "javacon";

// Clave de usuario
public String password = "C4jer0UT.2019";


//Abrir la conexion a la BD
public Connection conectarSQL() {
    try {
        Class.forName(driver);
        conn = DriverManager.getConnection(url, username, password);
        System.out.println("Conexion exitosa");
    } catch (ClassNotFoundException | SQLException exsql) {
        JOptionPane.showMessageDialog(null, "Error al conectar...\nERROR="
        + exsql, "ERROR", JOptionPane.ERROR_MESSAGE);
        System.exit(0);
    } catch (Exception ex) {

```

```

        JOptionPane.showMessageDialog(null,"ERROR= "+ex,"ERROR",
JOptionPane.ERROR_MESSAGE);

    }

    return conn;

}

//Cerrar la conexion a la BD

public void cerrarConexion(){

    try {

        conn.close();

        System.out.println("Conexion cerrada con exito!");

    } catch (Exception e) {

        JOptionPane.showMessageDialog(null,"No se pudo cerrar la conexión...\nERROR=
"+e,"ERROR", JOptionPane.ERROR_MESSAGE);

    }

}

}

```

mainInicial

```

package Modelo;

import Controlador.*;

import Vista.*;

import javax.swing.JOptionPane;

public class mainInicial {

    public static void main(String args[]) {

        modelo_login miModelo = new modelo_login();
    }
}

```

```

        controlador_login miControlador = new controlador_login();

        Login miLogin = new Login();

        miControlador.setMiVistaModelo(miLogin, miModelo);
    }

    public void cargaLogin() {

        modelo_login miModelo = new modelo_login();

        controlador_login miControlador = new controlador_login();

        Login miLogin = new Login();

        miControlador.setMiVistaModelo(miLogin, miModelo);
    }
}

```

Modelo_buzon

```

package Modelo;

import Controlador.controlador_principalMenu;

import java.*;

import java.awt.Desktop;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.sql.*;

public class modelo_buzon {

    private conexionMySQL miCnn = new conexionMySQL();

    public String abreCalif(int id) {

```

```

String url = "";

try {

    String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

    Statement stmt = miCnn.conectarSQL().createStatement();

    ResultSet rs = stmt.executeQuery(SQL);

    if (rs.next()) {

        url = rs.getString("calif");

    }

    rs.close();

    stmt.close();

}

catch (SQLException ex)

{

}

return url;

}

public String abreHorario(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getString("horario");

        }

    }

}

```

```

        rs.close();

        stmt.close();
    }

    catch (SQLException ex)

    {

    }

    return url;
}

public String abreEdoCuenta(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("edoCuenta");

        }

        rs.close();

        stmt.close();

    }

    catch (SQLException ex)

    {

    }

    return url;

}

```

```

public void abreArchivo(File archivo) {

    try {

        /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop
        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);

    } catch (IOException ex) {

        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

public void abreNavegador(String url) {

    try {

        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);

    } catch (Exception e) {

    }

}

}
}

```

Modelo_constancia

```

package Modelo;

import Controlador.controlador_principalMenu;

import java.*;

import java.awt.Desktop;

```

```

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.sql.*;

public class modelo_constancia {

    private conexionMySQL miCnn = new conexionMySQL();

    public String abreCalif(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

                url = rs.getNString("calif");

            }

            rs.close();

            stmt.close();

        }

        catch (SQLException ex)

        {

        }

        return url;

    }

}

```

```

public String abreHorario(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("horario");

        }

        rs.close();

        stmt.close();

    }

    catch (SQLException ex)

    {

    }

    return url;

}

public String abreEdoCuenta(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("edoCuenta");

        }

    }

    catch (SQLException ex)

    {

    }

    return url;

}

```



```

    }

    rs.close();

    stmt.close();

}

catch (SQLException ex)

{

}

return url;

}

public void abreArchivo(File archivo) {

    try {

        /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop
        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);

    } catch (IOException ex) {

        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

public void abreNavegador(String url) {

    try {

        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);

    } catch (Exception e) {

    }

}

```

Modelo_creditoUT

Proyecto CajeroUT - Documentación

```

    }

    rs.close();

    stmt.close();
}

catch (SQLException ex)

{

}

return monto;
}

public Integer actualizaCreditoUT(int id, int monto, int abono) {

    int nuevoMonto = monto + abono;

    try {

        String SQL = "UPDATE usrconexion SET CreditoUT = '"+nuevoMonto+"' WHERE id='"+id+"'";

        JOptionPane.showMessageDialog(null,SQL);

        Statement stmt = miCnn.conectarSQL().createStatement();

        stmt.executeUpdate(SQL);

        JOptionPane.showMessageDialog(null, "Se creditoUT ahora es de: "+nuevoMonto);

        stmt.close();

    }

    catch (SQLException ex)

    {

        JOptionPane.showMessageDialog(null, "ERROR al actualizar CreditoUT");

    }

    return monto;
}

```

```
}
```

```
public String abreCalif(int id) {  
  
    String url = "";  
  
    try {  
  
        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";  
  
        Statement stmt = miCnn.conectarSQL().createStatement();  
  
        ResultSet rs = stmt.executeQuery(SQL);  
  
        if (rs.next()) {  
  
            url = rs.getNString("calif");  
  
        }  
  
        rs.close();  
  
        stmt.close();  
  
    }  
  
    catch (SQLException ex)  
  
    {  
  
    }  
  
    return url;  
  
}
```

```
public String abreHorario(int id) {  
  
    String url = "";  
  
    try {  
  
        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";  
  
        Statement stmt = miCnn.conectarSQL().createStatement();  
  
        ResultSet rs = stmt.executeQuery(SQL);
```

```

        if (rs.next()) {
            url = rs.getNString("horario");
        }

        rs.close();

        stmt.close();
    }

    catch (SQLException ex)

    {

    }

    return url;
}

public String abreEdoCuenta(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("edoCuenta");

        }

        rs.close();

        stmt.close();

    }

    catch (SQLException ex)

    {

```

```

    }

    return url;
}

public void abreArchivo(File archivo) {
    try {
        /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop
        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);
    } catch (IOException ex) {
        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);
    }
}

public void abreNavegador(String url) {
    try {
        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);
    } catch (Exception e) {

    }

}
}

```

Modelo_evaluacion

```
package Modelo;
```

```

import Controlador.controlador_principalMenu;

import java.*;

import java.awt.Desktop;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.sql.*;

public class modelo_evaluacion {

    private conexionMySQL miCnn = new conexionMySQL();

    public String abreCalif(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

                url = rs.getNString("calif");

            }

            rs.close();

            stmt.close();

        }

        catch (SQLException ex)

```

```

    {
    }

    return url;
}

public String abreHorario(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("horario");

        }

        rs.close();

        stmt.close();

    }

    catch (SQLException ex)

    {

    }

    return url;
}

public String abreEdoCuenta(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

```



```

Statement stmt = miCnn.conectarSQL().createStatement();

ResultSet rs = stmt.executeQuery(SQL);

if (rs.next()) {

    url = rs.getNString("edoCuenta");

}

rs.close();

stmt.close();

}

catch (SQLException ex)

{

}

return url;

}

public void abreArchivo(File archivo) {

    try {

        /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop
        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);

    } catch (IOException ex) {

        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

public void abreNavegador(String url) {

    try {

```

```

        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);
    }catch(Exception e) {

    }

}

}

}

Modelo_login

package Modelo;

import javax.swing.JOptionPane;

import java.sql.*;

public class modelo_login {

    public int flagMatricula;

    public String usuarioActivo;

    public int flagRol;

    private conexionMySQL miCnn = new conexionMySQL();

    //Función que evalua los datos enviados desde el controlador del controlador del login

    public boolean autentifica(int usr, int pass){

        boolean respuesta = false;

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+usr+"' AND pass='"+pass+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

```

```

        respuesta = true;

        this.flagMatricula = rs.getInt("id");

        this.flagRol = rs.getInt("rol");

        this.usuarioActivo = rs.getNString("nombre");

        //JOptionPane.showMessageDialog(null, "nombre es: "+usuarioActivo+" rol: "+flagRol);

    }

    rs.close();

    stmt.close();

}

catch (SQLException ex)

{

    JOptionPane.showMessageDialog(null,"ERROR= "+ex,"ERROR",
JOptionPane.ERROR_MESSAGE);

}

return respuesta;

}

}

```

Modelo_principalMenu

```

package Modelo;

import Controlador.controlador_principalMenu;

import java.*;

import java.awt.Desktop;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

```

```

import java.util.logging.Logger;

import java.sql.*;

public class modelo_principalMenu {

    private conexionMySQL miCnn = new conexionMySQL();

    public String abreCalif(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

                url = rs.getNString("calif");

            }

            rs.close();

            stmt.close();

        }

        catch (SQLException ex)

        {

        }

        return url;

    }

    public String abreHorario(int id) {

        String url = "";

        try {

```

```

String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

Statement stmt = miCnn.conectarSQL().createStatement();

ResultSet rs = stmt.executeQuery(SQL);

if (rs.next()) {

    url = rs.getNString("horario");

}

rs.close();

stmt.close();

}

catch (SQLException ex)

{

}

return url;

}

public String abreEdoCuenta(int id) {

String url = "";

try {

String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

Statement stmt = miCnn.conectarSQL().createStatement();

ResultSet rs = stmt.executeQuery(SQL);

if (rs.next()) {

    url = rs.getNString("edoCuenta");

}

rs.close();

stmt.close();

}

}

```

```

    }

    catch (SQLException ex)

    {

    }

    return url;
}

public void abreArchivo(File archivo) {

    try {

        /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop
        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);

    } catch (IOException ex) {

        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

public void abreNavegador(String url) {

    try {

        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);

    }catch(Exception e) {

    }

}

}
}

```

Modelo_servCaja

```
package Modelo;

import Controlador.controlador_principalMenu;

import java.*;

import java.awt.Desktop;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.sql.*;

public class modelo_servCaja {

    private conexionMySQL miCnn = new conexionMySQL();

    public String abreCalif(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

                url = rs.getString("calif");

            }

            rs.close();

            stmt.close();

        }

    }

}
```

```

        catch (SQLException ex)

        {

        }

        return url;
    }

    public String abreHorario(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

                url = rs.getNString("horario");

            }

            rs.close();

            stmt.close();

        }

        catch (SQLException ex)

        {

        }

        return url;
    }

    public String abreEdoCuenta(int id) {

        String url = "";

        try {

```



```

String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

Statement stmt = miCnn.conectarSQL().createStatement();

ResultSet rs = stmt.executeQuery(SQL);

if (rs.next()) {

    url = rs.getNString("edoCuenta");

}

rs.close();

stmt.close();

}

catch (SQLException ex)

{

}

return url;

}

public void abreArchivo(File archivo) {

    try {

        /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop
        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);

    } catch (IOException ex) {

        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

public void abreNavegador(String url) {

```

```

    try {

        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);

    }catch(Exception e) {

    }

}

}

}

```

Modelo_soporte

```

package Modelo;

import Controlador.controlador_principalMenu;

import java.*;

import java.awt.Desktop;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.sql.*;

public class modelo_soporte {

    private conexionMySQL miCnn = new conexionMySQL();

    public String abreCalif(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

```

```

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("calif");

        }

        rs.close();

        stmt.close();

    }

    catch (SQLException ex)

    {

    }

    return url;

}

public String abreHorario(int id) {

    String url = "";

    try {

        String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

        Statement stmt = miCnn.conectarSQL().createStatement();

        ResultSet rs = stmt.executeQuery(SQL);

        if (rs.next()) {

            url = rs.getNString("horario");

        }

        rs.close();

        stmt.close();

    }

```

```

        catch (SQLException ex)

        {

        }

        return url;
    }

    public String abreEdoCuenta(int id) {

        String url = "";

        try {

            String SQL = "SELECT * FROM usrconexion WHERE id='"+id+"'";

            Statement stmt = miCnn.conectarSQL().createStatement();

            ResultSet rs = stmt.executeQuery(SQL);

            if (rs.next()) {

                url = rs.getString("edoCuenta");

            }

            rs.close();

            stmt.close();

        }

        catch (SQLException ex)

        {

        }

        return url;

    }

    public void abreArchivo(File archivo) {

        try {

            /*la clase Desktop nos permitirá visualizar el archivo mediante sus métodos getDesktop

```

```

        y el método open ( ), con esto, colocaremos como argumento el objeto File que creamos
        anteriormente.*/

        Desktop.getDesktop().open(archivo);

    } catch (IOException ex) {

        Logger.getLogger(controlador_principalMenu.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

public void abreNavegador(String url) {

    try {

        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);

    } catch (Exception e) {

    }

}

}

```

Vistas

Login

```

package Vista;

import Controlador.*;

import Modelo.*;

public class Login extends javax.swing.JFrame {

    /**

```

```

* Creates new form Login

*/
public Login() {
    initComponents();
}

private void initComponents() {

    frame_servTeclado = new javax.swing.JFrame();

    teclado_btn1 = new javax.swing.JButton();

    teclado_btn3 = new javax.swing.JButton();

    teclado_btn4 = new javax.swing.JButton();

    teclado_btn5 = new javax.swing.JButton();

    teclado_btn6 = new javax.swing.JButton();

    teclado_btn7 = new javax.swing.JButton();

    teclado_btn8 = new javax.swing.JButton();

    teclado_btn9 = new javax.swing.JButton();

    teclado_btnAceptar = new javax.swing.JButton();

    teclado_btn0 = new javax.swing.JButton();

    teclado_btnAtras = new javax.swing.JButton();

    teclado_btnLimpiar = new javax.swing.JButton();

    teclado_btn2 = new javax.swing.JButton();

    jLabel1 = new javax.swing.JLabel();

    jLabel2 = new javax.swing.JLabel();

    txt_loginMatricula = new javax.swing.JTextField();

    jLabel3 = new javax.swing.JLabel();

```

```

txt_loginPass = new javax.swing.JPasswordField();

btn_login = new javax.swing.JButton();

jLabel4 = new javax.swing.JLabel();

tBtn_tecladoMatricula = new javax.swing.JToggleButton();

tBtn_tecladoPass = new javax.swing.JToggleButton();


frame_servTeclado.setTitle("ServTeclado");

frame_servTeclado.setMinimumSize(new java.awt.Dimension(250, 396));

frame_servTeclado.setSize(new java.awt.Dimension(180, 230));


teclado_btn1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn1.setText("1");


teclado_btn3.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn3.setText("3");
teclado_btn3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        teclado_btn3ActionPerformed(evt);
    }
});


teclado_btn4.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn4.setText("4");


teclado_btn5.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N

```

```
teclado_btn5.setText("5");
```

```
teclado_btn6.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn6.setText("6");
```

```
teclado_btn7.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn7.setText("7");
```

```
teclado_btn8.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn8.setText("8");
```

```
teclado_btn9.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn9.setText("9");
```

```
teclado_btnAceptar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnAceptar.setText("Aceptar");
```

```
teclado_btn0.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn0.setText("0");
```

```
teclado_btnAtras.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnAtras.setText("«");
```

```
teclado_btnLimpiar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnLimpiar.setText("C");
```



```

teclado_btn2.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn2.setText("2");

javax.swing.GroupLayout frame_servTecladoLayout = new
javax.swing.GroupLayout(frame_servTeclado.getContentPane());

frame_servTeclado.getContentPane().setLayout(frame_servTecladoLayout);

frame_servTecladoLayout.setHorizontalGroup(

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(frame_servTecladoLayout.createSequentialGroup()

        .addContainerGap()

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(frame_servTecladoLayout.createSequentialGroup()

        .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGroup(frame_servTecladoLayout.createSequentialGroup()

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

```

```

        .addComponent(teclado_btn7, javax.swing.GroupLayout.DEFAULT_SIZE, 66,
Short.MAX_VALUE)

        .addComponent(teclado_btn0, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
ADING)

        .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE))))

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
ADING)

        .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)))

```

```

        .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

);

frame_servTecladoLayout.setVerticalGroup(

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(teclado_btn7, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TR
AILING)

        .addComponent(teclado_btn0, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 46,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap())

);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

```

```

jLabel1.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
jLabel1.setText("CajeroUT");

jLabel2.setText("Matrícula:");

jLabel3.setText("Contraseña:");

btn_login.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Imagenes/login36.png"))); // NOI18N

jLabel4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Imagenes/lobo_escudo.jpg"))); // NOI18N

tBtn_tecladoMatricula.setText(" Teclado");

tBtn_tecladoPass.setText(" Teclado");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);

layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, 150, true)
                .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 150, true)
                .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, 150, true)
                .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE, 150, true)
            )
            .addContainerGap(150, true)
        )
);

```

```

        .addGroup(layout.createSequentialGroup())

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(jLabel3)

        .addComponent(jLabel2)

        .addComponent(btn_login, javax.swing.GroupLayout.PREFERRED_SIZE, 177,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(0, 1, Short.MAX_VALUE))

    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup())

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(txt_loginMatricula)

        .addComponent(txt_loginPass))

        .addGap(7, 7, 7)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

        .addComponent(tBtn_tecladoMatricula,
javax.swing.GroupLayout.DEFAULT_SIZE, 81, Short.MAX_VALUE)

        .addComponent(tBtn_tecladoPass, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))

        .addGap(18, 18, 18))

    .addGroup(layout.createSequentialGroup())

        .addComponent(jLabel1)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

        .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 190,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap())

```

```

);

layout.setVerticalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup())

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(layout.createSequentialGroup())

                    .addGap(8, 8, 8)

                    .addComponent(jLabel1)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                    .addComponent(jLabel2)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(tBtn_tecladoMatricula)

                    .addComponent(txt_loginMatricula, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                    .addComponent(jLabel3)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(tBtn_tecladoPass)

                    .addComponent(txt_loginPass, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                    .addComponent(btn_login)

                    .addGap(0, 0, Short.MAX_VALUE))

```

```

        .addGroup(layout.createSequentialGroup()

        .addContainerGap()

        .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

        .addContainerGap())

    );

    pack();
} // </editor-fold>

private void teclado_btn3ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

}

/**
 * @param args the command line arguments
 */

// Variables declaration - do not modify
public javax.swing.JButton btn_login;
public javax.swing.JFrame frame_servTeclado;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
public javax.swing.JToggleButton tBtn_tecladoMatricula;

```



```

public javax.swing.JToggleButton tBtn_tecladoPass;

public javax.swing.JButton teclado_btn0;

public javax.swing.JButton teclado_btn1;

public javax.swing.JButton teclado_btn2;

public javax.swing.JButton teclado_btn3;

public javax.swing.JButton teclado_btn4;

public javax.swing.JButton teclado_btn5;

public javax.swing.JButton teclado_btn6;

public javax.swing.JButton teclado_btn7;

public javax.swing.JButton teclado_btn8;

public javax.swing.JButton teclado_btn9;

public javax.swing.JButton teclado_btnAceptar;

public javax.swing.JButton teclado_btnAtras;

public javax.swing.JButton teclado_btnLimpiar;

public javax.swing.JTextField txt_loginMatricula;

public javax.swing.JPasswordField txt_loginPass;

// End of variables declaration
}

```

ServBuzon

```

package Vista;

public class ServBuzon extends javax.swing.JFrame {

    /**

```

```

* Creates new form ServBuzon

*/

public ServBuzon() {

    initComponents();

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jScrollPane1 = new javax.swing.JScrollPane();
    txtArea_txtBuzon = new javax.swing.JTextArea();
    btn_enviarBuzon = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jMenuBar1 = new javax.swing.JMenuBar();
    jMenu_servCaja = new javax.swing.JMenu();
    jMenuItem_servCaja = new javax.swing.JMenuItem();
    jMenuItem_servCreditoUT = new javax.swing.JMenuItem();
    jMenuItem_servLoboletos = new javax.swing.JMenuItem();
    jMenuItem_servEvaluacion = new javax.swing.JMenuItem();
    jMenuItem_servBuzon = new javax.swing.JMenuItem();
    jMenu2 = new javax.swing.JMenu();
    jMenuItem_consCalificaciones = new javax.swing.JMenuItem();
    jMenuItem_consHorario = new javax.swing.JMenuItem();

```

```
jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();
jMenuItem_ConstMapa = new javax.swing.JMenuItem();
jMenuItem_calendario = new javax.swing.JMenuItem();
jMenu_tramites = new javax.swing.JMenu();
jMenuItem_becas = new javax.swing.JMenuItem();
jMenuItem_constancias = new javax.swing.JMenuItem();
jMenu4 = new javax.swing.JMenu();
jMenuItem_correo = new javax.swing.JMenuItem();
jMenuItem_navega = new javax.swing.JMenuItem();
jMenuItem_soporte = new javax.swing.JMenuItem();
jMenu_cerrarSesion = new javax.swing.JMenu();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

txtArea_txtBuzon.setColumns(20);
txtArea_txtBuzon.setRows(5);
jScrollPane1.setViewportView(txtArea_txtBuzon);

btn_enviarBuzon.setText("Enviar");

jLabel1.setText("AVISO:");

jLabel2.setText("Tu mensaje se enviará de forma anónima.");

jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));
```

```
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_servCaja.setText("Servicios");
```

```
jMenuItem_servCaja.setText("Caja");
```

```
jMenu_servCaja.add(jMenuItem_servCaja);
```

```
jMenuItem_servCreditoUT.setText("CreditoUT");
```

```
jMenu_servCaja.add(jMenuItem_servCreditoUT);
```

```
jMenuItem_servLoboletos.setText("Loboletos");
```

```
jMenu_servCaja.add(jMenuItem_servLoboletos);
```

```
jMenuItem_servEvaluacion.setText("Evaluación de servicios");
```

```
jMenu_servCaja.add(jMenuItem_servEvaluacion);
```

```
jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
```

```
jMenu_servCaja.add(jMenuItem_servBuzon);
```

```
jMenuBar1.add(jMenu_servCaja);
```

```
jMenu2.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu2.setText("Consultas");
```

```
jMenuItem_consCalificaciones.setText("Calificaciones");
```

```
jMenu2.add(jMenuItem_consCalificaciones);
```

```
jMenuItem_consHorario.setText("Horario");
```

```
jMenu2.add(jMenuItem_consHorario);
```

```
jMenuItem_consEdoCuenta.setText("Estado de cuenta");
```

```
jMenu2.add(jMenuItem_consEdoCuenta);
```

```
jMenuItem_Con Mapa.setText("Mapa Universidad");
```

```
jMenu2.add(jMenuItem_Con Mapa);
```

```
jMenuItem_calendario.setText("Calendario");
```

```
jMenu2.add(jMenuItem_calendario);
```

```
jMenuBar1.add(jMenu2);
```

```
jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_tramites.setText("Trámites");
```

```
jMenuItem_becas.setText("Becas");
```

```
jMenu_tramites.add(jMenuItem_becas);
```

```
jMenuItem_constancias.setText("Constancias");
```

```
jMenu_tramites.add(jMenuItem_constancias);
```

```
jMenuBar1.add(jMenu_tramites);

jMenu4.setForeground(new java.awt.Color(255, 255, 255));
jMenu4.setText("LobosConect");

jMenuItem_correo.setText("Correo");
jMenu4.add(jMenuItem_correo);

jMenuItem_navega.setText("Navegar");
jMenu4.add(jMenuItem_navega);

jMenuItem_soporte.setText("Soporte");
jMenu4.add(jMenuItem_soporte);

jMenuBar1.add(jMenu4);

jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));
jMenu_cerrarSesion.setText("Cerrar sesión");
jMenuBar1.add(jMenu_cerrarSesion);

setJMenuBar(jMenuBar1);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
```

```

layout.setHorizontalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()

            .addContainerGap()

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addComponent(jScrollPane1)

                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

                    .addGap(0, 0, Short.MAX_VALUE)

                    .addComponent(btn_enviarBuzon))

                .addGroup(layout.createSequentialGroup()

                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(jLabel1)

                        .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 406,
javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addGap(0, 0, Short.MAX_VALUE)))

            .addContainerGap())

        );

layout.setVerticalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

            .addContainerGap()

            .addComponent(jLabel1)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 27, Short.MAX_VALUE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

```

```

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 158,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(btn_enviarBuzon)

        .addGap(18, 18, 18))

    );

    pack();
} // </editor-fold>

public static void main(String args[]) {

    /* Set the Nimbus look and feel */

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
    * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */

    try {

        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {

            if ("Nimbus".equals(info.getName())) {

                javax.swing.UIManager.setLookAndFeel(info.getClassName());

                break;

            }

        }

    } catch (ClassNotFoundException ex) {

```



```
java.util.logging.Logger.getLogger(ServBuzon.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
```

```
    } catch (InstantiationException ex) {
```

```
java.util.logging.Logger.getLogger(ServBuzon.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
```

```
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(ServBuzon.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(ServBuzon.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new ServBuzon().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

```
// Variables declaration - do not modify
```

```
public javax.swing.JButton btn_enviarBuzon;
```

```
private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu4;

private javax.swing.JMenuBar jMenuBar1;

public javax.swing.JMenuItem jMenuItem_ConsMapa;

public javax.swing.JMenuItem jMenuItem_becas;

public javax.swing.JMenuItem jMenuItem_calendario;

public javax.swing.JMenuItem jMenuItem_consCalificaciones;

public javax.swing.JMenuItem jMenuItem_consEdoCuenta;

public javax.swing.JMenuItem jMenuItem_consHorario;

public javax.swing.JMenuItem jMenuItem_constancias;

public javax.swing.JMenuItem jMenuItem_correo;

public javax.swing.JMenuItem jMenuItem_navega;

private javax.swing.JMenuItem jMenuItem_servBuzon;

public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

public javax.swing.JMenuItem jMenuItem_soporte;

public javax.swing.JMenu jMenu_cerrarSesion;

public javax.swing.JMenu jMenu_servCaja;

public javax.swing.JMenu jMenu_tramites;

private javax.swing.JScrollPane jScrollPane1;

public javax.swing.JTextArea txtArea_txtBuzon;
```

```
// End of variables declaration  
}
```

ServCreditoUT

```
package Vista;
```

```
public class ServCreditoUT extends javax.swing.JFrame {
```

```
    /**
```

```
     * Creates new form ServCreditoUT
```

```
    */
```

```
    public ServCreditoUT() {
```

```
        initComponents();
```

```
    }
```

```
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
    private void initComponents() {
```

```
        frame_servTeclado = new javax.swing.JFrame();
```

```
        teclado_btn1 = new javax.swing.JButton();
```

```
        teclado_btn3 = new javax.swing.JButton();
```

```
        teclado_btn4 = new javax.swing.JButton();
```

```
        teclado_btn5 = new javax.swing.JButton();
```

```
        teclado_btn6 = new javax.swing.JButton();
```

```
teclado_btn7 = new javax.swing.JButton();
teclado_btn8 = new javax.swing.JButton();
teclado_btn9 = new javax.swing.JButton();
teclado_btnAceptar = new javax.swing.JButton();
teclado_btn0 = new javax.swing.JButton();
teclado_btnAtras = new javax.swing.JButton();
teclado_btnLimpiar = new javax.swing.JButton();
teclado_btn2 = new javax.swing.JButton();
jScrollPane1 = new javax.swing.JScrollPane();
txtArea_ticket = new javax.swing.JTextArea();
jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
lbl_estadoCreditoUT = new javax.swing.JLabel();
jLabel3 = new javax.swing.JLabel();
cb_montoAbono = new javax.swing.JComboBox<>();
jLabel4 = new javax.swing.JLabel();
txt_numTarjeta = new javax.swing.JTextField();
tBtn_tecladoTarjeta = new javax.swing.JToggleButton();
jLabel6 = new javax.swing.JLabel();
pass_cve = new javax.swing.JPasswordField();
tBtn_tecladoCVE = new javax.swing.JToggleButton();
jLabel5 = new javax.swing.JLabel();
cb_vencimientoMes = new javax.swing.JComboBox<>();
cb_vencimientoAnio = new javax.swing.JComboBox<>();
jLabel7 = new javax.swing.JLabel();
```

```
btn_pagar = new javax.swing.JButton();

jLabel8 = new javax.swing.JLabel();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu_servCaja = new javax.swing.JMenu();

jMenuItem_servCaja = new javax.swing.JMenuItem();

jMenuItem_servCreditoUT = new javax.swing.JMenuItem();

jMenuItem_servLoboletos = new javax.swing.JMenuItem();

jMenuItem_servEvaluacion = new javax.swing.JMenuItem();

jMenuItem_servBuzon = new javax.swing.JMenuItem();

jMenu2 = new javax.swing.JMenu();

jMenuItem_consCalificaciones = new javax.swing.JMenuItem();

jMenuItem_consHorario = new javax.swing.JMenuItem();

jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();

jMenuItem_Con Mapa = new javax.swing.JMenuItem();

jMenuItem_calendario = new javax.swing.JMenuItem();

jMenu_tramites = new javax.swing.JMenu();

jMenuItem_becas = new javax.swing.JMenuItem();

jMenuItem_constancias = new javax.swing.JMenuItem();

jMenu4 = new javax.swing.JMenu();

jMenuItem_correo = new javax.swing.JMenuItem();

jMenuItem_navega = new javax.swing.JMenuItem();

jMenuItem_soporte = new javax.swing.JMenuItem();

jMenu_cerrarSesion = new javax.swing.JMenu();


frame_servTeclado.setTitle("ServTeclado");
```

```
frame_servTeclado.setMinimumSize(new java.awt.Dimension(250, 396));
```

```
teclado_btn1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn1.setText("1");
```

```
teclado_btn3.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn3.setText("3");
```

```
teclado_btn3.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        teclado_btn3ActionPerformed(evt);
```

```
    }
```

```
});
```

```
teclado_btn4.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn4.setText("4");
```

```
teclado_btn5.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn5.setText("5");
```

```
teclado_btn6.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn6.setText("6");
```

```
teclado_btn7.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn7.setText("7");
```

```
teclado_btn8.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn8.setText("8");

teclado_btn9.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn9.setText("9");

teclado_btnAceptar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btnAceptar.setText("Aceptar");

teclado_btn0.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn0.setText("0");

teclado_btnAtras.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btnAtras.setText("«");

teclado_btnLimpiar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btnLimpiar.setText("C");

teclado_btn2.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn2.setText("2");

javax.swing.GroupLayout frame_servTecladoLayout = new
javax.swing.GroupLayout(frame_servTeclado.getContentPane());

frame_servTeclado.getContentPane().setLayout(frame_servTecladoLayout);

frame_servTecladoLayout.setHorizontalGroup(
```

```

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addContainerGap()

    .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(frame_servTecladoLayout.createSequentialGroup()

            .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGroup(frame_servTecladoLayout.createSequentialGroup()

            .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                .addComponent(teclado_btn7, javax.swing.GroupLayout.DEFAULT_SIZE, 66,
Short.MAX_VALUE)

                .addComponent(teclado_btn0, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

```



```

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE))))

    .addGroup(frame_servTecladoLayout.createSequentialGroup())

    .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING))

    .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addContainerGap(14, Short.MAX_VALUE))

    );

    frame_servTecladoLayout.setVerticalGroup(

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(frame_servTecladoLayout.createSequentialGroup())

    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

```

```
.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(teclado_btn7, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,  
    javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```



```
lbl_estadoCreditoUT.setText("jLabel3");
```

```
jLabel3.setText("Abonar:");
```

```
cb_montoAbono.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "20",  
"50", "100", "200", "300", "400", "500" }));
```

```
jLabel4.setText("Número de tarjeta:");
```

```
tBtn_tecladoTarjeta.setText("Teclado");
```

```
jLabel6.setText("CVE:");
```

```
tBtn_tecladoCVE.setText("Teclado");
```

```
jLabel5.setText("Vencimiento:");
```

```
cb_vencimientoMes.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "1",  
"2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12" }));
```

```
cb_vencimientoAnio.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {  
"19", "20", "21", "22", "23", "24", "25", "26", "27", "28" }));
```

```
jLabel7.setText("/");
```

```
btn_pagar.setText("Pagar");
```

```
jLabel8.setText("Elige el monto a abonar y llena los datos:");
```

```
jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));
```

```
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_servCaja.setText("Servicios");
```

```
jMenuItem_servCaja.setText("Caja");
```

```
jMenu_servCaja.add(jMenuItem_servCaja);
```

```
jMenuItem_servCreditoUT.setText("CreditoUT");
```

```
jMenu_servCaja.add(jMenuItem_servCreditoUT);
```

```
jMenuItem_servLoboletos.setText("Loboletos");
```

```
jMenu_servCaja.add(jMenuItem_servLoboletos);
```

```
jMenuItem_servEvaluacion.setText("Evaluación de servicios");
```

```
jMenu_servCaja.add(jMenuItem_servEvaluacion);
```

```
jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
```

```
jMenu_servCaja.add(jMenuItem_servBuzon);
```

```
jMenuBar1.add(jMenu_servCaja);

jMenu2.setForeground(new java.awt.Color(255, 255, 255));
jMenu2.setText("Consultas");

jMenuItem_consCalificaciones.setText("Calificaciones");
jMenu2.add(jMenuItem_consCalificaciones);

jMenuItem_consHorario.setText("Horario");
jMenu2.add(jMenuItem_consHorario);

jMenuItem_consEdoCuenta.setText("Estado de cuenta");
jMenu2.add(jMenuItem_consEdoCuenta);

jMenuItem_ConsMapa.setText("Mapa Universidad");
jMenu2.add(jMenuItem_ConsMapa);

jMenuItem_calendario.setText("Calendario");
jMenu2.add(jMenuItem_calendario);

jMenuBar1.add(jMenu2);

jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));
jMenu_tramites.setText("Trámites");
```

```
jMenuItem_becas.setText("Becas");  
jMenu_tramites.add(jMenuItem_becas);  
  
jMenuItem_constancias.setText("Constancias");  
jMenu_tramites.add(jMenuItem_constancias);  
  
jMenuBar1.add(jMenu_tramites);  
  
jMenu4.setForeground(new java.awt.Color(255, 255, 255));  
jMenu4.setText("LobosConect");  
  
jMenuItem_correo.setText("Correo");  
jMenu4.add(jMenuItem_correo);  
  
jMenuItem_navega.setText("Navegar");  
jMenu4.add(jMenuItem_navega);  
  
jMenuItem_soporte.setText("Soporte");  
jMenu4.add(jMenuItem_soporte);  
  
jMenuBar1.add(jMenu4);  
  
jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));  
jMenu_cerrarSesion.setText("Cerrar sesión");  
jMenuBar1.add(jMenu_cerrarSesion);
```

```

setJMenuBar(jMenuBar1);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(btn_pagar, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addContainerGap()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(10, 10, 10)
                    .addComponent(jLabel5)
                    .addGap(10, 10, 10)
                    .addComponent(cb_vencimientoMes, javax.swing.GroupLayout.PREFERRED_SIZE, 43, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(10, 10, 10)
                    .addComponent(jLabel7)
                    .addGap(10, 10, 10)
                    .addComponent(cb_vencimientoAnio, javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE)
                )
                .addGroup(layout.createSequentialGroup()
                    .addGap(10, 10, 10)
                    .addComponent(pass_cve, javax.swing.GroupLayout.PREFERRED_SIZE, 133, javax.swing.GroupLayout.PREFERRED_SIZE)
                )
            )
        )
    );

```



```

        .addComponent(txt_numTarjeta, javax.swing.GroupLayout.DEFAULT_SIZE, 135,
Short.MAX_VALUE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addComponent(tBtn_tecladoTarjeta,
javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT_SIZE, 72,
Short.MAX_VALUE)

            .addComponent(tBtn_tecladoCVE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

        .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addComponent(jLabel1)

            .addGroup(layout.createSequentialGroup())

                .addComponent(jLabel2)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                .addComponent(lbl_estadoCreditoUT))

            .addComponent(jLabel4)

            .addComponent(jLabel6)

            .addComponent(jLabel8)

            .addGroup(layout.createSequentialGroup())

                .addComponent(jLabel3)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addComponent(cb_montoAbono, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addGap(0, 0, Short.MAX_VALUE)))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 246,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addContainerGap()

    );

    layout.setVerticalGroup(

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()

            .addContainerGap()

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(layout.createSequentialGroup()

                    .addGap(8, 8, 8)

                    .addComponent(jLabel1)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel2)

                    .addComponent(lbl_estadoCreditoUT))

                    .addGap(18, 18, 18)

                    .addComponent(jLabel8)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 20,
Short.MAX_VALUE)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                    .addComponent(jLabel3)

                    .addComponent(cb_montoAbono, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                    .addComponent(jLabel4)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(txt_numTarjeta, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(tBtn_tecladoTarjeta))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jLabel6)

.addGap(2, 2, 2)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(pass_cve, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(tBtn_tecladoCVE))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(cb_vencimientoMes, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(jLabel7)

    .addComponent(cb_vencimientoAnio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(jLabel5))

.addGap(34, 34, 34)

.addComponent(btn_pagar))

.addComponent(jScrollPane1))

.addContainerGap())

);

```

```

    pack();
} // </editor-fold>

private void teclado_btn3ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {

    /* Set the Nimbus look and feel */

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */

    try {

        for (javax.swing.UIManager.LookAndFeelInfo info :
            javax.swing.UIManager.getInstalledLookAndFeels()) {

            if ("Nimbus".equals(info.getName())) {

                javax.swing.UIManager.setLookAndFeel(info.getClassName());

                break;

            }

        }

    } catch (ClassNotFoundException ex) {

```

```
java.util.logging.Logger.getLogger(ServCreditoUT.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (InstantiationException ex) {
```

```
java.util.logging.Logger.getLogger(ServCreditoUT.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(ServCreditoUT.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(ServCreditoUT.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new ServCreditoUT().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

```
// Variables declaration - do not modify
```

```
public javax.swing.JButton btn_pagar;
```

```
public javax.swing.JComboBox<String> cb_montoAbono;

public javax.swing.JComboBox<String> cb_vencimientoAnio;

public javax.swing.JComboBox<String> cb_vencimientoMes;

public javax.swing.JFrame frame_servTeclado;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu4;

private javax.swing.JMenuBar jMenuBar1;

public javax.swing.JMenuItem jMenuItem_ConsMapa;

public javax.swing.JMenuItem jMenuItem_becas;

public javax.swing.JMenuItem jMenuItem_calendario;

public javax.swing.JMenuItem jMenuItem_consCalificaciones;

public javax.swing.JMenuItem jMenuItem_consEdoCuenta;

public javax.swing.JMenuItem jMenuItem_consHorario;

public javax.swing.JMenuItem jMenuItem_constancias;

public javax.swing.JMenuItem jMenuItem_correo;

public javax.swing.JMenuItem jMenuItem_navega;

public javax.swing.JMenuItem jMenuItem_servBuzon;
```

```
public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

public javax.swing.JMenuItem jMenuItem_soporte;

public javax.swing.JMenu jMenu_cerrarSesion;

private javax.swing.JMenu jMenu_servCaja;

public javax.swing.JMenu jMenu_tramites;

private javax.swing.JScrollPane jScrollPane1;

public javax.swing.JLabel lbl_estadoCreditoUT;

public javax.swing.JPasswordField pass_cve;

public javax.swing.JToggleButton tBtn_tecladoCVE;

public javax.swing.JToggleButton tBtn_tecladoTarjeta;

public javax.swing.JButton teclado_btn0;

public javax.swing.JButton teclado_btn1;

public javax.swing.JButton teclado_btn2;

public javax.swing.JButton teclado_btn3;

public javax.swing.JButton teclado_btn4;

public javax.swing.JButton teclado_btn5;

public javax.swing.JButton teclado_btn6;

public javax.swing.JButton teclado_btn7;

public javax.swing.JButton teclado_btn8;

public javax.swing.JButton teclado_btn9;

public javax.swing.JButton teclado_btnAceptar;

public javax.swing.JButton teclado_btnAtras;
```

```

    public javax.swing.JButton teclado_btnLimpiar;

    public javax.swing.JTextArea txtArea_ticket;

    public javax.swing.JTextField txt_numTarjeta;

    // End of variables declaration
}

```

ServEvaluacion

```
package Vista;
```

```
public class ServEvaluacion extends javax.swing.JFrame {
```

```
    /**
```

```
     * Creates new form ServEvaluacion
```

```
    */
```

```
    public ServEvaluacion() {
```

```
        initComponents();
```

```
    }
```

```
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
```

```
    radiogroupAtencion = new javax.swing.ButtonGroup();
```

```
    radiogroupServicio = new javax.swing.ButtonGroup();
```



```
cb_servicio = new javax.swing.JComboBox<>();

jLabel1 = new javax.swing.JLabel();

jpanel_opciones = new javax.swing.JPanel();

jLabel2 = new javax.swing.JLabel();

rdBtn_AtMalo = new javax.swing.JRadioButton();

rdBtn_AtRegular = new javax.swing.JRadioButton();

rdBtn_AtBueno = new javax.swing.JRadioButton();

rdBtn_AtExcelente = new javax.swing.JRadioButton();

jLabel3 = new javax.swing.JLabel();

rdBtn_ServMalo = new javax.swing.JRadioButton();

rdBtn_ServReglar = new javax.swing.JRadioButton();

rdBtn_ServBueno = new javax.swing.JRadioButton();

rdBtn_ServExcelente = new javax.swing.JRadioButton();

jLabel4 = new javax.swing.JLabel();

cb_usaria = new javax.swing.JComboBox<>();

btn_enviarEvaluacion = new javax.swing.JButton();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu_servCaja = new javax.swing.JMenu();

jMenuItem_servCaja = new javax.swing.JMenuItem();

jMenuItem_servCreditoUT = new javax.swing.JMenuItem();

jMenuItem_servLoboletos = new javax.swing.JMenuItem();

jMenuItem_servEvaluacion = new javax.swing.JMenuItem();

jMenuItem_servBuzon = new javax.swing.JMenuItem();

jMenu2 = new javax.swing.JMenu();

jMenuItem_consCalificaciones = new javax.swing.JMenuItem();
```

```

jMenuItem_consHorario = new javax.swing.JMenuItem();

jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();

jMenuItem_ConstMapa = new javax.swing.JMenuItem();

jMenuItem_calendario = new javax.swing.JMenuItem();

jMenu_tramites = new javax.swing.JMenu();

jMenuItem_becas = new javax.swing.JMenuItem();

jMenuItem_constancias = new javax.swing.JMenuItem();

jMenu4 = new javax.swing.JMenu();

jMenuItem_correo = new javax.swing.JMenuItem();

jMenuItem_navega = new javax.swing.JMenuItem();

jMenuItem_soporte = new javax.swing.JMenuItem();

jMenu_cerrarSesion = new javax.swing.JMenu();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


cb_servicio.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {
"Seleccionar", "Cafetería", "Transporte", "Servicios médicos", "Cajas", "CajerosUT" }));


jLabel1.setText("Servicios de la UT:");


jpanel_opciones.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));


jLabel2.setText("¿Cómo Evaluaría la atención de este servicio?");


radiogroupAtencion.add(rdBtn_AtMalo);

```

```
rdBtn_AtMalo.setText("Malo");
```

```
radiogroupAtencion.add(rdBtn_AtRegular);
```

```
rdBtn_AtRegular.setText("Regular");
```

```
radiogroupAtencion.add(rdBtn_AtBueno);
```

```
rdBtn_AtBueno.setText("Bueno");
```

```
radiogroupAtencion.add(rdBtn_AtExcelente);
```

```
rdBtn_AtExcelente.setText("Excelente");
```

```
jLabel3.setText("¿Cómo evalúa el servicio brindado?");
```

```
radiogroupServicio.add(rdBtn_ServMalo);
```

```
rdBtn_ServMalo.setText("Malo");
```

```
radiogroupServicio.add(rdBtn_ServRegular);
```

```
rdBtn_ServRegular.setText("Regular");
```

```
radiogroupServicio.add(rdBtn_ServBueno);
```

```
rdBtn_ServBueno.setText("Bueno");
```

```
radiogroupServicio.add(rdBtn_ServExcelente);
```

```
rdBtn_ServExcelente.setText("Excelente");
```

```
jLabel4.setText("¿Usted volvería a hacer uso de este servicio?");
```

```
cb_usaria.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Si", "No",  
"Solo por necesidad" }));
```

```
    javax.swing.GroupLayout jPanel_opcionesLayout = new
    javax.swing.GroupLayout(jPanel_opciones);
```

```
jpanel_opciones.setLayout(jpanel_opcionesLayout);
```

```
jpanel_opcionesLayout.setHorizontalGroup(
```

```
jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(jpanel_opcionesLayout.createSequentialGroup())
```

```
.addContainerGap()
```

```
.addGroup(jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
```

```
.addGroup(jpanel_opcionesLayout.createSequentialGroup())
```

```
.addComponent(rdBtn_ServMalo)
```

```
.addGap(10, 10, 10)
```

```
.addComponent(rdBtn_ServReglar)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
```

```
.addComponent(rdBtn_ServBueno)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
```

```
.addComponent(rdBtn_ServExcelente))
```

```

.addGroup(jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(jpanel_opcionesLayout.createSequentialGroup()

        .addGap(10, 10, 10)

        .addComponent(rdBtn_AtMalo)

        .addGap(10, 10, 10)

        .addComponent(rdBtn_AtRegular)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(rdBtn_AtBueno)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(rdBtn_AtExcelente))

    .addComponent(jLabel2)

    .addComponent(jLabel3)))

.addGroup(jpanel_opcionesLayout.createSequentialGroup()

    .addComponent(jLabel4)

    .addGap(18, 18, 18)

    .addComponent(cb_usaria, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

.addContainerGap(33, Short.MAX_VALUE))

);

jpanel_opcionesLayout.setVerticalGroup(

    jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(jpanel_opcionesLayout.createSequentialGroup()

        .addContainerGap()

        .addComponent(jLabel2)

```

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addGroup(jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(rdBtn_AtMalo)

        .addComponent(rdBtn_AtRegular)

        .addComponent(rdBtn_AtBueno)

        .addComponent(rdBtn_AtExcelente))

    .addGap(18, 18, 18)

    .addComponent(jLabel3)

    .addGap(18, 18, 18)

    .addGroup(jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(rdBtn_ServMalo)

        .addComponent(rdBtn_ServReglar)

        .addComponent(rdBtn_ServBueno)

        .addComponent(rdBtn_ServExcelente))

    .addGap(18, 18, 18)

    .addGroup(jpanel_opcionesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(jLabel4)

        .addComponent(cb_usaria, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addContainerGap(12, Short.MAX_VALUE))

);

```

```
btn_enviarEvaluacion.setText("Enviar");

jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));

jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));
jMenu_servCaja.setText("Servicios");

jMenuItem_servCaja.setText("Caja");
jMenu_servCaja.add(jMenuItem_servCaja);

jMenuItem_servCreditoUT.setText("CreditoUT");
jMenu_servCaja.add(jMenuItem_servCreditoUT);

jMenuItem_servLoboletos.setText("Loboletos");
jMenu_servCaja.add(jMenuItem_servLoboletos);

jMenuItem_servEvaluacion.setText("Evaluación de servicios");
jMenu_servCaja.add(jMenuItem_servEvaluacion);

jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
jMenu_servCaja.add(jMenuItem_servBuzon);

jMenuBar1.add(jMenu_servCaja);
```

```
jMenu2.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu2.setText("Consultas");
```

```
jMenuItem_consCalificaciones.setText("Calificaciones");
```

```
jMenu2.add(jMenuItem_consCalificaciones);
```

```
jMenuItem_consHorario.setText("Horario");
```

```
jMenu2.add(jMenuItem_consHorario);
```

```
jMenuItem_consEdoCuenta.setText("Estado de cuenta");
```

```
jMenu2.add(jMenuItem_consEdoCuenta);
```

```
jMenuItem_Con Mapa.setText("Mapa Universidad");
```

```
jMenu2.add(jMenuItem_Con Mapa);
```

```
jMenuItem_calendario.setText("Calendario");
```

```
jMenu2.add(jMenuItem_calendario);
```

```
jMenuBar1.add(jMenu2);
```

```
jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_tramites.setText("Trámites");
```

```
jMenuItem_becas.setText("Becas");
```

```
jMenu_tramites.add(jMenuItem_becas);
```



```
jMenuItem_constancias.setText("Constancias");  
jMenu_tramites.add(jMenuItem_constancias);  
  
jMenuBar1.add(jMenu_tramites);  
  
jMenu4.setForeground(new java.awt.Color(255, 255, 255));  
jMenu4.setText("LobosConect");  
  
jMenuItem_correo.setText("Correo");  
jMenu4.add(jMenuItem_correo);  
  
jMenuItem_navega.setText("Navegar");  
jMenu4.add(jMenuItem_navega);  
  
jMenuItem_soporte.setText("Soporte");  
jMenu4.add(jMenuItem_soporte);  
  
jMenuBar1.add(jMenu4);  
  
jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));  
jMenu_cerrarSesion.setText("Cerrar sesión");  
jMenuBar1.add(jMenu_cerrarSesion);  
  
setJMenuBar(jMenuBar1);
```

```

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()

            .addGap(10, 10, 10)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addComponent(jpanel_opciones, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                .addGroup(layout.createSequentialGroup()

                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(cb_servicio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel1))

                    .addGap(10, 10, 10))

                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

                    .addGap(10, 10, 10)

                    .addComponent(btn_enviarEvaluacion)))

            .addGap(10, 10, 10))

        .addGap(10, 10, 10)

    );

layout.setVerticalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()

            .addGap(10, 10, 10)

            .addGroup(layout.createSequentialGroup()

                .addComponent(jpanel_opciones, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                .addGap(10, 10, 10)

                .addComponent(cb_servicio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addGap(10, 10, 10)

                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addGap(10, 10, 10)

                .addComponent(btn_enviarEvaluacion, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addGap(10, 10, 10))

            .addGap(10, 10, 10))

    );

```

```

        .addComponent(jLabel1)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(cb_servicio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(18, 18, 18)

        .addComponent(jpanel_opciones, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(btn_enviarEvaluacion)

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

    );

```

```

    pack();

```

```

} // </editor-fold>

```

```

/**

```

```

 * @param args the command line arguments

```

```

 */

```

```

public static void main(String args[]) {

```

```

    /* Set the Nimbus look and feel */

```

```

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

```

```

    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

```

```

     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

```

```

    */

```

```

    try {

```

```

        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {

            if ("Nimbus".equals(info.getName())) {

                javax.swing.UIManager.setLookAndFeel(info.getClassName());

                break;

            }

        }

    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(ServEvaluacion.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(ServEvaluacion.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(ServEvaluacion.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(ServEvaluacion.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    }

}

//</editor-fold>

/* Create and display the form */

java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

```

```

        new ServEvaluacion().setVisible(true);

    }

});

}

// Variables declaration - do not modify

public javax.swing.JButton btn_enviarEvaluacion;

public javax.swing.JComboBox<String> cb_servicio;

public javax.swing.JComboBox<String> cb_usaria;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu4;

private javax.swing.JMenuBar jMenuBar1;

public javax.swing.JMenuItem jMenuItem_ConsMapa;

public javax.swing.JMenuItem jMenuItem_becas;

public javax.swing.JMenuItem jMenuItem_calendario;

public javax.swing.JMenuItem jMenuItem_consCalificaciones;

public javax.swing.JMenuItem jMenuItem_consEdoCuenta;

public javax.swing.JMenuItem jMenuItem_consHorario;

public javax.swing.JMenuItem jMenuItem_constancias;

public javax.swing.JMenuItem jMenuItem_correo;

public javax.swing.JMenuItem jMenuItem_navega;

```

```
public javax.swing.JMenuItem jMenuItem_servBuzon;

public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

public javax.swing.JMenuItem jMenuItem_soporte;

public javax.swing.JMenu jMenu_cerrarSesion;

private javax.swing.JMenu jMenu_servCaja;

public javax.swing.JMenu jMenu_tramites;

public javax.swing.JPanel jpanel_opciones;

private javax.swing.ButtonGroup radiogroupAtencion;

private javax.swing.ButtonGroup radiogroupServicio;

public javax.swing.JRadioButton rdBtn_AtBueno;

public javax.swing.JRadioButton rdBtn_AtExcelente;

public javax.swing.JRadioButton rdBtn_AtMalo;

public javax.swing.JRadioButton rdBtn_AtRegular;

public javax.swing.JRadioButton rdBtn_ServBueno;

public javax.swing.JRadioButton rdBtn_ServExcelente;

public javax.swing.JRadioButton rdBtn_ServMalo;

public javax.swing.JRadioButton rdBtn_ServReglar;

// End of variables declaration

}
```

Soporte

```
package Vista;
```

```
public class ServSoporte extends javax.swing.JFrame {
```

```
    /**
```

```
     * Creates new form ServSoporte
```

```
     */
```

```
    public ServSoporte() {
```

```
        initComponents();
```

```
    }
```

```
    @SuppressWarnings("unchecked")
```

```
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
    private void initComponents() {
```

```
        lbl_mensajeSoporte = new javax.swing.JLabel();
```

```
        jLabel2 = new javax.swing.JLabel();
```

```
        jScrollPane1 = new javax.swing.JScrollPane();
```

```
        txtArea_soporte = new javax.swing.JTextArea();
```

```
        btn_enviarSoporte = new javax.swing.JButton();
```

```
        jMenuBar1 = new javax.swing.JMenuBar();
```

```
        jMenu_servCaja = new javax.swing.JMenu();
```

```
        jMenuItem_servCaja = new javax.swing.JMenuItem();
```

```
        jMenuItem_servCreditoUT = new javax.swing.JMenuItem();
```

```
jMenuItem_servLoboletos = new javax.swing.JMenuItem();
jMenuItem_servEvaluacion = new javax.swing.JMenuItem();
jMenuItem_servBuzon = new javax.swing.JMenuItem();
jMenu2 = new javax.swing.JMenu();
jMenuItem_consCalificaciones = new javax.swing.JMenuItem();
jMenuItem_consHorario = new javax.swing.JMenuItem();
jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();
jMenuItem_ConsMapa = new javax.swing.JMenuItem();
jMenuItem_calendario = new javax.swing.JMenuItem();
jMenu_tramites = new javax.swing.JMenu();
jMenuItem_becas = new javax.swing.JMenuItem();
jMenuItem_constancias = new javax.swing.JMenuItem();
jMenu4 = new javax.swing.JMenu();
jMenuItem_correo = new javax.swing.JMenuItem();
jMenuItem_navega = new javax.swing.JMenuItem();
jMenuItem_soporte = new javax.swing.JMenuItem();
jMenu_cerrarSesion = new javax.swing.JMenu();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

lbl_mensajeSoporte.setText("¿Tienes algún problema con CajeroUT?");

jLabel2.setText("Mandanos un mensaje para atenderlo a la brevedad:");

txtArea_soporte.setColumns(20);
```



```
txtArea_soporte.setRows(5);

jScrollPane1.setViewportViewView(txtArea_soporte);


btn_enviarSoporte.setText("Enviar");


jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));


jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));
jMenu_servCaja.setText("Servicios");


jMenuItem_servCaja.setText("Caja");
jMenu_servCaja.add(jMenuItem_servCaja);


jMenuItem_servCreditoUT.setText("CreditoUT");
jMenu_servCaja.add(jMenuItem_servCreditoUT);


jMenuItem_servLoboletos.setText("Loboletos");
jMenu_servCaja.add(jMenuItem_servLoboletos);


jMenuItem_servEvaluacion.setText("Evaluación de servicios");
jMenu_servCaja.add(jMenuItem_servEvaluacion);


jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
jMenu_servCaja.add(jMenuItem_servBuzon);
```

```
jMenuBar1.add(jMenu_servCaja);

jMenu2.setForeground(new java.awt.Color(255, 255, 255));
jMenu2.setText("Consultas");

jMenuItem_consCalificaciones.setText("Calificaciones");
jMenu2.add(jMenuItem_consCalificaciones);

jMenuItem_consHorario.setText("Horario");
jMenu2.add(jMenuItem_consHorario);

jMenuItem_consEdoCuenta.setText("Estado de cuenta");
jMenu2.add(jMenuItem_consEdoCuenta);

jMenuItem_ConsMapa.setText("Mapa Universidad");
jMenu2.add(jMenuItem_ConsMapa);

jMenuItem_calendario.setText("Calendario");
jMenu2.add(jMenuItem_calendario);

jMenuBar1.add(jMenu2);

jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));
jMenu_tramites.setText("Trámites");
```

```
jMenuItem_becas.setText("Becas");  
jMenu_tramites.add(jMenuItem_becas);  
  
jMenuItem_constancias.setText("Constancias");  
jMenu_tramites.add(jMenuItem_constancias);  
  
jMenuBar1.add(jMenu_tramites);  
  
jMenu4.setForeground(new java.awt.Color(255, 255, 255));  
jMenu4.setText("LobosConect");  
  
jMenuItem_correo.setText("Correo");  
jMenu4.add(jMenuItem_correo);  
  
jMenuItem_navega.setText("Navegar");  
jMenu4.add(jMenuItem_navega);  
  
jMenuItem_soporte.setText("Soporte");  
jMenu4.add(jMenuItem_soporte);  
  
jMenuBar1.add(jMenu4);  
  
jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));  
jMenu_cerrarSesion.setText("Cerrar sesión");
```

```

jMenuBar1.add(jMenu_cerrarSesion);

setJMenuBar(jMenuBar1);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jScrollPane1)
                .addGroup(layout.createSequentialGroup()
                    .addGap(10, 10, 10)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(lbl_mensajeSoporte)
                        .addComponent(jLabel2))
                    .addGap(10, 10, 10)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                        .addComponent(btn_enviarSoporte)
                        .addGap(10, 10, 10)))
                .addGap(10, 10, 10))
            .addContainerGap(10, Short.MAX_VALUE))
        .addGroup(layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addComponent(btn_cerrar)
            .addContainerGap(10, Short.MAX_VALUE))
    );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 100, false)
            .addGap(10, 10, 10)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(lbl_mensajeSoporte)
                .addComponent(jLabel2)
                .addGap(10, 10, 10)
                .addComponent(btn_enviarSoporte)
                .addGap(10, 10, 10)
                .addComponent(btn_cerrar))
            .addContainerGap(10, Short.MAX_VALUE))
    );

```

```

        .addGap(19, 19, 19)

        .addComponent(lbl_mensajeSoporte)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(jLabel2)

        .addGap(18, 18, 18)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 148,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(18, 18, 18)

        .addComponent(btn_enviarSoporte)

        .addContainerGap(14, Short.MAX_VALUE))

    );

```

```

    pack();

```

```

} // </editor-fold>

```

```

/**

```

```

 * @param args the command line arguments

```

```

 */

```

```

public static void main(String args[]) {

```

```

    /* Set the Nimbus look and feel */

```

```

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

```

```

    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

```

```

     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

```

```

    */

```

```

    try {

```

```

        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {

            if ("Nimbus".equals(info.getName())) {

                javax.swing.UIManager.setLookAndFeel(info.getClassName());

                break;

            }

        }

    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(ServSoporte.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);

        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(ServSoporte.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);

        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(ServSoporte.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);

        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(ServSoporte.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);

        }

    }

//</editor-fold>

    /* Create and display the form */

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

```

```

        new ServSoporte().setVisible(true);
    }
});
}

// Variables declaration - do not modify

public javax.swing.JButton btn_enviarSoporte;

private javax.swing.JLabel jLabel2;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu4;

private javax.swing.JMenuBar jMenuBar1;

public javax.swing.JMenuItem jMenuItem_ConsMapa;

public javax.swing.JMenuItem jMenuItem_becas;

public javax.swing.JMenuItem jMenuItem_calendario;

public javax.swing.JMenuItem jMenuItem_consCalificaciones;

public javax.swing.JMenuItem jMenuItem_consEdoCuenta;

public javax.swing.JMenuItem jMenuItem_consHorario;

public javax.swing.JMenuItem jMenuItem_constancias;

public javax.swing.JMenuItem jMenuItem_correo;

public javax.swing.JMenuItem jMenuItem_navega;

public javax.swing.JMenuItem jMenuItem_servBuzon;

public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

```

```

    public javax.swing.JMenuItem jMenuItem_soporte;

    public javax.swing.JMenu jMenu_cerrarSesion;

    private javax.swing.JMenu jMenu_servCaja;

    public javax.swing.JMenu jMenu_tramites;

    private javax.swing.JScrollPane jScrollPane1;

    public javax.swing.JLabel lbl_mensajeSoporte;

    public javax.swing.JTextArea txtArea_soporte;

    // End of variables declaration
}

```

TramConstancia

```

package Vista;

```

```

public class TramConstancia extends javax.swing.JFrame {

```

```

    /**

```

```

     * Creates new form TramConstancia

```

```

    */

```

```

    public TramConstancia() {

```

```

        initComponents();

```

```

    }

```

```

    @SuppressWarnings("unchecked")

```

```

    // <editor-fold defaultstate="collapsed" desc="Generated Code">

```

```

    private void initComponents() {

```



```
buttonGroup1 = new javax.swing.ButtonGroup();
frame_servTeclado = new javax.swing.JFrame();
teclado_btn1 = new javax.swing.JButton();
teclado_btn3 = new javax.swing.JButton();
teclado_btn4 = new javax.swing.JButton();
teclado_btn5 = new javax.swing.JButton();
teclado_btn6 = new javax.swing.JButton();
teclado_btn7 = new javax.swing.JButton();
teclado_btn8 = new javax.swing.JButton();
teclado_btn9 = new javax.swing.JButton();
teclado_btnAceptar = new javax.swing.JButton();
teclado_btn0 = new javax.swing.JButton();
teclado_btnAtras = new javax.swing.JButton();
teclado_btnLimpiar = new javax.swing.JButton();
teclado_btn2 = new javax.swing.JButton();
jLabel2 = new javax.swing.JLabel();
rd_pagoCreditoUT = new javax.swing.JRadioButton();
rd_pagoTarjeta = new javax.swing.JRadioButton();
jLabel3 = new javax.swing.JLabel();
txt_numTarjeta = new javax.swing.JTextField();
tBtn_tecladoTarjeta = new javax.swing.JToggleButton();
jLabel6 = new javax.swing.JLabel();
pass_cve = new javax.swing.JPasswordField();
tBtn_tecladoCVE = new javax.swing.JToggleButton();
```

```
jLabel4 = new javax.swing.JLabel();

cb_vencimientoMes = new javax.swing.JComboBox<>();

cb_vencimientoAnio = new javax.swing.JComboBox<>();

btn_pagar = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

txtArea_ticket = new javax.swing.JTextArea();

jLabel1 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel7 = new javax.swing.JLabel();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu_servCaja = new javax.swing.JMenu();

jMenuItem_servCaja = new javax.swing.JMenuItem();

jMenuItem_servCreditoUT = new javax.swing.JMenuItem();

jMenuItem_servLoboletos = new javax.swing.JMenuItem();

jMenuItem_servEvaluacion = new javax.swing.JMenuItem();

jMenuItem_servBuzon = new javax.swing.JMenuItem();

jMenu2 = new javax.swing.JMenu();

jMenuItem_consCalificaciones = new javax.swing.JMenuItem();

jMenuItem_consHorario = new javax.swing.JMenuItem();

jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();

jMenuItem_ConsMapa = new javax.swing.JMenuItem();

jMenuItem_calendario = new javax.swing.JMenuItem();

jMenu_tramites = new javax.swing.JMenu();

jMenuItem_becas = new javax.swing.JMenuItem();

jMenuItem_constancias = new javax.swing.JMenuItem();
```

```
jMenu4 = new javax.swing.JMenu();

jMenuItem_correo = new javax.swing.JMenuItem();

jMenuItem_navega = new javax.swing.JMenuItem();

jMenuItem_soporte = new javax.swing.JMenuItem();

jMenu_cerrarSesion = new javax.swing.JMenu();


frame_servTeclado.setTitle("ServTeclado");

frame_servTeclado.setMinimumSize(new java.awt.Dimension(250, 396));


teclado_btn1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn1.setText("1");


teclado_btn3.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn3.setText("3");
teclado_btn3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        teclado_btn3ActionPerformed(evt);
    }
});


teclado_btn4.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn4.setText("4");


teclado_btn5.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn5.setText("5");
```

```
teclado_btn6.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn6.setText("6");
```

```
teclado_btn7.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn7.setText("7");
```

```
teclado_btn8.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn8.setText("8");
```

```
teclado_btn9.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn9.setText("9");
```

```
teclado_btnAceptar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnAceptar.setText("Aceptar");
```

```
teclado_btn0.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn0.setText("0");
```

```
teclado_btnAtras.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnAtras.setText("«");
```

```
teclado_btnLimpiar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnLimpiar.setText("C");
```

```

teclado_btn2.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N

teclado_btn2.setText("2");


    javax.swing.GroupLayout frame_servTecladoLayout = new
javax.swing.GroupLayout(frame_servTeclado.getContentPane());

    frame_servTeclado.getContentPane().setLayout(frame_servTecladoLayout);

    frame_servTecladoLayout.setHorizontalGroup(

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(frame_servTecladoLayout.createSequentialGroup()

            .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(frame_servTecladoLayout.createSequentialGroup()

                    .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                    .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                    .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addGroup(frame_servTecladoLayout.createSequentialGroup()

                        .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                            .addComponent(teclado_btn7, javax.swing.GroupLayout.DEFAULT_SIZE, 66,
Short.MAX_VALUE)

```

```

        .addComponent(teclado_btn0, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE))))

.addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE))

```

```

        .addContainerGap(14, Short.MAX_VALUE))

    );

    frame_servTecladoLayout.setVerticalGroup(

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
                SELINE)

                    .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
                    javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
                    javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
                    javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
                SELINE)

                    .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
                    javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
                    javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
                    javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
                SELINE)

```

```

        .addComponent(teclado_btn7, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING))

        .addComponent(teclado_btn0, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE))

        .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 46,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addContainerGap())

    );

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel2.setText("Forma de pago:");

```



```
buttonGroup1.add(rd_pagoCreditoUT);

rd_pagoCreditoUT.setSelected(true);

rd_pagoCreditoUT.setText("CreditoUT");


buttonGroup1.add(rd_pagoTarjeta);

rd_pagoTarjeta.setText("Tarjeta");


jLabel3.setText("Número de tarjeta:");


tBtn_tecladoTarjeta.setText("Teclado");


jLabel6.setText("CVE:");


tBtn_tecladoCVE.setText("Teclado");


jLabel4.setText("Vencimiento:");


cb_vencimientoMes.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "1",
"2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12" }));


cb_vencimientoAnio.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {
"19", "20", "21", "22", "23", "24", "25", "26", "27", "28" }));


btn_pagar.setText("Pagar");


txtArea_ticket.setColumns(20);
```

```

txtArea_ticket.setRows(5);

txtArea_ticket.setText("PREVISUALIZACIÓN\nTicket de
pago\n\n\n\n\n\n\n\n\n\n\n*Presenta en este ticket\nen servicios escolares para\npoder
recoger tu constancia.");

jScrollPane1.setViewportViewView(txtArea_ticket);


jLabel1.setText("Pago de constancia");


jLabel5.setText("*Si tienes algún adeudo el sistema no te");


jLabel7.setText("dejará realizar el pago");


jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));


jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));
jMenu_servCaja.setText("Servicios");


jMenuItem_servCaja.setText("Caja");
jMenu_servCaja.add(jMenuItem_servCaja);


jMenuItem_servCreditoUT.setText("CreditoUT");
jMenu_servCaja.add(jMenuItem_servCreditoUT);


jMenuItem_servLoboletos.setText("Loboletos");
jMenu_servCaja.add(jMenuItem_servLoboletos);

```

```
jMenuItem_servEvaluacion.setText("Evaluación de servicios");
```

```
jMenu_servCaja.add(jMenuItem_servEvaluacion);
```

```
jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
```

```
jMenu_servCaja.add(jMenuItem_servBuzon);
```

```
jMenuBar1.add(jMenu_servCaja);
```

```
jMenu2.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu2.setText("Consultas");
```

```
jMenuItem_consCalificaciones.setText("Calificaciones");
```

```
jMenu2.add(jMenuItem_consCalificaciones);
```

```
jMenuItem_consHorario.setText("Horario");
```

```
jMenu2.add(jMenuItem_consHorario);
```

```
jMenuItem_consEdoCuenta.setText("Estado de cuenta");
```

```
jMenu2.add(jMenuItem_consEdoCuenta);
```

```
jMenuItem_ConsMapa.setText("Mapa Universidad");
```

```
jMenu2.add(jMenuItem_ConsMapa);
```

```
jMenuItem_calendario.setText("Calendario");
```

```
jMenu2.add(jMenuItem_calendario);

jMenuBar1.add(jMenu2);

jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));
jMenu_tramites.setText("Trámites");

jMenuItem_becas.setText("Becas");
jMenu_tramites.add(jMenuItem_becas);

jMenuItem_constancias.setText("Constancias");
jMenu_tramites.add(jMenuItem_constancias);

jMenuBar1.add(jMenu_tramites);

jMenu4.setForeground(new java.awt.Color(255, 255, 255));
jMenu4.setText("LobosConect");

jMenuItem_correo.setText("Correo");
jMenu4.add(jMenuItem_correo);

jMenuItem_navega.setText("Navegar");
jMenu4.add(jMenuItem_navega);

jMenuItem_soporte.setText("Soporte");
```

```
jMenu4.add(jMenuItem_soporte);
```

```
jMenuBar1.add(jMenu4);
```

```
jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));
```

```
jMenu_cerrarSesion.setText("Cerrar sesión");
```

```
jMenuBar1.add(jMenu_cerrarSesion);
```

```
setJMenuBar(jMenuBar1);
```

```
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
```

```
getContentPane().setLayout(layout);
```

```
layout.setHorizontalGroup(
```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(layout.createSequentialGroup())
```

```
.addContainerGap()
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addComponent(rd_pagoCreditoUT)
```

```
.addComponent(rd_pagoTarjeta)
```

```
.addComponent(jLabel1)
```

```
.addComponent(jLabel5)
```

```
.addComponent(jLabel2)
```

```
.addComponent(jLabel7))
```

```
.addContainerGap(281, Short.MAX_VALUE))
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```

        .addGroup(layout.createSequentialGroup())

        .addContainerGap()

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addComponent(jLabel6)

            .addComponent(btn_pagar, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addComponent(jLabel3)

            .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

            .addGroup(layout.createSequentialGroup()

                .addComponent(jLabel4)

                .addGap(18, 18, 18)

                .addComponent(cb_vencimientoMes,
javax.swing.GroupLayout.PREFERRED_SIZE, 43, javax.swing.GroupLayout.PREFERRED_SIZE))

                .addComponent(pass_cve))

                .addComponent(txt_numTarjeta))

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

            .addComponent(tBtn_tecladoTarjeta,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

            .addComponent(tBtn_tecladoCVE, javax.swing.GroupLayout.DEFAULT_SIZE,
83, Short.MAX_VALUE))

```

```

        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()

        .addGap(14, 14, 14)

        .addComponent(cb_vencimientoAnio,
javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE))))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 246,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap()))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

        .addContainerGap()

        .addComponent(jLabel1)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(jLabel5)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jLabel7)

        .addGap(16, 16, 16)

        .addComponent(jLabel2)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(rd_pagoCreditoUT)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(rd_pagoTarjeta)

        .addContainerGap(188, Short.MAX_VALUE))

```

```

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup())

                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING,
false)

                    .addComponent(jScrollPane1)

                    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup())

                        .addGap(155, 155, 155)

                        .addComponent(jLabel3)

                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(txt_numTarjeta, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addComponent(tBtn_tecladoTarjeta))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addComponent(jLabel6)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(pass_cve, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(tBtn_tecladoCVE))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```



```

        .addComponent(jLabel4)

        .addComponent(cb_vencimientoMes,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(cb_vencimientoAnio,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(btn_pagar)))

    .addContainerGap()))

);

pack();
} // </editor-fold>

private void teclado_btn3ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {

    /* Set the Nimbus look and feel */

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

        * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

```

```

    */

    try {

        for (javax.swing.UIManager.LookAndFeelInfo info :
            javax.swing.UIManager.getInstalledLookAndFeels()) {

            if ("Nimbus".equals(info.getName())) {

                javax.swing.UIManager.setLookAndFeel(info.getClassName());

                break;

            }

        }

    } catch (ClassNotFoundException ex) {

        java.util.logging.Logger.getLogger(TramConstancia.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

        java.util.logging.Logger.getLogger(TramConstancia.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (IllegalAccessException ex) {

        java.util.logging.Logger.getLogger(TramConstancia.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

        java.util.logging.Logger.getLogger(TramConstancia.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    }

}

//</editor-fold>

/* Create and display the form */

```

```

        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {

                new TramConstancia().setVisible(true);

            }

        });
    }

    // Variables declaration - do not modify

    public javax.swing.JButton btn_pagar;

    private javax.swing.ButtonGroup buttonGroup1;

    public javax.swing.JComboBox<String> cb_vencimientoAnio;

    public javax.swing.JComboBox<String> cb_vencimientoMes;

    public javax.swing.JFrame frame_servTeclado;

    private javax.swing.JLabel jLabel1;

    private javax.swing.JLabel jLabel2;

    private javax.swing.JLabel jLabel3;

    private javax.swing.JLabel jLabel4;

    private javax.swing.JLabel jLabel5;

    private javax.swing.JLabel jLabel6;

    private javax.swing.JLabel jLabel7;

    private javax.swing.JMenu jMenu2;

    private javax.swing.JMenu jMenu4;

    private javax.swing.JMenuBar jMenuBar1;

    public javax.swing.JMenuItem jMenuItem_ConsMapa;

    public javax.swing.JMenuItem jMenuItem_becas;

```

```
public javax.swing.JMenuItem jMenuItem_calendario;

public javax.swing.JMenuItem jMenuItem_consCalificaciones;

public javax.swing.JMenuItem jMenuItem_consEdoCuenta;

public javax.swing.JMenuItem jMenuItem_consHorario;

public javax.swing.JMenuItem jMenuItem_constancias;

public javax.swing.JMenuItem jMenuItem_correo;

public javax.swing.JMenuItem jMenuItem_navega;

public javax.swing.JMenuItem jMenuItem_servBuzon;

public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

public javax.swing.JMenuItem jMenuItem_soporte;

public javax.swing.JMenu jMenu_cerrarSesion;

private javax.swing.JMenu jMenu_servCaja;

public javax.swing.JMenu jMenu_tramites;

private javax.swing.JScrollPane jScrollPane1;

public javax.swing.JPasswordField pass_cve;

public javax.swing.JRadioButton rd_pagoCreditoUT;

public javax.swing.JRadioButton rd_pagoTarjeta;

public javax.swing.JToggleButton tBtn_tecladoCVE;

public javax.swing.JToggleButton tBtn_tecladoTarjeta;

public javax.swing.JButton teclado_btn0;

public javax.swing.JButton teclado_btn1;

public javax.swing.JButton teclado_btn2;
```

```

public javax.swing.JButton teclado_btn3;

public javax.swing.JButton teclado_btn4;

public javax.swing.JButton teclado_btn5;

public javax.swing.JButton teclado_btn6;

public javax.swing.JButton teclado_btn7;

public javax.swing.JButton teclado_btn8;

public javax.swing.JButton teclado_btn9;

public javax.swing.JButton teclado_btnAceptar;

public javax.swing.JButton teclado_btnAtras;

public javax.swing.JButton teclado_btnLimpiar;

public javax.swing.JTextArea txtArea_ticket;

public javax.swing.JTextField txt_numTarjeta;

// End of variables declaration
}

```

principalMenu

```

package Vista;

public class principalMenu extends javax.swing.JFrame {

    /**
     * Creates new form principalMenu
     */
    public principalMenu() {

        initComponents();
    }
}

```

```
}
```

```
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
```

```
    jLabel1 = new javax.swing.JLabel();
```

```
    lbl_usrSaudo = new javax.swing.JLabel();
```

```
    jMenuBar1 = new javax.swing.JMenuBar();
```

```
    jMenu_servCaja = new javax.swing.JMenu();
```

```
    jMenuItem_servCaja = new javax.swing.JMenuItem();
```

```
    jMenuItem_servCreditoUT = new javax.swing.JMenuItem();
```

```
    jMenuItem_servLoboletos = new javax.swing.JMenuItem();
```

```
    jMenuItem_servEvaluacion = new javax.swing.JMenuItem();
```

```
    jMenuItem_servBuzon = new javax.swing.JMenuItem();
```

```
    jMenu_Consultas = new javax.swing.JMenu();
```

```
    jMenuItem_consCalificaciones = new javax.swing.JMenuItem();
```

```
    jMenuItem_consHorario = new javax.swing.JMenuItem();
```

```
    jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();
```

```
    jMenuItem_ConsMapa = new javax.swing.JMenuItem();
```

```
    jMenuItem_calendario = new javax.swing.JMenuItem();
```

```
    jMenu_tramites = new javax.swing.JMenu();
```

```
    jMenuItem_becas = new javax.swing.JMenuItem();
```

```
    jMenuItem_constancias = new javax.swing.JMenuItem();
```

```
    jMenu4 = new javax.swing.JMenu();
```

```
jMenuItem_correo = new javax.swing.JMenuItem();

jMenuItem_navega = new javax.swing.JMenuItem();

jMenuItem_soporte = new javax.swing.JMenuItem();

jMenu_cerrarSesion = new javax.swing.JMenu();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Imagenes/bg_avion.png"))); // NOI18N


lbl_usrSaudo.setText("Usr");


jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));


jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));
jMenu_servCaja.setText("Servicios");


jMenuItem_servCaja.setText("Caja");
jMenu_servCaja.add(jMenuItem_servCaja);


jMenuItem_servCreditoUT.setText("CreditoUT");
jMenu_servCaja.add(jMenuItem_servCreditoUT);


jMenuItem_servLoboletos.setText("Loboletos");
jMenu_servCaja.add(jMenuItem_servLoboletos);
```

```
jMenuItem_servEvaluacion.setText("Evaluación de servicios");
jMenu_servCaja.add(jMenuItem_servEvaluacion);

jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
jMenu_servCaja.add(jMenuItem_servBuzon);

jMenuBar1.add(jMenu_servCaja);

jMenu_Consultas.setForeground(new java.awt.Color(255, 255, 255));
jMenu_Consultas.setText("Consultas");

jMenuItem_consCalificaciones.setText("Calificaciones");
jMenu_Consultas.add(jMenuItem_consCalificaciones);

jMenuItem_consHorario.setText("Horario");
jMenu_Consultas.add(jMenuItem_consHorario);

jMenuItem_consEdoCuenta.setText("Estado de cuenta");
jMenu_Consultas.add(jMenuItem_consEdoCuenta);

jMenuItem_ConsMapa.setText("Mapa Universidad");
jMenu_Consultas.add(jMenuItem_ConsMapa);

jMenuItem_calendario.setText("Calendario");
```



```
jMenu_Conultas.add(jMenuItem_calendario);

jMenuBar1.add(jMenu_Conultas);

jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));
jMenu_tramites.setText("Trámites");

jMenuItem_becas.setText("Becas");
jMenu_tramites.add(jMenuItem_becas);

jMenuItem_constancias.setText("Constancias");
jMenu_tramites.add(jMenuItem_constancias);

jMenuBar1.add(jMenu_tramites);

jMenu4.setForeground(new java.awt.Color(255, 255, 255));
jMenu4.setText("LobosConect");

jMenuItem_correo.setText("Correo");
jMenu4.add(jMenuItem_correo);

jMenuItem_navega.setText("Navegar");
jMenu4.add(jMenuItem_navega);

jMenuItem_soporte.setText("Soporte");
```

```

jMenu4.add(jMenuItem_soporte);

jMenuBar1.add(jMenu4);

jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));
jMenu_cerrarSesion.setText("Cerrar sesión");
jMenuBar1.add(jMenu_cerrarSesion);

setJMenuBar(jMenuBar1);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);

layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
            .addGap(0, 0, Short.MAX_VALUE)
            .addComponent(lbl_usrSaudo)
            .addGap(0, 0, Short.MAX_VALUE))
);

layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 284,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 11,
Short.MAX_VALUE)

        .addComponent(lbl_usrSaudo)

        .addContainerGap()

    );

    pack();
} // </editor-fold>

/**
 * @param args the command line arguments
 */

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;
private javax.swing.JMenu jMenu4;
private javax.swing.JMenuBar jMenuBar1;
public javax.swing.JMenuItem jMenuItem_Con Mapa;
public javax.swing.JMenuItem jMenuItem_becas;
public javax.swing.JMenuItem jMenuItem_calendario;
public javax.swing.JMenuItem jMenuItem_consCalificaciones;
public javax.swing.JMenuItem jMenuItem_consEdoCuenta;
public javax.swing.JMenuItem jMenuItem_consHorario;
public javax.swing.JMenuItem jMenuItem_constancias;
public javax.swing.JMenuItem jMenuItem_correo;
public javax.swing.JMenuItem jMenuItem_navega;

```

```

public javax.swing.JMenuItem jMenuItem_servBuzon;

public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

public javax.swing.JMenuItem jMenuItem_servSoporte;

public javax.swing.JMenu jMenu_consultas;

public javax.swing.JMenu jMenu_cerrarSesion;

public javax.swing.JMenu jMenu_servCaja;

public javax.swing.JMenu jMenu_tramites;

public javax.swing.JLabel lbl_usrSaudo;

// End of variables declaration
}

```

ServCaja

```

package Vista;

import Controlador.*;

import Modelo.*;

public class servCaja extends javax.swing.JFrame {

    /**
     * Creates new form principalMenu
     */
    public servCaja() {

```

```

        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        frame_servTeclado = new javax.swing.JFrame();
        teclado_btn1 = new javax.swing.JButton();
        teclado_btn3 = new javax.swing.JButton();
        teclado_btn4 = new javax.swing.JButton();
        teclado_btn5 = new javax.swing.JButton();
        teclado_btn6 = new javax.swing.JButton();
        teclado_btn7 = new javax.swing.JButton();
        teclado_btn8 = new javax.swing.JButton();
        teclado_btn9 = new javax.swing.JButton();
        teclado_btnAceptar = new javax.swing.JButton();
        teclado_btn0 = new javax.swing.JButton();
        teclado_btnAtras = new javax.swing.JButton();
        teclado_btnLimpiar = new javax.swing.JButton();
        teclado_btn2 = new javax.swing.JButton();
        radioGroup_formaPago = new javax.swing.ButtonGroup();
        jLabel1 = new javax.swing.JLabel();
        cb_tipoPago = new javax.swing.JComboBox<>();
        jLabel2 = new javax.swing.JLabel();
    }

```

```
rd_pagoCreditoUT = new javax.swing.JRadioButton();

rd_pagoTarjeta = new javax.swing.JRadioButton();

jLabel3 = new javax.swing.JLabel();

txt_numTarjeta = new javax.swing.JTextField();

jLabel4 = new javax.swing.JLabel();

cb_vencimientoMes = new javax.swing.JComboBox<>();

jLabel5 = new javax.swing.JLabel();

cb_vencimientoAnio = new javax.swing.JComboBox<>();

jLabel6 = new javax.swing.JLabel();

pass_cve = new javax.swing.JPasswordField();

jScrollPane1 = new javax.swing.JScrollPane();

txtArea_recibo = new javax.swing.JTextArea();

btn_pagar = new javax.swing.JButton();

tBtn_tecladoTarjeta = new javax.swing.JToggleButton();

tBtn_tecladoCVE = new javax.swing.JToggleButton();

cb_tipoPago_loboleto = new javax.swing.JComboBox<>();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu_servCaja = new javax.swing.JMenu();

jMenuItem_servCaja = new javax.swing.JMenuItem();

jMenuItem_servCreditoUT = new javax.swing.JMenuItem();

jMenuItem_servLoboletos = new javax.swing.JMenuItem();

jMenuItem_servEvaluacion = new javax.swing.JMenuItem();

jMenuItem_servBuzon = new javax.swing.JMenuItem();

jMenu2 = new javax.swing.JMenu();

jMenuItem_consCalificaciones = new javax.swing.JMenuItem();
```

```

jMenuItem_consHorario = new javax.swing.JMenuItem();

jMenuItem_consEdoCuenta = new javax.swing.JMenuItem();

jMenuItem_ConsMapa = new javax.swing.JMenuItem();

jMenuItem_calendario = new javax.swing.JMenuItem();

jMenu_tramites = new javax.swing.JMenu();

jMenuItem_becas = new javax.swing.JMenuItem();

jMenuItem_constancias = new javax.swing.JMenuItem();

jMenu4 = new javax.swing.JMenu();

jMenuItem_correo = new javax.swing.JMenuItem();

jMenuItem_navega = new javax.swing.JMenuItem();

jMenuItem_soporte = new javax.swing.JMenuItem();

jMenu_cerrarSesion = new javax.swing.JMenu();


frame_servTeclado.setTitle("ServTeclado");

frame_servTeclado.setMinimumSize(new java.awt.Dimension(250, 396));


teclado_btn1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn1.setText("1");


teclado_btn3.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
teclado_btn3.setText("3");

teclado_btn3.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        teclado_btn3ActionPerformed(evt);

    }

})

```

```
});
```

```
teclado_btn4.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn4.setText("4");
```

```
teclado_btn5.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn5.setText("5");
```

```
teclado_btn6.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn6.setText("6");
```

```
teclado_btn7.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn7.setText("7");
```

```
teclado_btn8.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn8.setText("8");
```

```
teclado_btn9.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn9.setText("9");
```

```
teclado_btnAceptar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btnAceptar.setText("Aceptar");
```

```
teclado_btn0.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
teclado_btn0.setText("0");
```



```
teclado_btnAtras.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N  
teclado_btnAtras.setText("<");  
  
teclado_btnLimpiar.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N  
teclado_btnLimpiar.setText("C");  
  
teclado_btn2.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N  
teclado_btn2.setText("2");  
  
javax.swing.GroupLayout frame_servTecladoLayout = new  
javax.swing.GroupLayout(frame_servTeclado.getContentPane());  
  
frame_servTeclado.getContentPane().setLayout(frame_servTecladoLayout);  
  
frame_servTecladoLayout.setHorizontalGroup(  
frame_servTecladoLayout.createParallelGroup(GroupLayout.Alignment.LEADING)  
.addGroup(frame_servTecladoLayout.createSequentialGroup()  
.addGroup(frame_servTecladoLayout.createSequentialGroup()  
.addComponent(teclado_btnAtras, GroupLayout.PREFERRED_SIZE, 66,  
GroupLayout.PREFERRED_SIZE)  
.addPreferredGap(LayoutStyle.ComponentPlacement.RELATED)  
.addComponent(teclado_btnLimpiar, GroupLayout.PREFERRED_SIZE, 66,  
GroupLayout.PREFERRED_SIZE)  
.addPreferredGap(LayoutStyle.ComponentPlacement.RELATED)  
.addComponent(teclado_btn2, GroupLayout.PREFERRED_SIZE, 66,  
GroupLayout.PREFERRED_SIZE))
```

```

        .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

            .addComponent(teclado_btn7, javax.swing.GroupLayout.DEFAULT_SIZE, 66,
Short.MAX_VALUE)

            .addComponent(teclado_btn0, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGroup(frame_servTecladoLayout.createSequentialGroup())

            .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE,
66, javax.swing.GroupLayout.PREFERRED_SIZE))))

        .addGroup(frame_servTecladoLayout.createSequentialGroup())

        .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addContainerGap(14, Short.MAX_VALUE))

);

frame_servTecladoLayout.setVerticalGroup(

frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(frame_servTecladoLayout.createSequentialGroup()

                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(teclado_btn1, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn3, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(teclado_btn4, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(teclado_btn5, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(teclado_btn7, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn8, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btn9, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TR
AILING)

        .addComponent(teclado_btn0, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(frame_servTecladoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(teclado_btnAtras, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(teclado_btnLimpiar, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(teclado_btnAceptar, javax.swing.GroupLayout.PREFERRED_SIZE, 46,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap())

```

```
);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jLabel1.setText("Tipo de pago:");

    cb_tipoPago.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {
"Colegiatura", "Examen", "Inscripción", "Titulación", "Loboletos" }));

jLabel2.setText("Forma de pago:");

radioGroup_formaPago.add(rd_pagoCreditoUT);

rd_pagoCreditoUT.setSelected(true);

rd_pagoCreditoUT.setText("CreditoUT");

radioGroup_formaPago.add(rd_pagoTarjeta);

rd_pagoTarjeta.setText("Tarjeta");

jLabel3.setText("Número de tarjeta:");

jLabel4.setText("Vencimiento:");

    cb_vencimientoMes.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "1",
"2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12" }));

jLabel5.setText("/");
```

```
cb_vencimientoAnio.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {  
"19", "20", "21", "22", "23", "24", "25", "26", "27", "28" }));  
  
jLabel6.setText("CVE:");  
  
txtArea_recibo.setColumns(20);  
txtArea_recibo.setRows(5);  
jScrollPane1.setViewportView(txtArea_recibo);  
  
btn_pagar.setText("Pagar");  
  
tBtn_tecladoTarjeta.setText("Teclado");  
  
tBtn_tecladoCVE.setText("Teclado");  
  
cb_tipoPago_loboleto.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] {  
"Seleccionar", "Lobobus", "Impresiones", "Copias" }));  
  
jMenuBar1.setBackground(new java.awt.Color(6, 12, 60));  
jMenuBar1.setForeground(new java.awt.Color(255, 255, 255));  
  
jMenu_servCaja.setForeground(new java.awt.Color(255, 255, 255));  
jMenu_servCaja.setText("Servicios");  
  
jMenuItem_servCaja.setText("Caja");
```

```
jMenu_servCaja.add(jMenuItem_servCaja);

jMenuItem_servCreditoUT.setText("CreditoUT");
jMenu_servCaja.add(jMenuItem_servCreditoUT);

jMenuItem_servLoboletos.setText("Loboletos");
jMenu_servCaja.add(jMenuItem_servLoboletos);

jMenuItem_servEvaluacion.setText("Evaluación de servicios");
jMenu_servCaja.add(jMenuItem_servEvaluacion);

jMenuItem_servBuzon.setText("Buzón de quejas y sugerencias");
jMenu_servCaja.add(jMenuItem_servBuzon);

jMenuBar1.add(jMenu_servCaja);

jMenu2.setForeground(new java.awt.Color(255, 255, 255));
jMenu2.setText("Consultas");

jMenuItem_consCalificaciones.setText("Calificaciones");
jMenu2.add(jMenuItem_consCalificaciones);

jMenuItem_consHorario.setText("Horario");
jMenu2.add(jMenuItem_consHorario);
```

```
jMenuItem_consEdoCuenta.setText("Estado de cuenta");  
jMenu2.add(jMenuItem_consEdoCuenta);  
  
jMenuItem_Con Mapa.setText("Mapa Universidad");  
jMenu2.add(jMenuItem_Con Mapa);  
  
jMenuItem_calendario.setText("Calendario");  
jMenu2.add(jMenuItem_calendario);  
  
jMenuBar1.add(jMenu2);  
  
jMenu_tramites.setForeground(new java.awt.Color(255, 255, 255));  
jMenu_tramites.setText("Trámites");  
  
jMenuItem_becas.setText("Becas");  
jMenu_tramites.add(jMenuItem_becas);  
  
jMenuItem_constancias.setText("Constancias");  
jMenu_tramites.add(jMenuItem_constancias);  
  
jMenuBar1.add(jMenu_tramites);  
  
jMenu4.setForeground(new java.awt.Color(255, 255, 255));  
jMenu4.setText("LobosConect");
```



```

jMenuItem_correo.setText("Correo");

jMenu4.add(jMenuItem_correo);


jMenuItem_navega.setText("Navegar");

jMenu4.add(jMenuItem_navega);


jMenuItem_soporte.setText("Soporte");

jMenu4.add(jMenuItem_soporte);


jMenuBar1.add(jMenu4);


jMenu_cerrarSesion.setForeground(new java.awt.Color(255, 255, 255));

jMenu_cerrarSesion.setText("Cerrar sesión");

jMenuBar1.add(jMenu_cerrarSesion);


setJMenuBar(jMenuBar1);


javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()

            .addGap(10, 10, 10)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addComponent(rd_pagoTarjeta)

            )

        )

);

```

```

        .addComponent(rd_pagoCreditoUT)

        .addComponent(jLabel1)

        .addGroup(layout.createSequentialGroup())

            .addComponent(cb_tipoPago, javax.swing.GroupLayout.PREFERRED_SIZE, 122,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            .addComponent(cb_tipoPago_loboleto, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addComponent(jLabel6)

        .addComponent(btn_pagar, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        .addComponent(jLabel3)

        .addComponent(jLabel2)

        .addGroup(layout.createSequentialGroup())

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                    .addGroup(layout.createSequentialGroup())

                        .addComponent(jLabel4)

                        .addGap(18, 18, 18)

                        .addComponent(cb_vencimientoMes,
javax.swing.GroupLayout.PREFERRED_SIZE, 43, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addComponent(pass_cve))

                        .addComponent(txt_numTarjeta))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            )

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

    .addComponent(tBtn_tecladoTarjeta, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

    .addComponent(tBtn_tecladoCVE, javax.swing.GroupLayout.DEFAULT_SIZE, 83,
Short.MAX_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup())

    .addComponent(jLabel5)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addComponent(cb_vencimientoAnio,
javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE))))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 246,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addContainerGap())

);

layout.setVerticalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(layout.createSequentialGroup())

    .addContainerGap()

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING,
false)

        .addComponent(jScrollPane1)

        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup())

            .addComponent(jLabel1)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(cb_tipoPago, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(cb_tipoPago_loboleto,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addComponent(jLabel2)

    .addGap(8, 8, 8)

    .addComponent(rd_pagoCreditoUT)

    .addGap(3, 3, 3)

    .addComponent(rd_pagoTarjeta)

    .addGap(33, 33, 33)

    .addComponent(jLabel3)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(txt_numTarjeta, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(tBtn_tecladoTarjeta))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addComponent(jLabel6)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(pass_cve, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(tBtn_tecladoCVE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(jLabel4)

        .addComponent(cb_vencimientoMes, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(jLabel5)

        .addComponent(cb_vencimientoAnio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(btn_pagar)))

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

);

pack();
} // </editor-fold>

private void teclado_btn3ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {

```

```

/* Set the Nimbus look and feel */

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
   * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
   */

try {

    for (javax.swing.UIManager.LookAndFeelInfo info :
        javax.swing.UIManager.getInstalledLookAndFeels()) {

        if ("Nimbus".equals(info.getName())) {

            javax.swing.UIManager.setLookAndFeel(info.getClassName());

            break;

        }

    }

} catch (ClassNotFoundException ex) {

    java.util.logging.Logger.getLogger(servCaja.class.getName()).log(java.util.logging.Level.SEVERE,
        null, ex);

    } catch (InstantiationException ex) {

    java.util.logging.Logger.getLogger(servCaja.class.getName()).log(java.util.logging.Level.SEVERE,
        null, ex);

    } catch (IllegalAccessException ex) {

    java.util.logging.Logger.getLogger(servCaja.class.getName()).log(java.util.logging.Level.SEVERE,
        null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

    java.util.logging.Logger.getLogger(servCaja.class.getName()).log(java.util.logging.Level.SEVERE,
        null, ex);

```

```

    }

    //</editor-fold>

    //</editor-fold>

    /* Create and display the form */

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new servCaja().setVisible(true);

        }

    });
}

// Variables declaration - do not modify

public javax.swing.JButton btn_pagar;

public javax.swing.JComboBox<String> cb_tipoPago;

public javax.swing.JComboBox<String> cb_tipoPago_loboleto;

public javax.swing.JComboBox<String> cb_vencimientoAnio;

public javax.swing.JComboBox<String> cb_vencimientoMes;

public javax.swing.JFrame frame_servTeclado;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

```

```
private javax.swing.JMenu jMenuItem2;

private javax.swing.JMenu jMenuItem4;

private javax.swing.JMenuBar jMenuItemBar1;

public javax.swing.JMenuItem jMenuItem_ConsMapa;

public javax.swing.JMenuItem jMenuItem_becas;

public javax.swing.JMenuItem jMenuItem_calendario;

public javax.swing.JMenuItem jMenuItem_consCalificaciones;

public javax.swing.JMenuItem jMenuItem_consEdoCuenta;

public javax.swing.JMenuItem jMenuItem_consHorario;

public javax.swing.JMenuItem jMenuItem_constancias;

public javax.swing.JMenuItem jMenuItem_correo;

public javax.swing.JMenuItem jMenuItem_navega;

public javax.swing.JMenuItem jMenuItem_servBuzon;

public javax.swing.JMenuItem jMenuItem_servCaja;

public javax.swing.JMenuItem jMenuItem_servCreditoUT;

public javax.swing.JMenuItem jMenuItem_servEvaluacion;

public javax.swing.JMenuItem jMenuItem_servLoboletos;

public javax.swing.JMenuItem jMenuItem_soporte;

public javax.swing.JMenu jMenuItem_cerrarSesion;

private javax.swing.JMenu jMenuItem_servCaja;

public javax.swing.JMenu jMenuItem_tramites;

private javax.swing.JScrollPane jScrollPane1;

public javax.swing.JPasswordField pass_cve;

private javax.swing.ButtonGroup radioButton_formaPago;

public javax.swing.JRadioButton rd_pagoCreditoUT;
```



```

public javax.swing.JRadioButton rd_pagoTarjeta;

public javax.swing.JToggleButton tBtn_tecladoCVE;

public javax.swing.JToggleButton tBtn_tecladoTarjeta;

public javax.swing.JButton teclado_btn0;

public javax.swing.JButton teclado_btn1;

public javax.swing.JButton teclado_btn2;

public javax.swing.JButton teclado_btn3;

public javax.swing.JButton teclado_btn4;

public javax.swing.JButton teclado_btn5;

public javax.swing.JButton teclado_btn6;

public javax.swing.JButton teclado_btn7;

public javax.swing.JButton teclado_btn8;

public javax.swing.JButton teclado_btn9;

public javax.swing.JButton teclado_btnAceptar;

public javax.swing.JButton teclado_btnAtras;

public javax.swing.JButton teclado_btnLimpiar;

public javax.swing.JTextArea txtArea_recibo;

public javax.swing.JTextField txt_numTarjeta;

// End of variables declaration
}

```

Controladores

Controlador_buzon

```
package Controlador;
```

```
import Modelo.*;

import Vista.*;

import java.awt.Desktop;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

import java.awt.event.WindowEvent;

import java.awt.event.WindowListener;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.BorderFactory;

import javax.swing.JOptionPane;

import javax.swing.*;

import javax.swing.event.*;

import javax.swing.event.MenuEvent;

import javax.swing.event.MenuListener;


public class controlador_buzon implements ActionListener, WindowListener, MenuListener{

    public modelo_buzon miModelo;

    public ServBuzon miBuzon;

    public principalMenu miMenu;

    public mainInicial SalidaLogin;

    public int flagRol;
```

```

public int flagmatricula;

public String usrActivo;


public controlador_buzon(modelo_buzon miModelo, ServBuzon miBuzon, principalMenu
miMenu, mainInicial SalidaLogin) {

    this.miModelo = miModelo;

    this.miBuzon = miBuzon;

    this.miMenu = miMenu;

    this.SalidaLogin = SalidaLogin;

    this.miBuzon.btn_enviarBuzon.addActionListener(this);

    this.miBuzon.jMenuItem_servCaja.addActionListener(this);

    this.miBuzon.jMenuItem_servCreditoUT.addActionListener(this);

    this.miBuzon.jMenuItem_servLoboletos.addActionListener(this);

    this.miBuzon.jMenuItem_servEvaluacion.addActionListener(this);

    this.miBuzon.jMenuItem_consCalificaciones.addActionListener(this);

    this.miBuzon.jMenuItem_consHorario.addActionListener(this);

    this.miBuzon.jMenuItem_consEdoCuenta.addActionListener(this);

    this.miBuzon.jMenuItem_ConsMapa.addActionListener(this);

    this.miBuzon.jMenuItem_calendario.addActionListener(this);

    this.miBuzon.jMenuItem_becas.addActionListener(this);

    this.miBuzon.jMenuItem_constancias.addActionListener(this);

    this.miBuzon.jMenuItem_correo.addActionListener(this);

    this.miBuzon.jMenuItem_navega.addActionListener(this);

    this.miBuzon.jMenuItem_soporte.addActionListener(this);

    this.miBuzon.jMenu_cerrarSesion.addMenuListener(this);

    flagRol = 0;

```

```

    flagmatricula = 0;

    usrActivo = "";
}

@Override

//Método que recibe los eventos de acciones
public void actionPerformed(ActionEvent ae) {

    if (ae.getSource() == miBuzon.btn_enviarBuzon) {

        if(miBuzon.txtArea_txtBuzon.getText().equals("")) {

            JOptionPane.showMessageDialog(null, "No se detecto mensaje. Escriba el mensaje que desea enviar");

        }

        else {

            JOptionPane.showMessageDialog(null, "Su mensaje anonimo ha sido enviado correctamente");

            miBuzon.txtArea_txtBuzon.setText("");

        }

    }

}

//Al clic en la opción caja del menú
if (ae.getSource() == miBuzon.jMenuItem_servCaja) {

    //Instancia de una nueva caja
    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja coómo parametro paara el constructor en el controlador de caja
    controlador_servCaja miControladorCaja = new controlador_servCaja(miCaja,
    modeloCaja,miMenu, SalidaLogin);

```

```

        //Desde el controlador se hace visible la caja

        miControladorCaja.miCaja.setVisible(true);

        this.miBuzon.dispose();
    }

    //Al clic en la opción Loboletos del menú

    if (ae.getSource() == miBuzon.jMenuItem_servLoboletos) {

        //Instancia de una nueva caja

        servCaja miCaja = new servCaja();

        modelo_servCaja modeloCaja = new modelo_servCaja();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

        controlador_servCaja miControladorCaja = new controlador_servCaja(miCaja,
        modeloCaja,miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorCaja.flagRol = this.flagRol;

        miControladorCaja.flagmatricula = this.flagmatricula;

        miControladorCaja.usrActivo = this.usrActivo;

        miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

        miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

        //Desde el controlador se hace visible la caja

        miControladorCaja.miCaja.setVisible(true);

        this.miBuzon.dispose();
    }

    //Al clic en la opción CreditoUT del menú

    if (ae.getSource() == miBuzon.jMenuItem_servCreditoUT) {

        //Instancia de CreditoUT

        modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();
    }

```

```

ServCreditoUT vistaCreditoUt = new ServCreditoUT();

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

controlador_creditoUT          miControladorCreditoUT          =          new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUt, miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCreditoUT.flagRol = this.flagRol;

miControladorCreditoUT.flagmatricula = this.flagmatricula;

miControladorCreditoUT.usrActivo = this.usrActivo;

miControladorCreditoUT.cargaEstadoCreditoUT();

//Desde el controlador se hace visible el buzón

miControladorCreditoUT.miCreditoUT.setVisible(true);

this.miBuzon.dispose();
}

//Al clic en la opción Evaluación del menú

if (ae.getSource() == miBuzon.jMenuItem_servEvaluacion) {

    //Instancia de Evaluación

    modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

    ServEvaluacion vistaEvaluacion = new ServEvaluacion();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_evaluacion          miControladorEvaluacion          =          new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorEvaluacion.flagRol = this.flagRol;

    miControladorEvaluacion.flagmatricula = this.flagmatricula;

    miControladorEvaluacion.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón

```

```

        miControladorEvaluacion.miEvaluacion.setVisible(true);

        this.miBuzon.dispose();
    }

    //Si se da clic en el menu a la opción Calificaciones
    if (ae.getSource() == miBuzon.jMenuItem_consCalificaciones) {

        File archivo = new File(miModelo.abreCalif(flagmatricula));

        miModelo.abreArchivo(archivo);
    }

    //Si se da clic en el menu a la opción Horario
    if (ae.getSource() == miBuzon.jMenuItem_consHorario) {

        JOptionPane.showInputDialog("La url es: "+miModelo.abreHorario(flagmatricula));

        miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

        //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
        dic-2019/TICASI-7-E-9.pdf");
    }

    //Si se da clic en el menu a la opción Estado de cuenta
    if (ae.getSource() == miBuzon.jMenuItem_consEdoCuenta) {

        File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));

        miModelo.abreArchivo(archivo);
    }

    //Si se da clic en el menu a la opción de ver mapa
    if (ae.getSource() == miBuzon.jMenuItem_ConsMapa) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/mapaUT.jpg");

        miModelo.abreArchivo(archivo);
    }

```

```

    }

    //Si se da clic en el menu a la opción de ver calendario
    if (ae.getSource() == miBuzon.jMenuItem_calendario) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/calendario.pdf");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción becas
    if (ae.getSource() == miBuzon.jMenuItem_becas) {

        miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

    }

    //Al clic en la opción Constancias del menú
    if (ae.getSource() == miBuzon.jMenuItem_constancias) {

        //Instancia de buzón

        modelo_constancia modeloConstancia = new modelo_constancia();

        TramConstancia vistaConstancia = new TramConstancia();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

        controlador_constancia      miControladorConstancia      =      new
controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorConstancia.flagRol = this.flagRol;

        miControladorConstancia.flagmatricula = this.flagmatricula;

        miControladorConstancia.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón

        miControladorConstancia.miConstancia.setVisible(true);
    }

```



```

        this.miBuzon.dispose();
    }

    //Si se da clic en el menu a la opción correo
    if (ae.getSource() == miBuzon.jMenuItem_correo) {

        miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");
    }

    //Si se da clic en el menu a la opción navegar
    if (ae.getSource() == miBuzon.jMenuItem_navega) {

        miModelo.abreNavegador("http://www.google.com.mx");
    }

    //Al clic en la opción Soporte del menú
    if (ae.getSource() == miBuzon.jMenuItem_soporte) {

        //Instancia de Soporte

        modelo_soporte modeloSoporte = new modelo_soporte();

        ServSoporte vistaSoporte = new ServSoporte();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
        controlador_soporte miControladorSoporte = new controlador_soporte(modeloSoporte,
        vistaSoporte, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorSoporte.flagRol = this.flagRol;

        miControladorSoporte.flagmatricula = this.flagmatricula;

        miControladorSoporte.usrActivo = this.usrActivo;

        miControladorSoporte.mensajeUsuarioSoporte();

        if(miControladorSoporte.flagRol == 1) {

            miControladorSoporte.miSoporte.jMenuItem_servCaja.setVisible(false);

            miControladorSoporte.miSoporte.jMenuItem_servCreditoUT.setVisible(false);

```

```

        miControladorSoporte.miSoporte.jMenuItem_servLoboletos.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_servEvaluacion.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_consCalificaciones.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_consEdoCuenta.setVisible(false);

        miControladorSoporte.miSoporte.jMenu_tramites.setVisible(false);
    }

    else {

    }

    //Desde el controlador se hace visible el Soporte
    miControladorSoporte.miSoporte.setVisible(true);

    this.miBuzon.dispose();
}
}

@Override

public void windowOpened(WindowEvent we) {

}

@Override

public void windowClosing(WindowEvent we) {

}

@Override

```

```

public void windowClosed(WindowEvent we) {
}

@Override

public void windowIconified(WindowEvent we) {
}

@Override

public void windowDeiconified(WindowEvent we) {
}

@Override

public void windowActivated(WindowEvent we) {
}

@Override

public void windowDeactivated(WindowEvent we) {
}

@Override

//Si se preciona en el menú cerrar sesión

public void menuSelected(MenuEvent e) {

    int reply = JOptionPane. showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar
sesión", JOptionPane.YES_NO_CANCEL_OPTION);

    if(reply == JOptionPane.YES_OPTION) {

        //Carga el método que inicia el login desde la clase main y cierra la ventana acual

```

```

        SalidaLogin.cargaLogin();

        this.miBuzon.dispose();
    }

    else {

    }

}

@Override

public void menuDeselected(MenuEvent e) {

}

@Override

public void menuCanceled(MenuEvent e) {

}

}

```

Controlador_constancia

```

package Controlador;

import Modelo.*;

import Vista.*;

import java.awt.Desktop;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

import java.awt.event.WindowEvent;

```

```

import java.awt.event.WindowListener;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.BorderFactory;

import javax.swing.JOptionPane;

import javax.swing.*;

import javax.swing.event.*;

import javax.swing.event.MenuEvent;

import javax.swing.event.MenuListener;


/**
 *
 * @author Usuario
 */

public class controlador_constancia implements ActionListener, WindowListener, MenuListener {

    public modelo_constancia miModelo;

    public TramConstancia miConstancia;

    public principalMenu miMenu;

    public mainInicial SalidaLogin;

    public int flagRol;

    public int flagmatricula;

    public String usrActivo;

```

```

public controlador_constancia(modelo_constancia miModelo, TramConstancia
miConstancia, principalMenu miMenu, mainInicial SalidaLogin) {

    this.miModelo = miModelo;

    this.miConstancia = miConstancia;

    this.miMenu = miMenu;


    this.miConstancia.rd_pagoCreditoUT.addActionListener(this);

    this.miConstancia.rd_pagoTarjeta.addActionListener(this);

    this.miConstancia.txt_numTarjeta.addActionListener(this);

    this.miConstancia.txt_numTarjeta.setEnabled(false);

    this.miConstancia.tBtn_tecladoTarjeta.addActionListener(this);

    this.miConstancia.pass_cve.addActionListener(this);

    this.miConstancia.pass_cve.setEnabled(false);

    this.miConstancia.tBtn_tecladoCVE.addActionListener(this);

    this.miConstancia.cb_vencimientoMes.addActionListener(this);

    this.miConstancia.cb_vencimientoAnio.addActionListener(this);

    this.miConstancia.txtArea_ticket.setEnabled(false);

    this.miConstancia.btn_pagar.addActionListener(this);

    this.miConstancia.frame_servTeclado.addWindowListener(this);

    this.miConstancia.teclado_btn0.addActionListener(this);

    this.miConstancia.teclado_btn1.addActionListener(this);

    this.miConstancia.teclado_btn2.addActionListener(this);

    this.miConstancia.teclado_btn3.addActionListener(this);

    this.miConstancia.teclado_btn4.addActionListener(this);

    this.miConstancia.teclado_btn5.addActionListener(this);

    this.miConstancia.teclado_btn6.addActionListener(this);

```

```
this.miConstancia.teclado_btn7.addActionListener(this);

this.miConstancia.teclado_btn8.addActionListener(this);

this.miConstancia.teclado_btn9.addActionListener(this);

this.miConstancia.teclado_btnAtras.addActionListener(this);

this.miConstancia.teclado_btnLimpiar.addActionListener(this);

this.miConstancia.teclado_btnAceptar.addActionListener(this);


this.SalidaLogin = SalidaLogin;

this.miConstancia.jMenuItem_servCaja.addActionListener(this);

this.miConstancia.jMenuItem_servCreditoUT.addActionListener(this);

this.miConstancia.jMenuItem_servLoboletos.addActionListener(this);

this.miConstancia.jMenuItem_servEvaluacion.addActionListener(this);

this.miConstancia.jMenuItem_consCalificaciones.addActionListener(this);

this.miConstancia.jMenuItem_consHorario.addActionListener(this);

this.miConstancia.jMenuItem_consEdoCuenta.addActionListener(this);

this.miConstancia.jMenuItem_ConsMapa.addActionListener(this);

this.miConstancia.jMenuItem_calendario.addActionListener(this);

this.miConstancia.jMenuItem_becas.addActionListener(this);

this.miConstancia.jMenuItem_constancias.addActionListener(this);

this.miConstancia.jMenuItem_correo.addActionListener(this);

this.miConstancia.jMenuItem_navega.addActionListener(this);

this.miConstancia.jMenuItem_soporte.addActionListener(this);

this.miConstancia.jMenu_cerrarSesion.addMenuListener(this);

this.miConstancia.jMenuItem_servBuzon.addActionListener(this);

flagRol = 0;
```

```

    flagmatricula = 0;

    usrActivo = "";

    radioPagoValida();
}

@Override

//Método que recibe los eventos de acciones
public void actionPerformed(ActionEvent ae) {

    //Si cambia la selección del radio button group, mediante un método

    if (ae.getSource() == miConstancia.rd_pagoCreditoUT || ae.getSource() ==
miConstancia.rd_pagoTarjeta) {

        radioPagoValida();

    }

    //Comparamos el estado de los toggle buttons

    //Si se cliquea el toggle button de tarjeta

    if (ae.getSource() == miConstancia.tBtn_tecladoTarjeta) {

        //controla estados de los botones y muestra el teclado

        if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {

            miConstancia.frame_servTeclado.setVisible(true);

            miConstancia.tBtn_tecladoCVE.setSelected(false);

        } else {

            miConstancia.frame_servTeclado.setVisible(false);

        }

    }

    //En caso de que se clique el toggle de CVE

    else if (ae.getSource() == miConstancia.tBtn_tecladoCVE) {

```



```

if (miConstancia.tBtn_tecladoCVE.isSelected()) {

    miConstancia.tBtn_tecladoTarjeta.setSelected(false);

    miConstancia.frame_servTeclado.setVisible(true);

    miConstancia.teclado_btnAtras.setVisible(false);

} else {

    miConstancia.frame_servTeclado.setVisible(false);

}

}

//Eventos del clic de botones del teclado controlado por toggle buttons

if (ae.getSource() == miConstancia.teclado_btn0) {

    String nTarjeta = miConstancia.txt_numTarjeta.getText();

    String nCVE = miConstancia.pass_cve.getText();

    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {

        miConstancia.txt_numTarjeta.setText(nTarjeta+"0");

    }

    else {

        miConstancia.pass_cve.setText(nCVE + "0");

    }

}

if (ae.getSource() == miConstancia.teclado_btn1) {

    String nTarjeta = miConstancia.txt_numTarjeta.getText();

    String nCVE = miConstancia.pass_cve.getText();

    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {

```

```

        miConstancia.txt_numTarjeta.setText(nTarjeta+"1");
    }
    else {
        miConstancia.pass_cve.setText(nCVE + "1");
    }
}

if (ae.getSource() == miConstancia.teclado_btn2) {
    String nTarjeta = miConstancia.txt_numTarjeta.getText();
    String nCVE = miConstancia.pass_cve.getText();

    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {
        miConstancia.txt_numTarjeta.setText(nTarjeta+"2");
    }
    else {
        miConstancia.pass_cve.setText(nCVE + "2");
    }
}

if (ae.getSource() == miConstancia.teclado_btn3) {
    String nTarjeta = miConstancia.txt_numTarjeta.getText();
    String nCVE = miConstancia.pass_cve.getText();

    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {
        miConstancia.txt_numTarjeta.setText(nTarjeta+"3");
    }
    else {

```

```

        miConstancia.pass_cve.setText(nCVE + "3");
    }
}

if (ae.getSource() == miConstancia.teclado_btn4) {

    String nTarjeta = miConstancia.txt_numTarjeta.getText();

    String nCVE = miConstancia.pass_cve.getText();

    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {

        miConstancia.txt_numTarjeta.setText(nTarjeta+"4");

    }

    else {

        miConstancia.pass_cve.setText(nCVE + "4");

    }

}

if (ae.getSource() == miConstancia.teclado_btn5) {

    String nTarjeta = miConstancia.txt_numTarjeta.getText();

    String nCVE = miConstancia.pass_cve.getText();

    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {

        miConstancia.txt_numTarjeta.setText(nTarjeta+"5");

    }

    else {

        miConstancia.pass_cve.setText(nCVE + "5");

    }

}

```

```
if (ae.getSource() == miConstancia.teclado_btn6) {  
  
    String nTarjeta = miConstancia.txt_numTarjeta.getText();  
  
    String nCVE = miConstancia.pass_cve.getText();  
  
  
    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {  
  
        miConstancia.txt_numTarjeta.setText(nTarjeta+"6");  
  
    }  
  
    else {  
  
        miConstancia.pass_cve.setText(nCVE + "6");  
  
    }  
}  
  
if (ae.getSource() == miConstancia.teclado_btn7) {  
  
    String nTarjeta = miConstancia.txt_numTarjeta.getText();  
  
    String nCVE = miConstancia.pass_cve.getText();  
  
  
    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {  
  
        miConstancia.txt_numTarjeta.setText(nTarjeta+"7");  
  
    }  
  
    else {  
  
        miConstancia.pass_cve.setText(nCVE + "7");  
  
    }  
}  
  
if (ae.getSource() == miConstancia.teclado_btn8) {  
  
    String nTarjeta = miConstancia.txt_numTarjeta.getText();  
  
    String nCVE = miConstancia.pass_cve.getText();
```

```

        if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {
            miConstancia.txt_numTarjeta.setText(nTarjeta+"8");
        }
        else {
            miConstancia.pass_cve.setText(nCVE + "8");
        }
    }

    if (ae.getSource() == miConstancia.teclado_btn9) {
        String nTarjeta = miConstancia.txt_numTarjeta.getText();
        String nCVE = miConstancia.pass_cve.getText();

        if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {
            miConstancia.txt_numTarjeta.setText(nTarjeta+"9");
        }
        else {
            miConstancia.pass_cve.setText(nCVE + "9");
        }
    }

    //Acciones de limpiar campos con el botón C del teclado
    if (ae.getSource() == miConstancia.teclado_btnLimpiar) {
        if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {
            miConstancia.txt_numTarjeta.setText("");
        }
        else {

```

```

        miConstancia.pass_cve.setText("");
    }
}

//Borrar último caracter de las cajas de texto y password
if (ae.getSource() == miConstancia.teclado_btnAtras) {
    String nTarjeta = miConstancia.txt_numTarjeta.getText();
    String nCVE = miConstancia.pass_cve.getText();
    if (miConstancia.tBtn_tecladoTarjeta.isSelected()) {
        if(nTarjeta.length() != 0) {
            nTarjeta = nTarjeta.substring(0, nTarjeta.length()-1);
            miConstancia.txt_numTarjeta.setText(nTarjeta);
        }
        else {}
    }
    else {}
}

//Cerrar el teclado
if (ae.getSource() == miConstancia.teclado_btnAceptar) {
    miConstancia.frame_servTeclado.dispose();
    miConstancia.tBtn_tecladoCVE.setSelected(false);
    miConstancia.tBtn_tecladoTarjeta.setSelected(false);
}

//Al dar clic en pagar
if (ae.getSource() == miConstancia.btn_pagar) {

```

```

if((miConstancia.rd_pagoTarjeta.isSelected())&&(miConstancia.txt_numTarjeta.getText().equals("")
) || miConstancia.pass_cve.getText().equals("")) {

    JOptionPane.showMessageDialog(null, "ERROR: Favor de completar sus datos de pago");

}

else {

    JOptionPane.showMessageDialog(null, "Su pagao fue aceptado. Espere a que se genere su
comprobante, de igual forma este se enviará a su correo institucional");

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
proyecto

    File archivo = new File("src/Documentos/CAJERO-UT-COMPROBANTE DE PAGO.pdf");

    miModelo.abreArchivo(archivo);

}

}

//Al clic en la opción caja del menú

if (ae.getSource() == miConstancia.jMenuItem_servCaja) {

    //Instancia de una nueva caja

    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

    controlador_servCaja miControladorCaja = new controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

    //Desde el controlador se hace visible la caja

    miControladorCaja.miCaja.setVisible(true);

    this.miConstancia.dispose();

}

```

```

//Al clic en la opción CreditoUT del menú

if (ae.getSource() == miConstancia.jMenuItem_servCreditoUT) {

    //Instancia de CreditoUT

    modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();

    ServCreditoUT vistaCreditoUt = new ServCreditoUT();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_creditoUT      miControladorCreditoUT      =      new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUt, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorCreditoUT.flagRol = this.flagRol;

    miControladorCreditoUT.flagmatricula = this.flagmatricula;

    miControladorCreditoUT.usrActivo = this.usrActivo;

    miControladorCreditoUT.cargaEstadoCreditoUT();

    //Desde el controlador se hace visible el buzón

    miControladorCreditoUT.miCreditoUT.setVisible(true);

    this.miConstancia.dispose();

}

//Al clic en la opción Loboletos del menú

if (ae.getSource() == miConstancia.jMenuItem_servLoboletos) {

    //Instancia de una nueva caja

    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

    controlador_servCaja      miControladorCaja      =      new      controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

    //Paso los datos de usuario

```



```

miControladorCaja.flagRol = this.flagRol;

miControladorCaja.flagmatricula = this.flagmatricula;

miControladorCaja.usrActivo = this.usrActivo;

miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

//Desde el controlador se hace visible la caja

miControladorCaja.miCaja.setVisible(true);

this.miConstancia.dispose();

}

//Al clic en la opción Evaluación del menú

if (ae.getSource() == miConstancia.jMenuItem_servEvaluacion) {

    //Instancia de buzón

    modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

    ServEvaluacion vistaEvaluacion = new ServEvaluacion();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_evaluacion      miControladorEvaluacion      =      new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorEvaluacion.flagRol = this.flagRol;

    miControladorEvaluacion.flagmatricula = this.flagmatricula;

    miControladorEvaluacion.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón

    miControladorEvaluacion.miEvaluacion.setVisible(true);

    this.miConstancia.dispose();

}

//Al clic en la opción Buzón del menú

```

```

if (ae.getSource() == miConstancia.jMenuItem_servBuzon) {

    //Instancia de buzón

    modelo_buzon modeloBuzon = new modelo_buzon();

    ServBuzon vistaBuzon = new ServBuzon();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_buzon miControladorBuzon = new controlador_buzon(modeloBuzon,
vistaBuzon, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorBuzon.flagRol = this.flagRol;

    miControladorBuzon.flagmatricula = this.flagmatricula;

    miControladorBuzon.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón

    miControladorBuzon.miBuzon.setVisible(true);

    this.miConstancia.dispose();

}

//Si se da clic en el menu a la opción Calificaciones

if (ae.getSource() == miConstancia.jMenuItem_consCalificaciones) {

    File archivo = new File(miModelo.abreCalif(flagmatricula));

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción Horario

if (ae.getSource() == miConstancia.jMenuItem_consHorario) {

    miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

    //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
dic-2019/TICASI-7-E-9.pdf");

}

```

```

//Si se da clic en el menu a la opción Estado de cuenta

if (ae.getSource() == miConstancia.jMenuItem_consEdoCuenta) {

    File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción de ver mapa

if (ae.getSource() == miConstancia.jMenuItem_ConsMapa) {

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
    proyecto

    File archivo = new File("src/Documentos/mapaUT.jpg");

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción de ver calendario

if (ae.getSource() == miConstancia.jMenuItem_calendario) {

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
    proyecto

    File archivo = new File("src/Documentos/calendario.pdf");

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción becas

if (ae.getSource() == miConstancia.jMenuItem_becas) {

    miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

}

//Al clic en la opción Constancias del menú

if (ae.getSource() == miConstancia.jMenuItem_constancias) {

    //Instancia de buzón

```

```

    modelo_constancia modeloConstancia = new modelo_constancia();

    TramConstancia vistaConstancia = new TramConstancia();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_constancia          miControladorConstancia          =          new
controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorConstancia.flagRol = this.flagRol;

    miControladorConstancia.flagmatricula = this.flagmatricula;

    miControladorConstancia.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón

    miControladorConstancia.miConstancia.setVisible(true);

    this.miConstancia.dispose();
}

//Si se da clic en el menú a la opción correo
if (ae.getSource() == miConstancia.jMenuItem_correo) {

    miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");

}

//Si se da clic en el menú a la opción navegar
if (ae.getSource() == miConstancia.jMenuItem_navega) {

    miModelo.abreNavegador("http://www.google.com.mx");

}

//Al clic en la opción Soporte del menú
if (ae.getSource() == miConstancia.jMenuItem_soporte) {

    //Instancia de buzón

    modelo_soporte modeloSoporte = new modelo_soporte();

    ServSoporte vistaSoporte = new ServSoporte();

```

```

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
        controlador_soporte miControladorSoporte = new controlador_soporte(modeloSoporte,
vistaSoporte, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorSoporte.flagRol = this.flagRol;

        miControladorSoporte.flagmatricula = this.flagmatricula;

        miControladorSoporte.usrActivo = this.usrActivo;

        miControladorSoporte.mensajeUsuarioSoporte();

        //Desde el controlador se hace visible el buzón

        miControladorSoporte.miSoporte.setVisible(true);

        this.miConstancia.dispose();
    }
}

```

@Override

```

public void windowOpened(WindowEvent we) {

}

```

@Override

```

public void windowClosing(WindowEvent we) {

    miConstancia.tBtn_tecladoTarjeta.setSelected(false);

    this.miConstancia.tBtn_tecladoCVE.setSelected(false);

}

```

@Override

```

public void windowClosed(WindowEvent we) {

```

```
}
```

```
@Override
```

```
public void windowIconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowDeiconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowActivated(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowDeactivated(WindowEvent we) {  
}
```

```
@Override
```

```
//Si se preciona en el menú cerrar sesión  
public void menuSelected(MenuEvent e) {  
    int reply = JOptionPane.showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar  
sesión", JOptionPane.YES_NO_CANCEL_OPTION);  
    if(reply == JOptionPane.YES_OPTION) {  
        //Carga el método que incia el login desde la clase main y cierra la ventana acual  
        SalidaLogin.cargaLogin();  
    }  
}
```

```

        this.miConstancia.dispose();
    }
    else {

    }
}

@Override

public void menuDeselected(MenuEvent e) {

}

@Override

public void menuCanceled(MenuEvent e) {

}

public void radioPagoValida(){
    if(miConstancia.rd_pagoCreditoUT.isSelected()) {
        miConstancia.cb_vencimientoMes.setEnabled(false);
        miConstancia.cb_vencimientoAnio.setEnabled(false);
        miConstancia.tBtn_tecladoTarjeta.setEnabled(false);
        miConstancia.tBtn_tecladoCVE.setEnabled(false);
    }
    else if (miConstancia.rd_pagoTarjeta.isSelected()) {
        miConstancia.cb_vencimientoMes.setEnabled(true);
        miConstancia.cb_vencimientoAnio.setEnabled(true);
    }
}

```

```
        miConstancia.tBtn_tecladoTarjeta.setEnabled(true);

        miConstancia.tBtn_tecladoCVE.setEnabled(true);
    }
}
}
```

Controlador_creditoUT

```
package Controlador;

import Modelo.*;

import Vista.*;

import java.awt.Desktop;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

import java.awt.event.WindowEvent;

import java.awt.event.WindowListener;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.BorderFactory;

import javax.swing.JOptionPane;

import javax.swing.*;

import javax.swing.event.*;

import javax.swing.event.MenuEvent;
```



```

import javax.swing.event.MenuListener;

public class controlador_creditoUT implements ActionListener, WindowListener, MenuListener {

    public modelo_creditoUT miModelo;

    public ServCreditoUT miCreditoUT;

    public principalMenu miMenu;

    public mainInicial SalidaLogin;

    public int flagRol;

    public int flagmatricula;

    public String usrActivo;

    public controlador_creditoUT(modelo_creditoUT miModelo, ServCreditoUT miCreditoUT,
principalMenu miMenu, mainInicial SalidaLogin) {

        this.miModelo = miModelo;

        this.miCreditoUT = miCreditoUT;

        this.miMenu = miMenu;

        this.miCreditoUT.txt_numTarjeta.addActionListener(this);

        this.miCreditoUT.txt_numTarjeta.setEnabled(false);

        this.miCreditoUT.tBtn_tecladoTarjeta.addActionListener(this);

        this.miCreditoUT.pass_cve.addActionListener(this);

        this.miCreditoUT.pass_cve.setEnabled(false);

        this.miCreditoUT.tBtn_tecladoCVE.addActionListener(this);

        this.miCreditoUT.cb_vencimientoMes.addActionListener(this);

        this.miCreditoUT.cb_vencimientoAnio.addActionListener(this);

        this.miCreditoUT.txtArea_ticket.setEnabled(false);

```

```
this.miCreditoUT.btn_pagar.addActionListener(this);

this.miCreditoUT.frame_servTeclado.addWindowListener(this);

this.miCreditoUT.teclado_btn0.addActionListener(this);
this.miCreditoUT.teclado_btn1.addActionListener(this);
this.miCreditoUT.teclado_btn2.addActionListener(this);
this.miCreditoUT.teclado_btn3.addActionListener(this);
this.miCreditoUT.teclado_btn4.addActionListener(this);
this.miCreditoUT.teclado_btn5.addActionListener(this);
this.miCreditoUT.teclado_btn6.addActionListener(this);
this.miCreditoUT.teclado_btn7.addActionListener(this);
this.miCreditoUT.teclado_btn8.addActionListener(this);
this.miCreditoUT.teclado_btn9.addActionListener(this);
this.miCreditoUT.teclado_btnAtras.addActionListener(this);
this.miCreditoUT.teclado_btnLimpiar.addActionListener(this);
this.miCreditoUT.teclado_btnAceptar.addActionListener(this);


this.SalidaLogin = SalidaLogin;

this.miCreditoUT.jMenuItem_servCaja.addActionListener(this);
this.miCreditoUT.jMenuItem_servCreditoUT.addActionListener(this);
this.miCreditoUT.jMenuItem_servLoboletos.addActionListener(this);
this.miCreditoUT.jMenuItem_servEvaluacion.addActionListener(this);
this.miCreditoUT.jMenuItem_consCalificaciones.addActionListener(this);
this.miCreditoUT.jMenuItem_consHorario.addActionListener(this);
this.miCreditoUT.jMenuItem_consEdoCuenta.addActionListener(this);
this.miCreditoUT.jMenuItem_ConsMapa.addActionListener(this);
```

```

this.miCreditoUT.jMenuItem_calendario.addActionListener(this);
this.miCreditoUT.jMenuItem_becas.addActionListener(this);
this.miCreditoUT.jMenuItem_constancias.addActionListener(this);
this.miCreditoUT.jMenuItem_correo.addActionListener(this);
this.miCreditoUT.jMenuItem_navega.addActionListener(this);
this.miCreditoUT.jMenuItem_soporte.addActionListener(this);
this.miCreditoUT.jMenu_cerrarSesion.addMenuListener(this);
this.miCreditoUT.jMenuItem_servBuzon.addActionListener(this);

flagRol = 0;

flagmatricula = 0;

usrActivo = "";
}

```

@Override

//Método que recibe los eventos de acciones

```

public void actionPerformed(ActionEvent ae) {

    //Comparamos el estado de los toggle buttons

    //Si se cliquea el toggle button de tarjeta

    if (ae.getSource() == miCreditoUT.tBtn_tecladoTarjeta) {

        //controla estados de los botones y muestra el teclado

        if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

            miCreditoUT.frame_servTeclado.setVisible(true);

            miCreditoUT.tBtn_tecladoCVE.setSelected(false);

        } else {

            miCreditoUT.frame_servTeclado.setVisible(false);

```

```

    }
}

//En caso de que se clique el toggle de CVE
else if (ae.getSource() == miCreditoUT.tBtn_tecladoCVE) {
    if (miCreditoUT.tBtn_tecladoCVE.isSelected()) {
        miCreditoUT.tBtn_tecladoTarjeta.setSelected(false);
        miCreditoUT.frame_servTeclado.setVisible(true);
        miCreditoUT.teclado_btnAtras.setVisible(false);
    } else {
        miCreditoUT.frame_servTeclado.setVisible(false);
    }
}

//Eventos del clic de botones del teclado controlado por toggle buttons
if (ae.getSource() == miCreditoUT.teclado_btn0) {
    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();
    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {
        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"0");
    }
    else {
        miCreditoUT.pass_cve.setText(nCVE + "0");
    }
}

if (ae.getSource() == miCreditoUT.teclado_btn1) {

```

```

String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

String nCVE = miCreditoUT.pass_cve.getText();


if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

    miCreditoUT.txt_numTarjeta.setText(nTarjeta+"1");

}

else {

    miCreditoUT.pass_cve.setText(nCVE + "1");

}

}

if (ae.getSource() == miCreditoUT.teclado_btn2) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();


    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"2");

    }

    else {

        miCreditoUT.pass_cve.setText(nCVE + "2");

    }

}

if (ae.getSource() == miCreditoUT.teclado_btn3) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

```

```

if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

    miCreditoUT.txt_numTarjeta.setText(nTarjeta+"3");

}

else {

    miCreditoUT.pass_cve.setText(nCVE + "3");

}

}

if (ae.getSource() == miCreditoUT.teclado_btn4) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"4");

    }

    else {

        miCreditoUT.pass_cve.setText(nCVE + "4");

    }

}

if (ae.getSource() == miCreditoUT.teclado_btn5) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"5");

    }

}

```

```

else {

    miCreditoUT.pass_cve.setText(nCVE + "5");

}

}

if (ae.getSource() == miCreditoUT.teclado_btn6) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"6");

    }

    else {

        miCreditoUT.pass_cve.setText(nCVE + "6");

    }

}

if (ae.getSource() == miCreditoUT.teclado_btn7) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"7");

    }

    else {

        miCreditoUT.pass_cve.setText(nCVE + "7");

    }

}

```

```

}

if (ae.getSource() == miCreditoUT.teclado_btn8) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"8");

    }

    else {

        miCreditoUT.pass_cve.setText(nCVE + "8");

    }

}

if (ae.getSource() == miCreditoUT.teclado_btn9) {

    String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

    String nCVE = miCreditoUT.pass_cve.getText();

    if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

        miCreditoUT.txt_numTarjeta.setText(nTarjeta+"9");

    }

    else {

        miCreditoUT.pass_cve.setText(nCVE + "9");

    }

}

//Acciones de limpiar campos con el botón C del teclado

if (ae.getSource() == miCreditoUT.teclado_btnLimpiar) {

```



```

        if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

            miCreditoUT.txt_numTarjeta.setText("");

        }

        else {

            miCreditoUT.pass_cve.setText("");

        }

    }

    //Borrar último caracter de las cajas de texto y password

    if (ae.getSource() == miCreditoUT.teclado_btnAtras) {

        String nTarjeta = miCreditoUT.txt_numTarjeta.getText();

        String nCVE = miCreditoUT.pass_cve.getText();

        if (miCreditoUT.tBtn_tecladoTarjeta.isSelected()) {

            if(nTarjeta.length() != 0) {

                nTarjeta = nTarjeta.substring(0, nTarjeta.length()-1);

                miCreditoUT.txt_numTarjeta.setText(nTarjeta);

            }

            else {}

        }

        else {}

    }

    //Cerrar el teclado

    if (ae.getSource() == miCreditoUT.teclado_btnAceptar) {

        miCreditoUT.frame_servTeclado.dispose();

        miCreditoUT.tBtn_tecladoCVE.setSelected(false);

        miCreditoUT.tBtn_tecladoTarjeta.setSelected(false);
    }

```

```

    }

    //Al dar clic en pagar

    if (ae.getSource() == miCreditoUT.btn_pagar) {

        if(miCreditoUT.txt_numTarjeta.getText().equals("") ||
        miCreditoUT.pass_cve.getText().equals("")) {

            JOptionPane.showMessageDialog(null, "ERROR: Favor de completar sus datos de pago");

        }

        else {

            JOptionPane.showMessageDialog(null, "Su pago fue aceptado. Espere a que se genere su
            comprobante, de igual forma este se enviará a su correo institucional");

            miModelo.actualizaCreditoUT(flagmatricula,
            Integer.parseInt(miCreditoUT.lbl_estadoCreditoUT.getText()),
            Integer.parseInt(String.valueOf(miCreditoUT.cb_montoAbono.getSelectedItemAt(0))));

            //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
            proyecto

            File archivo = new File("src/Documentos/CAJERO-UT-COMPROBANTE DE PAGO.pdf");

            miModelo.abreArchivo(archivo);

            cargaEstadoCreditoUT();

        }

    }

    //Al clic en la opción caja del menú

    if (ae.getSource() == miCreditoUT.jMenuItem_servCaja) {

        //Instancia de una nueva caja

        servCaja miCaja = new servCaja();

        modelo_servCaja modeloCaja = new modelo_servCaja();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

```

```

        controlador_servCaja    miControladorCaja    =    new    controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

        //Desde el controlador se hace visible la caja

        miControladorCaja.miCaja.setVisible(true);

        this.miCreditoUT.dispose();
    }

    //Al clic en la opción CreditoUT del menú

    if (ae.getSource() == miCreditoUT.jMenuItem_servCreditoUT) {

        //Instancia de CreditoUT

        modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();

        ServCreditoUT vistaCreditoUt = new ServCreditoUT();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

        controlador_creditoUT    miControladorCreditoUT    =    new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUt, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorCreditoUT.flagRol = this.flagRol;

        miControladorCreditoUT.flagmatricula = this.flagmatricula;

        miControladorCreditoUT.usrActivo = this.usrActivo;

        miControladorCreditoUT.cargaEstadoCreditoUT();

        //Desde el controlador se hace visible el buzón

        miControladorCreditoUT.miCreditoUT.setVisible(true);

        this.miCreditoUT.dispose();
    }

    //Al clic en la opción Loboletos del menú

    if (ae.getSource() == miCreditoUT.jMenuItem_servLoboletos) {

        //Instancia de una nueva caja

```

```

servCaja miCaja = new servCaja();

modelo_servCaja modeloCaja = new modelo_servCaja();

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

controlador_servCaja    miControladorCaja    =    new    controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCaja.flagRol = this.flagRol;

miControladorCaja.flagmatricula = this.flagmatricula;

miControladorCaja.usrActivo = this.usrActivo;

miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

//Desde el controlador se hace visible la caja

miControladorCaja.miCaja.setVisible(true);

this.miCreditoUT.dispose();
}

//Al clic en la opción Evaluación del menú

if (ae.getSource() == miCreditoUT.jMenuItem_servEvaluacion) {

    //Instancia de buzón

    modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

    ServEvaluacion vistaEvaluacion = new ServEvaluacion();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_evaluacion    miControladorEvaluacion    =    new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorEvaluacion.flagRol = this.flagRol;

    miControladorEvaluacion.flagmatricula = this.flagmatricula;

```

```

miControladorEvaluacion.usrActivo = this.usrActivo;

//Desde el controlador se hace visible el buzón
miControladorEvaluacion.miEvaluacion.setVisible(true);
this.miCreditoUT.dispose();
}

//Al clic en la opción Buzón del menú
if (ae.getSource() == miCreditoUT.jMenuItem_servBuzon) {

    //Instancia de buzón
    modelo_buzon modeloBuzon = new modelo_buzon();

    ServBuzon vistaBuzon = new ServBuzon();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
    controlador_buzon miControladorBuzon = new controlador_buzon(modeloBuzon,
    vistaBuzon, miMenu, SalidaLogin);

    //Paso los datos de usuario
    miControladorBuzon.flagRol = this.flagRol;

    miControladorBuzon.flagmatricula = this.flagmatricula;

    miControladorBuzon.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón
    miControladorBuzon.miBuzon.setVisible(true);

    this.miCreditoUT.dispose();
}

//Si se da clic en el menu a la opción Calificaciones
if (ae.getSource() == miCreditoUT.jMenuItem_consCalificaciones) {

    File archivo = new File(miModelo.abreCalif(flagmatricula));

    miModelo.abreArchivo(archivo);
}

```

```

//Si se da clic en el menu a la opción Horario

if (ae.getSource() == miCreditoUT.jMenuItem_consHorario) {

    miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

    //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
dic-2019/TICASI-7-E-9.pdf");

}

//Si se da clic en el menu a la opción Estado de cuenta

if (ae.getSource() == miCreditoUT.jMenuItem_consEdoCuenta) {

    File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción de ver mapa

if (ae.getSource() == miCreditoUT.jMenuItem_ConsMapa) {

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
proyecto

    File archivo = new File("src/Documentos/mapaUT.jpg");

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción de ver calendario

if (ae.getSource() == miCreditoUT.jMenuItem_calendario) {

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
proyecto

    File archivo = new File("src/Documentos/calendario.pdf");

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción becas

```

```

if (ae.getSource() == miCreditoUT.jMenuItem_becas) {

    miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

}

//Al clic en la opción Constancias del menú

if (ae.getSource() == miCreditoUT.jMenuItem_constancias) {

    //Instancia de buzón

    modelo_constancia modeloConstancia = new modelo_constancia();

    TramConstancia vistaConstancia = new TramConstancia();

    //se pasa la nueva caja como parametro para el constructor en el controlador de buzón

    controlador_constancia          miControladorConstancia          =          new
controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorConstancia.flagRol = this.flagRol;

    miControladorConstancia.flagmatricula = this.flagmatricula;

    miControladorConstancia.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón

    miControladorConstancia.miConstancia.setVisible(true);

    this.miCreditoUT.dispose();

}

//Si se da clic en el menu a la opción correo

if (ae.getSource() == miCreditoUT.jMenuItem_correo) {

    miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");

}

//Si se da clic en el menu a la opción navegar

if (ae.getSource() == miCreditoUT.jMenuItem_navega) {

    miModelo.abreNavegador("http://www.google.com.mx");

```

```

    }

    //Al clic en la opción Soporte del menú
    if (ae.getSource() == miCreditoUT.jMenuItem_soporte) {

        //Instancia de buzón

        modelo_soporte modeloSoporte = new modelo_soporte();

        ServSoporte vistaSoporte = new ServSoporte();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

        controlador_soporte miControladorSoporte = new controlador_soporte(modeloSoporte,
        vistaSoporte, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorSoporte.flagRol = this.flagRol;

        miControladorSoporte.flagmatricula = this.flagmatricula;

        miControladorSoporte.usrActivo = this.usrActivo;

        miControladorSoporte.mensajeUsuarioSoporte();

        //Desde el controlador se hace visible el buzón

        miControladorSoporte.miSoporte.setVisible(true);

        this.miCreditoUT.dispose();

    }

}

@Override

public void windowOpened(WindowEvent we) {

}

@Override

public void windowClosing(WindowEvent we) {

```



```
miCreditoUT.tBtn_tecladoTarjeta.setSelected(false);  
this.miCreditoUT.tBtn_tecladoCVE.setSelected(false);  
}
```

```
@Override  
public void windowClosed(WindowEvent we) {  
}
```

```
@Override  
public void windowIconified(WindowEvent we) {  
}
```

```
@Override  
public void windowDeiconified(WindowEvent we) {  
}
```

```
@Override  
public void windowActivated(WindowEvent we) {  
}
```

```
@Override  
public void windowDeactivated(WindowEvent we) {  
}
```

```
@Override
```

```

//Si se preciona en el menú cerrar sesión

public void menuSelected(MenuEvent e) {

    int reply = JOptionPane.showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar
sesión", JOptionPane.YES_NO_CANCEL_OPTION);

    if(reply == JOptionPane.YES_OPTION) {

        //Carga el método que incia el login desde la clase main y cierra la ventana actual

        SalidaLogin.cargaLogin();

        this.miCreditoUT.dispose();

    }

    else {

        }

    }

@Override

public void menuDeselected(MenuEvent e) {

}

@Override

public void menuCanceled(MenuEvent e) {

}

}

public void cargaEstadoCreditoUT() {

}

miCreditoUT.lbl_estadoCreditoUT.setText(Integer.toString(miModelo.buscaCreditoUT(flagmatricul
a)));

```

```
}  
}
```

Controlador_evaluacion

```
package Controlador;  
  
import Modelo.*;  
  
import Vista.*;  
  
import java.awt.Desktop;  
  
import java.awt.event.ActionEvent;  
  
import java.awt.event.ActionListener;  
  
import java.io.File;  
  
import java.awt.event.WindowEvent;  
  
import java.awt.event.WindowListener;  
  
import java.io.File;  
  
import java.io.IOException;  
  
import java.util.logging.Level;  
  
import java.util.logging.Logger;  
  
import javax.swing.BorderFactory;  
  
import javax.swing.JOptionPane;  
  
import javax.swing.*;  
  
import javax.swing.event.*;  
  
import javax.swing.event.MenuEvent;  
  
import javax.swing.event.MenuListener;  
  
public class controlador_evaluacion implements ActionListener, WindowListener, MenuListener {
```

```

public modelo_evaluacion miModelo;

public ServEvaluacion miEvaluacion;

public principalMenu miMenu;

public mainInicial SalidaLogin;

public int flagRol;

public int flagmatricula;

public String usrActivo;


    public controlador_evaluacion(modelo_evaluacion miModelo, ServEvaluacion miEvaluacion,
principalMenu miMenu, mainInicial SalidaLogin) {

        this.miModelo = miModelo;

        this.miEvaluacion = miEvaluacion;

        this.miMenu = miMenu;

        this.SalidaLogin = SalidaLogin;


        this.miEvaluacion.cb_servicio.addActionListener(this);

        this.miEvaluacion.btn_enviarEvaluacion.addActionListener(this);

        this.miEvaluacion.cb_usaria.addActionListener(this);

        this.miEvaluacion.cb_usaria.setEnabled(false);

        this.miEvaluacion.rdBtn_AtBueno.setEnabled(false);

        this.miEvaluacion.rdBtn_AtExcelente.setEnabled(false);

        this.miEvaluacion.rdBtn_AtMalo.setEnabled(false);

        this.miEvaluacion.rdBtn_AtRegular.setEnabled(false);

        this.miEvaluacion.rdBtn_ServBueno.setEnabled(false);

        this.miEvaluacion.rdBtn_ServExcelente.setEnabled(false);

        this.miEvaluacion.rdBtn_ServMalo.setEnabled(false);

```

```
this.miEvaluacion.rdBtn_ServReglar.setEnabled(false);

this.miEvaluacion.btn_enviarEvaluacion.setEnabled(false);

this.miEvaluacion.rdBtn_AtExcelente.setSelected(true);

this.miEvaluacion.rdBtn_ServExcelente.setSelected(true);

this.miEvaluacion.jMenuItem_servCaja.addActionListener(this);

this.miEvaluacion.jMenuItem_servCreditoUT.addActionListener(this);

this.miEvaluacion.jMenuItem_servLoboletos.addActionListener(this);

this.miEvaluacion.jMenuItem_servEvaluacion.addActionListener(this);

this.miEvaluacion.jMenuItem_consCalificaciones.addActionListener(this);

this.miEvaluacion.jMenuItem_consHorario.addActionListener(this);

this.miEvaluacion.jMenuItem_consEdoCuenta.addActionListener(this);

this.miEvaluacion.jMenuItem_ConsMapa.addActionListener(this);

this.miEvaluacion.jMenuItem_calendario.addActionListener(this);

this.miEvaluacion.jMenuItem_becas.addActionListener(this);

this.miEvaluacion.jMenuItem_constancias.addActionListener(this);

this.miEvaluacion.jMenuItem_correo.addActionListener(this);

this.miEvaluacion.jMenuItem_navega.addActionListener(this);

this.miEvaluacion.jMenuItem_soporte.addActionListener(this);

this.miEvaluacion.jMenuItem_servBuzon.addActionListener(this);

this.miEvaluacion.jMenu_cerrarSesion.addMenuListener(this);

flagRol = 0;

flagmatricula = 0;

usrActivo = "";

}
```

```

@Override

//Método que recibe los eventos de acciones
public void actionPerformed(ActionEvent ae) {

    if (ae.getSource() == miEvaluacion.cb_servicio) {

        String itemSelec = (String)miEvaluacion.cb_servicio.getSelectedItem();

        if (!("Seleccionar".equals(itemSelec))) {

            this.miEvaluacion.cb_usaria.setEnabled(true);

            this.miEvaluacion.rdBtn_AtBueno.setEnabled(true);

            this.miEvaluacion.rdBtn_AtExcelente.setEnabled(true);

            this.miEvaluacion.rdBtn_AtMalo.setEnabled(true);

            this.miEvaluacion.rdBtn_AtRegular.setEnabled(true);

            this.miEvaluacion.rdBtn_ServBueno.setEnabled(true);

            this.miEvaluacion.rdBtn_ServExcelente.setEnabled(true);

            this.miEvaluacion.rdBtn_ServMalo.setEnabled(true);

            this.miEvaluacion.rdBtn_ServReglar.setEnabled(true);

            this.miEvaluacion.btn_enviarEvaluacion.setEnabled(true);

        }

        else {

            this.miEvaluacion.cb_usaria.setEnabled(false);

            this.miEvaluacion.rdBtn_AtBueno.setEnabled(false);

            this.miEvaluacion.rdBtn_AtExcelente.setEnabled(false);

            this.miEvaluacion.rdBtn_AtMalo.setEnabled(false);

            this.miEvaluacion.rdBtn_AtRegular.setEnabled(false);

            this.miEvaluacion.rdBtn_ServBueno.setEnabled(false);

            this.miEvaluacion.rdBtn_ServExcelente.setEnabled(false);

```

```

        this.miEvaluacion.rdBtn_ServMalo.setEnabled(false);

        this.miEvaluacion.rdBtn_ServReglar.setEnabled(false);

        this.miEvaluacion.btn_enviarEvaluacion.setEnabled(false);
    }
}

if (ae.getSource() == miEvaluacion.btn_enviarEvaluacion) {
    JOptionPane.showMessageDialog(null, "Tu evaluación se ha enviado correctamente.");
}

//Al clic en la opción caja del menú
if (ae.getSource() == miEvaluacion.jMenuItem_servCaja) {
    //Instancia de una nueva caja
    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja coómo parametro paara el constructor en el controlador de caja
    controlador_servCaja miControladorCaja = new controlador_servCaja(miCaja,
    modeloCaja,miMenu, SalidaLogin);

    //Desde el controlador se hace visible la caja
    miControladorCaja.miCaja.setVisible(true);

    this.miEvaluacion.dispose();
}

//Al clic en la opción CreditoUT del menú
if (ae.getSource() == miEvaluacion.jMenuItem_servCreditoUT) {
    //Instancia de CreditoUT
    modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();

    ServCreditoUT vistaCreditoUt = new ServCreditoUT();
}

```

```

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_creditoUT          miControladorCreditoUT          =          new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUT, miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCreditoUT.flagRol = this.flagRol;

miControladorCreditoUT.flagmatricula = this.flagmatricula;

miControladorCreditoUT.usrActivo = this.usrActivo;

miControladorCreditoUT.cargaEstadoCreditoUT();

//Desde el controlador se hace visible el buzón

miControladorCreditoUT.miCreditoUT.setVisible(true);

this.miEvaluacion.dispose();
}

//Al clic en la opción Loboletos del menú

if (ae.getSource() == miEvaluacion.jMenuItem_servLoboletos) {

    //Instancia de una nueva caja

    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

    controlador_servCaja    miControladorCaja    =    new    controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCaja.flagRol = this.flagRol;

miControladorCaja.flagmatricula = this.flagmatricula;

miControladorCaja.usrActivo = this.usrActivo;

miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

```



```

//Desde el controlador se hace visible la caja

miControladorCaja.miCaja.setVisible(true);

this.miEvaluacion.dispose();
}

//Al clic en la opción Evaluación del menú

if (ae.getSource() == miEvaluacion.jMenuItem_servEvaluacion) {

    //Instancia de buzón

    modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

    ServEvaluacion vistaEvaluacion = new ServEvaluacion();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_evaluacion      miControladorEvaluacion      =      new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorEvaluacion.flagRol = this.flagRol;

    miControladorEvaluacion.flagmatricula = this.flagmatricula;

    miControladorEvaluacion.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón

    miControladorEvaluacion.miEvaluacion.setVisible(true);

    this.miEvaluacion.dispose();
}

//Al clic en la opción Buzón del menú

if (ae.getSource() == miEvaluacion.jMenuItem_servBuzon) {

    //Instancia de buzón

    modelo_buzon modeloBuzon = new modelo_buzon();

    ServBuzon vistaBuzon = new ServBuzon();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

```

```

        controlador_buzon  miControladorBuzon  =  new  controlador_buzon(modeloBuzon,
vistaBuzon, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorBuzon.flagRol = this.flagRol;

        miControladorBuzon.flagmatricula = this.flagmatricula;

        miControladorBuzon.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón

        miControladorBuzon.miBuzon.setVisible(true);

        this.miEvaluacion.dispose();
    }

    //Si se da clic en el menú a la opción Calificaciones
    if (ae.getSource() == miEvaluacion.jMenuItem_consCalificaciones) {

        File archivo = new File(miModelo.abreCalif(flagmatricula));

        miModelo.abreArchivo(archivo);
    }

    //Si se da clic en el menú a la opción Horario

    if (ae.getSource() == miEvaluacion.jMenuItem_consHorario) {

        JOptionPane.showInputDialog("La url es: "+miModelo.abreHorario(flagmatricula));

        miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

        //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
dic-2019/TICASI-7-E-9.pdf");
    }

    //Si se da clic en el menú a la opción Estado de cuenta

    if (ae.getSource() == miEvaluacion.jMenuItem_consEdoCuenta) {

        File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));

        miModelo.abreArchivo(archivo);
    }

```

```

    }

    //Si se da clic en el menu a la opción de ver mapa
    if (ae.getSource() == miEvaluacion.jMenuItem_ConsMapa) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/mapaUT.jpg");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción de ver calendario
    if (ae.getSource() == miEvaluacion.jMenuItem_calendario) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/calendario.pdf");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción becas
    if (ae.getSource() == miEvaluacion.jMenuItem_becas) {

        miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

    }

    //Al clic en la opción Constancias del menú
    if (ae.getSource() == miEvaluacion.jMenuItem_constancias) {

        //Instancia de buzón

        modelo_constancia modeloConstancia = new modelo_constancia();

        TramConstancia vistaConstancia = new TramConstancia();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

```

```

        controlador_constancia        miControladorConstancia        =        new
controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorConstancia.flagRol = this.flagRol;

        miControladorConstancia.flagmatricula = this.flagmatricula;

        miControladorConstancia.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón

        miControladorConstancia.miConstancia.setVisible(true);

        this.miEvaluacion.dispose();
    }

    //Si se da clic en el menú a la opción correo
    if (ae.getSource() == miEvaluacion.jMenuItem_correo) {

        miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");

    }

    //Si se da clic en el menú a la opción navegar
    if (ae.getSource() == miEvaluacion.jMenuItem_navega) {

        miModelo.abreNavegador("http://www.google.com.mx");

    }

    //Al clic en la opción Soporte del menú
    if (ae.getSource() == miEvaluacion.jMenuItem_soporte) {

        //Instancia de buzón

        modelo_soporte modeloSoporte = new modelo_soporte();

        ServSoporte vistaSoporte = new ServSoporte();

        //se pasa la nueva caja como parametro para el constructor en el controlador de buzón

        controlador_soporte miControladorSoporte = new controlador_soporte(modeloSoporte,
vistaSoporte, miMenu, SalidaLogin);
    }

```

```

        //Paso los datos de usuario

        miControladorSoporte.flagRol = this.flagRol;

        miControladorSoporte.flagmatricula = this.flagmatricula;

        miControladorSoporte.usrActivo = this.usrActivo;

        miControladorSoporte.mensajeUsuarioSoporte();

        //Desde el controlador se hace visible el buzón

        miControladorSoporte.miSoporte.setVisible(true);

        this.miEvaluacion.dispose();

    }
}

@Override

public void windowOpened(WindowEvent we) {

}

@Override

public void windowClosing(WindowEvent we) {

}

@Override

public void windowClosed(WindowEvent we) {

}

@Override

```

```

public void windowIconified(WindowEvent we) {

}

@Override

public void windowDeiconified(WindowEvent we) {

}

@Override

public void windowActivated(WindowEvent we) {

}

@Override

public void windowDeactivated(WindowEvent we) {

}

@Override

//Si se preciona en el menú cerrar sesión

public void menuSelected(MenuEvent e) {

    int reply = JOptionPane.showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar
sesión", JOptionPane.YES_NO_CANCEL_OPTION);

    if(reply == JOptionPane.YES_OPTION) {

        //Carga el método que inicia el login desde la clase main y cierra la ventana acual

        SalidaLogin.cargaLogin();

        this.miEvaluacion.dispose();

    }

    else {

```

```

    }

}

@Override

public void menuDeselected(MenuEvent e) {

}

@Override

public void menuCanceled(MenuEvent e) {

}

}
}

```

Controlador_login

```

package Controlador;

import Vista.*;

import Modelo.*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JOptionPane;

import javax.swing.ImageIcon;

import java.awt.event.WindowEvent;

import java.awt.event.WindowListener;

public class controlador_login implements ActionListener, WindowListener {

    Login miLogin = new Login();

```

```
modelo_login miModeloLogin = new modelo_login();

//Constructor que inicializa los objetos vista y modelo

public void setMiVistaModelo(Login miLogin, modelo_login miModeloLogin) {

    this.miLogin = miLogin;

    this.miModeloLogin = miModeloLogin;

    //Hago visible el login

    miLogin.setVisible(true);

    miLogin.btn_login.addActionListener(this);

    this.miLogin.tBtn_tecladoMatricula.addActionListener(this);

    this.miLogin.tBtn_tecladoPass.addActionListener(this);

    this.miLogin.txt_loginMatricula.enable(false);

    this.miLogin.txt_loginPass.enable(false);

    this.miLogin.frame_servTeclado.addWindowListener(this);

    this.miLogin.teclado_btn0.addActionListener(this);

    this.miLogin.teclado_btn1.addActionListener(this);

    this.miLogin.teclado_btn2.addActionListener(this);

    this.miLogin.teclado_btn3.addActionListener(this);

    this.miLogin.teclado_btn4.addActionListener(this);

    this.miLogin.teclado_btn5.addActionListener(this);

    this.miLogin.teclado_btn6.addActionListener(this);

    this.miLogin.teclado_btn7.addActionListener(this);

    this.miLogin.teclado_btn8.addActionListener(this);

    this.miLogin.teclado_btn9.addActionListener(this);

    this.miLogin.teclado_btnAtras.addActionListener(this);

    this.miLogin.teclado_btnLimpiar.addActionListener(this);
```



```

        this.miLogin.teclado_btnAceptar.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e) {

        //Autentifica al usuario comparando si la información es correcta mediante la función
        autentifica en modelo_login

        if(e.getSource() == miLogin.btn_login) {

            if(miLogin.txt_loginMatricula.getText().equals("") ||
            miLogin.txt_loginPass.getText().equals("")) {

                JOptionPane.showMessageDialog(null, "Favor de llenar los campos");

            }

            else {

                if(miModeloLogin.autentifica(Integer.parseInt(miLogin.txt_loginMatricula.getText()),Integer.parseInt(
                miLogin.txt_loginPass.getText())) {

                    JOptionPane.showMessageDialog(null, "Hola " +miModeloLogin.usuarioActivo+",
                    Bienvenido/a");

                    //Si la información es correcta abrirá el menú principal

                    AbrirMenu();

                }

                else {

                    JOptionPane.showMessageDialog(null, "Información incorrecta");

                }

            }

        }

        /*if(e.getSource() == miLogin.btn_login) {

            if(miModeloLogin.autentifica(miLogin.txt_loginMatricula.getText(),
            miLogin.txt_loginPass.getText())) {

                JOptionPane.showMessageDialog(null, "Acceso permitido");

```

```

        //Si la información es correcta abrirá el menú principal

        AbrirMenu();

    }

    else {

        JOptionPane.showMessageDialog(null, "Información incorrecta");

    }

}*/

//Si se cliquea el toggle button de matricula

if (e.getSource() == miLogin.tBtn_tecladoMatricula) {

    /*Imagelcon icon = new Imagelcon("src/imagenes/cena.gif");

    JOptionPane.showMessageDialog(null, "Tecleo matricula", "Seguro",
    JOptionPane.PLAIN_MESSAGE,icon);*/

    //controla estados de los botones y muestra el teclado

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.frame_servTeclado.setVisible(true);

        miLogin.tBtn_tecladoPass.setSelected(false);

    } else {

        miLogin.frame_servTeclado.setVisible(false);

    }

}

//En caso de que se clique el toggle de password

else if (e.getSource() == miLogin.tBtn_tecladoPass) {

    if (miLogin.tBtn_tecladoPass.isSelected()) {

        miLogin.tBtn_tecladoPass.setSelected(false);

        miLogin.frame_servTeclado.setVisible(true);

        miLogin.teclado_btnAtras.setVisible(false);

```

```

    } else {

        miLogin.frame_servTeclado.setVisible(false);

    }

}

//Eventos del clic de botones del teclado controlado por toggle buttons

if (e.getSource() == miLogin.teclado_btn0) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"0");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "0");

    }

}

if (e.getSource() == miLogin.teclado_btn1) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"1");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "1");

    }

}

```

```

    }
}

if (e.getSource() == miLogin.teclado_btn2) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"2");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "2");

    }

}

if (e.getSource() == miLogin.teclado_btn3) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"3");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "3");

    }

}

if (e.getSource() == miLogin.teclado_btn4) {

```

```

String nTarjeta = miLogin.txt_loginMatricula.getText();

String nCVE = miLogin.txt_loginPass.getText();


if (miLogin.tBtn_tecladoMatricula.isSelected()) {

    miLogin.txt_loginMatricula.setText(nTarjeta+"4");

}

else {

    miLogin.txt_loginPass.setText(nCVE + "4");

}

}

if (e.getSource() == miLogin.teclado_btn5) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();


    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"5");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "5");

    }

}

if (e.getSource() == miLogin.teclado_btn6) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

```

```

if (miLogin.tBtn_tecladoMatricula.isSelected()) {

    miLogin.txt_loginMatricula.setText(nTarjeta+"6");

}

else {

    miLogin.txt_loginPass.setText(nCVE + "6");

}

}

if (e.getSource() == miLogin.teclado_btn7) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();


    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"7");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "7");

    }

}

if (e.getSource() == miLogin.teclado_btn8) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();


    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"8");

    }

}

```

```

else {

    miLogin.txt_loginPass.setText(nCVE + "8");

}

}

if (e.getSource() == miLogin.teclado_btn9) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText(nTarjeta+"9");

    }

    else {

        miLogin.txt_loginPass.setText(nCVE + "9");

    }

}

//Acciones de limpiar campos con el botón C del teclado

if (e.getSource() == miLogin.teclado_btnLimpiar) {

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        miLogin.txt_loginMatricula.setText("");

    }

    else {

        miLogin.txt_loginPass.setText("");

    }

}

//Borrar último carácter de las cajas de texto y password

```

```

if (e.getSource() == miLogin.teclado_btnAtras) {

    String nTarjeta = miLogin.txt_loginMatricula.getText();

    String nCVE = miLogin.txt_loginPass.getText();

    if (miLogin.tBtn_tecladoMatricula.isSelected()) {

        if(nTarjeta.length() != 0) {

            nTarjeta = nTarjeta.substring(0, nTarjeta.length()-1);

            miLogin.txt_loginMatricula.setText(nTarjeta);

        }

        else {}

    }

    else {}

}

//Cerrar el teclado

if (e.getSource() == miLogin.teclado_btnAceptar) {

    miLogin.frame_servTeclado.dispose();

    miLogin.tBtn_tecladoMatricula.setSelected(false);

    miLogin.tBtn_tecladoPass.setSelected(false);

}

}

//método para abrir el menú

public void AbrirMenu(){

    //this.miLogin.dispose();

    //Instancio el modelo y el mwnu y se lo mando cómo parametro al constructor del controlador
    de menu

    modelo_principalMenu modeloMenu = new modelo_principalMenu();

```



```

principalMenu miMenu = new principalMenu();

mainInicial miMain = new mainInicial();

controlador_principalMenu controladorMenu = new
controlador_principalMenu(modeloMenu, miMenu, miMain);

controladorMenu.flagRol = miModeloLogin.flagRol;

controladorMenu.flagmatricula = miModeloLogin.flagMatricula;

controladorMenu.usrActivo = miModeloLogin.usuarioActivo;

controladorMenu.setUsrVisible(miModeloLogin.usuarioActivo);

if(miModeloLogin.flagRol == 1) {

    controladorMenu.miMenu.jMenuItem_servCaja.setVisible(false);

    controladorMenu.miMenu.jMenuItem_servCreditoUT.setVisible(false);

    controladorMenu.miMenu.jMenuItem_servLoboletos.setVisible(false);

    controladorMenu.miMenu.jMenuItem_servEvaluacion.setVisible(false);

    controladorMenu.miMenu.jMenuItem_consCalificaciones.setVisible(false);

    controladorMenu.miMenu.jMenuItem_consEdoCuenta.setVisible(false);

    controladorMenu.miMenu.jMenuItem_tramites.setVisible(false);

}

else {

}

//Abro el menu principal

controladorMenu.miMenu.setVisible(true);

//oculto el login

this.miLogin.setVisible(false);

}

@Override

```

```
public void windowOpened(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowClosing(WindowEvent we) {  
    miLogin.tBtn_tecladoMatricula.setSelected(false);  
    this.miLogin.tBtn_tecladoPass.setSelected(false);  
}
```

```
@Override
```

```
public void windowClosed(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowIconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowDeiconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowActivated(WindowEvent we) {  
}
```

```
@Override
```

```
        public void windowDeactivated(WindowEvent we) {  
        }  
    }  
}
```

Controlador_principalMenu

```
package Controlador;  
  
import Modelo.*;  
  
import Vista.*;  
  
import java.awt.Desktop;  
  
import java.awt.event.ActionEvent;  
  
import java.awt.event.ActionListener;  
  
import java.io.File;  
  
import java.io.IOException;  
  
import java.util.logging.Level;  
  
import java.util.logging.Logger;  
  
import javax.swing.BorderFactory;  
  
import javax.swing.JOptionPane;  
  
import javax.swing.*;  
  
import javax.swing.event.*;  
  
  
public class controlador_principalMenu implements ActionListener, MenuListener {  
  
    public modelo_principalMenu miModelo;  
  
    public principalMenu miMenu;  
  
    public mainInicial SalidaLogin;  
  
    public int flagRol;  
  
    public int flagmatricula;
```

```

public String usrActivo;

//Constructor del controlador

public controlador_principalMenu(modelo_principalMenu miModelo, principalMenu miMenu,
mainInicial SalidaLogin){

    this.miMenu = miMenu;

    this.miModelo = miModelo;

    this.SalidaLogin = SalidaLogin;

    this.miMenu.jMenuItem_servCaja.addActionListener(this);

    this.miMenu.jMenuItem_servCreditoUT.addActionListener(this);

    this.miMenu.jMenuItem_consCalificaciones.addActionListener(this);

    this.miMenu.jMenuItem_consHorario.addActionListener(this);

    this.miMenu.jMenuItem_consEdoCuenta.addActionListener(this);

    this.miMenu.jMenuItem_ConsMapa.addActionListener(this);

    this.miMenu.jMenuItem_calendario.addActionListener(this);

    this.miMenu.jMenuItem_becas.addActionListener(this);

    this.miMenu.jMenuItem_constancias.addActionListener(this);

    this.miMenu.jMenuItem_correo.addActionListener(this);

    this.miMenu.jMenuItem_navega.addActionListener(this);

    this.miMenu.jMenuItem_soporte.addActionListener(this);

    this.miMenu.jMenuItem_servBuzon.addActionListener(this);

    this.miMenu.jMenuItem_servLoboletos.addActionListener(this);

    this.miMenu.jMenuItem_servEvaluacion.addActionListener(this);

    this.miMenu.jMenuItem_cerrarSesion.addMenuListener(this);

    flagRol = 0;

    flagmatricula = 0;

```

```

        usrActivo = "";
    }

    //Metodo que recibe los eventos de acción mediante la evaluación de condicionales
    public void actionPerformed(ActionEvent ae) {

        //Al clic en la opción caja del menú
        if (ae.getSource() == miMenu.jMenuItem_servCaja) {

            //Instancia de una nueva caja
            servCaja miCaja = new servCaja();

            modelo_servCaja modeloCaja = new modelo_servCaja();

            //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja
            controlador_servCaja miControladorCaja = new controlador_servCaja(miCaja,
            modeloCaja,miMenu, SalidaLogin);

            //Paso los datos de usuario
            miControladorCaja.flagRol = this.flagRol;

            miControladorCaja.flagmatricula = this.flagmatricula;

            miControladorCaja.usrActivo = this.usrActivo;

            //Desde el controlador se hace visible la caja
            miControladorCaja.miCaja.setVisible(true);

            this.miMenu.dispose();
        }

        //Al clic en la opción CreditoUT del menú
        if (ae.getSource() == miMenu.jMenuItem_servCreditoUT) {

            //Instancia de CreditoUT
            modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();

            ServCreditoUT vistaCreditoUt = new ServCreditoUT();

```

```

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

controlador_creditoUT      miControladorCreditoUT      =      new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUT, miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCreditoUT.flagRol = this.flagRol;

miControladorCreditoUT.flagmatricula = this.flagmatricula;

miControladorCreditoUT.usrActivo = this.usrActivo;

miControladorCreditoUT.cargaEstadoCreditoUT();

//Desde el controlador se hace visible el buzón

miControladorCreditoUT.miCreditoUT.setVisible(true);

this.miMenu.dispose();
}

//Al clic en la opción Loboletos del menú

if (ae.getSource() == miMenu.jMenuItem_servLoboletos) {

//Instancia de una nueva caja

servCaja miCaja = new servCaja();

modelo_servCaja modeloCaja = new modelo_servCaja();

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

controlador_servCaja      miControladorCaja      =      new      controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCaja.flagRol = this.flagRol;

miControladorCaja.flagmatricula = this.flagmatricula;

miControladorCaja.usrActivo = this.usrActivo;

miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

```

```

//Desde el controlador se hace visible la caja
miControladorCaja.miCaja.setVisible(true);

this.miMenu.dispose();
}

//Al clic en la opción Evaluación del menú
if (ae.getSource() == miMenu.jMenuItem_servEvaluacion) {

    //Instancia de buzón
    modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

    ServEvaluacion vistaEvaluacion = new ServEvaluacion();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
    controlador_evaluacion miControladorEvaluacion = new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

    //Paso los datos de usuario
    miControladorEvaluacion.flagRol = this.flagRol;

    miControladorEvaluacion.flagmatricula = this.flagmatricula;

    miControladorEvaluacion.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón
    miControladorEvaluacion.miEvaluacion.setVisible(true);

    this.miMenu.dispose();
}

//Al clic en la opción Buzón del menú
if (ae.getSource() == miMenu.jMenuItem_servBuzon) {

    //Instancia de buzón
    modelo_buzon modeloBuzon = new modelo_buzon();

    ServBuzon vistaBuzon = new ServBuzon();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

```

```

        controlador_buzon  miControladorBuzon  =  new  controlador_buzon(modeloBuzon,
vistaBuzon, miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorBuzon.flagRol = this.flagRol;

miControladorBuzon.flagmatricula = this.flagmatricula;

miControladorBuzon.usrActivo = this.usrActivo;

if(miControladorBuzon.flagRol == 1) {

    miControladorBuzon.miBuzon.jMenuItem_servCaja.setVisible(false);

    miControladorBuzon.miBuzon.jMenuItem_servCreditoUT.setVisible(false);

    miControladorBuzon.miBuzon.jMenuItem_servLoboletos.setVisible(false);

    miControladorBuzon.miBuzon.jMenuItem_servEvaluacion.setVisible(false);

    miControladorBuzon.miBuzon.jMenuItem_consCalificaciones.setVisible(false);

    miControladorBuzon.miBuzon.jMenuItem_consEdoCuenta.setVisible(false);

    miControladorBuzon.miBuzon.jMenuItem_tramites.setVisible(false);

}

else {

}

//Desde el controlador se hace visible el buzón

miControladorBuzon.miBuzon.setVisible(true);

this.miMenu.dispose();

}

//Si se da clic en el menú a la opción Calificaciones

if (ae.getSource() == miMenu.jMenuItem_consCalificaciones) {

    File archivo = new File(miModelo.abreCalif(flagmatricula));

    miModelo.abreArchivo(archivo);

```



```

}

//Si se da clic en el menu a la opción Horario
if (ae.getSource() == miMenu.jMenuItem_consHorario) {

    JOptionPane.showInputDialog("La url es: "+miModelo.abreHorario(flagmatricula));

    miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

    //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
dic-2019/TICASI-7-E-9.pdf");

}

//Si se da clic en el menu a la opción Estado de cuenta
if (ae.getSource() == miMenu.jMenuItem_consEdoCuenta) {

    File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción de ver mapa
if (ae.getSource() == miMenu.jMenuItem_ConsMapa) {

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
proyecto

    File archivo = new File("src/Documentos/mapaUT.jpg");

    miModelo.abreArchivo(archivo);

}

//Si se da clic en el menu a la opción de ver calendario
if (ae.getSource() == miMenu.jMenuItem_calendario) {

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
proyecto

    File archivo = new File("src/Documentos/calendario.pdf");

    miModelo.abreArchivo(archivo);

```

```

    }

    //Si se da clic en el menu a la opción becas
    if (ae.getSource() == miMenu.jMenuItem_becas) {

        miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

    }

    //Al clic en la opción Constancias del menú
    if (ae.getSource() == miMenu.jMenuItem_constancias) {

        //Instancia de buzón

        modelo_constancia modeloConstancia = new modelo_constancia();

        TramConstancia vistaConstancia = new TramConstancia();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

        controlador_constancia          miControladorConstancia          =          new
controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorConstancia.flagRol = this.flagRol;

        miControladorConstancia.flagmatricula = this.flagmatricula;

        miControladorConstancia.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón

        miControladorConstancia.miConstancia.setVisible(true);

        this.miMenu.dispose();

    }

    //Si se da clic en el menu a la opción correo
    if (ae.getSource() == miMenu.jMenuItem_correo) {

        miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");

    }

    //Si se da clic en el menu a la opción navegar

```

```

if (ae.getSource() == miMenu.jMenuItem_navega) {

    miModelo.abreNavegador("http://www.google.com.mx");

}

//Al clic en la opción Soporte del menú

if (ae.getSource() == miMenu.jMenuItem_soporte) {

    //Instancia de Soporte

    modelo_soporte modeloSoporte = new modelo_soporte();

    ServSoporte vistaSoporte = new ServSoporte();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

    controlador_soporte miControladorSoporte = new controlador_soporte(modeloSoporte,
    vistaSoporte, miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorSoporte.flagRol = this.flagRol;

    miControladorSoporte.flagmatricula = this.flagmatricula;

    miControladorSoporte.usrActivo = this.usrActivo;

    miControladorSoporte.mensajeUsuarioSoporte();

    if(miControladorSoporte.flagRol == 1) {

        miControladorSoporte.miSoporte.jMenuItem_servCaja.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_servCreditoUT.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_servLoboletos.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_servEvaluacion.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_consCalificaciones.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_consEdoCuenta.setVisible(false);

        miControladorSoporte.miSoporte.jMenuItem_tramites.setVisible(false);

    }

    else {

```

```

    }

    //Desde el controlador se hace visible el Soporte
    miControladorSoporte.miSoporte.setVisible(true);

    this.miMenu.dispose();
}

}

@Override

//Si se preciona en el menú cerrar sesión
public void menuSelected(MenuEvent e) {

    int reply = JOptionPane.showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar
sesión", JOptionPane.YES_NO_CANCEL_OPTION);

    if(reply == JOptionPane.YES_OPTION) {

        //Carga el método que incia el login desde la clase main y cierra la ventana acual

        SalidaLogin.cargaLogin();

        this.miMenu.dispose();

    }

    else {

    }

}

@Override

public void menuDeselected(MenuEvent e) {

}

```

```

@Override

public void menuCanceled(MenuEvent e) {

}

/*public void Abrir_servCaja() {

    miMenu.dispose();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    servCaja vistaCaja = new servCaja();

    controlador_servCaja controladorCaja = new controlador_servCaja(vistaCaja, modeloCaja,
miMenu);

    controladorCaja.miCaja.setVisible(true);

}*/

public void setUsrVisible(String usr) {

    miMenu.lbl_usrSaudo.setText("Usuario activo: "+usr);

}

}

```

Controlador_ServCaja

```

package Controlador;

import Modelo.*;

import Vista.*;

import java.awt.Desktop;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

```

```

import java.awt.event.WindowEvent;

import java.awt.event.WindowListener;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.BorderFactory;

import javax.swing.JOptionPane;

import javax.swing.*;

import javax.swing.event.*;

import javax.swing.event.MenuEvent;

import javax.swing.event.MenuListener;


/**
 *
 * @author Usuario
 */
public class controlador_servCaja implements ActionListener, WindowListener, MenuListener {

    public modelo_servCaja miModelo;

    public servCaja miCaja;

    public principalMenu miMenu;

    public mainInicial SalidaLogin;

    public int flagRol;

    public int flagmatricula;

    public String usrActivo;

```

```
//Constructor que inicializa todos los elementos que se utilizaan en la pantalla de Caja

public controlador_servCaja(servCaja miCaja, modelo_servCaja miModelo, principalMenu
miMenu, mainInicial SalidaLogin) {

    this.miModelo = miModelo;

    this.miCaja = miCaja;

    this.miMenu = miMenu;

    this.miCaja.cb_tipoPago.addActionListener(this);

    this.miCaja.rd_pagoCreditoUT.addActionListener(this);

    this.miCaja.rd_pagoTarjeta.addActionListener(this);

    this.miCaja.txt_numTarjeta.addActionListener(this);

    this.miCaja.txt_numTarjeta.setEnabled(false);

    this.miCaja.cb_tipoPago_loboleto.addActionListener(this);

    this.miCaja.cb_tipoPago_loboleto.setEnabled(false);

    this.miCaja.tBtn_tecladoTarjeta.addActionListener(this);

    this.miCaja.pass_cve.addActionListener(this);

    this.miCaja.pass_cve.setEnabled(false);

    this.miCaja.tBtn_tecladoCVE.addActionListener(this);

    this.miCaja.cb_vencimientoMes.addActionListener(this);

    this.miCaja.cb_vencimientoAnio.addActionListener(this);

    this.miCaja.txtArea_recibo.setEnabled(false);

    this.miCaja.btn_pagar.addActionListener(this);

    this.miCaja.frame_servTeclado.addWindowListener(this);

    this.miCaja.teclado_btn0.addActionListener(this);

    this.miCaja.teclado_btn1.addActionListener(this);

    this.miCaja.teclado_btn2.addActionListener(this);
```

```
this.miCaja.teclado_btn3.addActionListener(this);

this.miCaja.teclado_btn4.addActionListener(this);

this.miCaja.teclado_btn5.addActionListener(this);

this.miCaja.teclado_btn6.addActionListener(this);

this.miCaja.teclado_btn7.addActionListener(this);

this.miCaja.teclado_btn8.addActionListener(this);

this.miCaja.teclado_btn9.addActionListener(this);

this.miCaja.teclado_btnAtras.addActionListener(this);

this.miCaja.teclado_btnLimpiar.addActionListener(this);

this.miCaja.teclado_btnAceptar.addActionListener(this);


this.SalidaLogin = SalidaLogin;

this.miCaja.jMenuItem_servCaja.addActionListener(this);

this.miCaja.jMenuItem_servCreditoUT.addActionListener(this);

this.miCaja.jMenuItem_servLoboletos.addActionListener(this);

this.miCaja.jMenuItem_servEvaluacion.addActionListener(this);

this.miCaja.jMenuItem_consCalificaciones.addActionListener(this);

this.miCaja.jMenuItem_consHorario.addActionListener(this);

this.miCaja.jMenuItem_consEdoCuenta.addActionListener(this);

this.miCaja.jMenuItem_ConsMapa.addActionListener(this);

this.miCaja.jMenuItem_calendario.addActionListener(this);

this.miCaja.jMenuItem_becas.addActionListener(this);

this.miCaja.jMenuItem_constancias.addActionListener(this);

this.miCaja.jMenuItem_correo.addActionListener(this);

this.miCaja.jMenuItem_navega.addActionListener(this);
```



```

this.miCaja.jMenuItem_soporte.addActionListener(this);

this.miCaja.jMenu_cerrarSesion.addMenuListener(this);

this.miCaja.jMenuItem_servBuzon.addActionListener(this);

flagRol = 0;

flagmatricula = 0;

usrActivo = "";

//Al presentar la vista Caja se comprueba el estado de los radio buttons
radioPagoValida();
}

```

```
@Override
```

```
//Método que recibe los eventos de acciones
```

```
public void actionPerformed(ActionEvent ae) {
```

```
    //Comparación de selección del combobox
```

```
    if (ae.getSource() == miCaja.cb_tipoPago) {
```

```
        String itemSelec = (String)miCaja.cb_tipoPago.getSelectedItem();
```

```
        if ("Colegiatura".equals(itemSelec)) {
```

```
            miCaja.cb_tipoPago_loboleto.setEnabled(false);
```

```
            miCaja.cb_tipoPago_loboleto.setSelectedIndex(0);
```

```
            miCaja.txtArea_recibo.setText("PAGO COLEGIATURA...");
```

```
        }
```

```
        else if ("Inscripción".equals(itemSelec)) {
```

```
            miCaja.cb_tipoPago_loboleto.setEnabled(false);
```

```
            miCaja.cb_tipoPago_loboleto.setSelectedIndex(0);
```

```
            miCaja.txtArea_recibo.setText("PAGO INSCRIPCIÓN...");
```

```

    }

    else if ("Examen".equals(itemSelec)) {

        miCaja.cb_tipoPago_loboleto.setEnabled(false);

        miCaja.cb_tipoPago_loboleto.setSelectedIndex(0);

        miCaja.txtArea_recibo.setText("PAGO EXAMENES...");

    }

    else if ("Titulación".equals(itemSelec)) {

        miCaja.cb_tipoPago_loboleto.setEnabled(false);

        miCaja.cb_tipoPago_loboleto.setSelectedIndex(0);

        miCaja.txtArea_recibo.setText("PAGO TITULACIÓN...");

    }

    else if ("Loboletos".equals(itemSelec)) {

        miCaja.txtArea_recibo.setText("PAGO LOBOLETOS...");

        miCaja.cb_tipoPago_loboleto.setEnabled(true);

    }

}

if (ae.getSource() == miCaja.cb_tipoPago_loboleto) {

    String itemSelec2 = (String)miCaja.cb_tipoPago_loboleto.getSelectedItem();

    if ("Seleccionar".equals(itemSelec2)) {

        miCaja.txtArea_recibo.setText("PAGO LOBOLETOS...");

    }

    else if ("Lobobus".equals(itemSelec2)) {

        miCaja.txtArea_recibo.setText("PAGO LOBOLETOS: LOBOBUS...");

    }

    else if ("Impresiones".equals(itemSelec2)) {

```

```

        miCaja.txtArea_recibo.setText("PAGO LOBOLETOS: IMPRESIONES...");
    }

    else if ("Copias".equals(itemSelec2)) {

        miCaja.txtArea_recibo.setText("PAGO LOBOLETOS: COPIAS...");

    }

}

//Si cambia la selección del radio button group, mediante un método
if (ae.getSource() == miCaja.rd_pagoCreditoUT || ae.getSource() == miCaja.rd_pagoTarjeta) {

    radioPagoValida();

}

//Comparamos el estado de los toggle buttons
//Si se cliquea el toggle button de tarjeta
if (ae.getSource() == miCaja.tBtn_tecladoTarjeta) {

    //controla estados de los botones y muestra el teclado

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.frame_servTeclado.setVisible(true);

        miCaja.tBtn_tecladoCVE.setSelected(false);

    } else {

        miCaja.frame_servTeclado.setVisible(false);

    }

}

//En caso de que se clique el toggle de CVE
else if (ae.getSource() == miCaja.tBtn_tecladoCVE) {

    if (miCaja.tBtn_tecladoCVE.isSelected()) {

        miCaja.tBtn_tecladoTarjeta.setSelected(false);

    }

}

```

```

        miCaja.frame_servTeclado.setVisible(true);

        miCaja.teclado_btnAtras.setVisible(false);

    } else {

        miCaja.frame_servTeclado.setVisible(false);

    }

}

//Eventos del clic de botones del teclado controlado por toggle buttons
if (ae.getSource() == miCaja.teclado_btn0) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.txt_numTarjeta.setText(nTarjeta+"0");

    }

    else {

        miCaja.pass_cve.setText(nCVE + "0");

    }

}

if (ae.getSource() == miCaja.teclado_btn1) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.txt_numTarjeta.setText(nTarjeta+"1");

    }

}

```

```

else {

    miCaja.pass_cve.setText(nCVE + "1");

}

}

if (ae.getSource() == miCaja.teclado_btn2) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.txt_numTarjeta.setText(nTarjeta+"2");

    }

    else {

        miCaja.pass_cve.setText(nCVE + "2");

    }

}

if (ae.getSource() == miCaja.teclado_btn3) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.txt_numTarjeta.setText(nTarjeta+"3");

    }

    else {

        miCaja.pass_cve.setText(nCVE + "3");

    }

}

```

```

}

if (ae.getSource() == miCaja.teclado_btn4) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

    String nCVE = miCaja.pass_cve.getText();


    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.txt_numTarjeta.setText(nTarjeta+"4");

    }

    else {

        miCaja.pass_cve.setText(nCVE + "4");

    }

}

if (ae.getSource() == miCaja.teclado_btn5) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

    String nCVE = miCaja.pass_cve.getText();


    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

        miCaja.txt_numTarjeta.setText(nTarjeta+"5");

    }

    else {

        miCaja.pass_cve.setText(nCVE + "5");

    }

}

if (ae.getSource() == miCaja.teclado_btn6) {

    String nTarjeta = miCaja.txt_numTarjeta.getText();

```

```

String nCVE = miCaja.pass_cve.getText();

if (miCaja.tBtn_tecladoTarjeta.isSelected()) {
    miCaja.txt_numTarjeta.setText(nTarjeta+"6");
}
else {
    miCaja.pass_cve.setText(nCVE + "6");
}
}

if (ae.getSource() == miCaja.teclado_btn7) {
    String nTarjeta = miCaja.txt_numTarjeta.getText();
    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {
        miCaja.txt_numTarjeta.setText(nTarjeta+"7");
    }
    else {
        miCaja.pass_cve.setText(nCVE + "7");
    }
}

if (ae.getSource() == miCaja.teclado_btn8) {
    String nTarjeta = miCaja.txt_numTarjeta.getText();
    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

```

```

        miCaja.txt_numTarjeta.setText(nTarjeta+"8");
    }
    else {
        miCaja.pass_cve.setText(nCVE + "8");
    }
}

if (ae.getSource() == miCaja.teclado_btn9) {
    String nTarjeta = miCaja.txt_numTarjeta.getText();
    String nCVE = miCaja.pass_cve.getText();

    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {
        miCaja.txt_numTarjeta.setText(nTarjeta+"9");
    }
    else {
        miCaja.pass_cve.setText(nCVE + "9");
    }
}

//Acciones de limpiar campos con el botón C del teclado
if (ae.getSource() == miCaja.teclado_btnLimpiar) {
    if (miCaja.tBtn_tecladoTarjeta.isSelected()) {
        miCaja.txt_numTarjeta.setText("");
    }
    else {
        miCaja.pass_cve.setText("");
    }
}

```



```

    }

    //Borrar último caracter de las cajas de texto y password
    if (ae.getSource() == miCaja.teclado_btnAtras) {

        String nTarjeta = miCaja.txt_numTarjeta.getText();

        String nCVE = miCaja.pass_cve.getText();

        if (miCaja.tBtn_tecladoTarjeta.isSelected()) {

            if(nTarjeta.length() != 0) {

                nTarjeta = nTarjeta.substring(0, nTarjeta.length()-1);

                miCaja.txt_numTarjeta.setText(nTarjeta);

            }

            else {}

        }

        else {}

    }

    //Cerrar el teclado

    if (ae.getSource() == miCaja.teclado_btnAceptar) {

        miCaja.frame_servTeclado.dispose();

        miCaja.tBtn_tecladoCVE.setSelected(false);

        miCaja.tBtn_tecladoTarjeta.setSelected(false);

    }

    //Al dar clic en pagar

    if (ae.getSource() == miCaja.btn_pagar) {

        if((miCaja.rd_pagoTarjeta.isSelected())&&(miCaja.txt_numTarjeta.getText().equals("") ||
miCaja.pass_cve.getText().equals(""))) {

            JOptionPane.showMessageDialog(null, "ERROR: Favor de completar sus datos de pago");

        }

    }

```

```

else {

    JOptionPane.showMessageDialog(null, "Su pagao fue aceptado. Espere a que se genere su
comprobante, de igual forma este se enviará a su correo institucional");

    //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
proyecto

    File archivo = new File("src/Documentos/CAJERO-UT-COMPROBANTE DE PAGO.pdf");

    miModelo.abreArchivo(archivo);

}

}

//Al clic en la opción caja del menú

if (ae.getSource() == miCaja.jMenuItem_servCaja) {

    //Instancia de una nueva caja

    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

    controlador_servCaja miControladorCaja = new controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

    //Desde el controlador se hace visible la caja

    miControladorCaja.miCaja.setVisible(true);

    this.miCaja.dispose();

}

//Al clic en la opción CreditoUT del menú

if (ae.getSource() == miCaja.jMenuItem_servCreditoUT) {

    //Instancia de CreditoUT

    modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();

```

```

ServCreditoUT vistaCreditoUt = new ServCreditoUT();

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

controlador_creditoUT      miControladorCreditoUT      =      new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUt, miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCreditoUT.flagRol = this.flagRol;

miControladorCreditoUT.flagmatricula = this.flagmatricula;

miControladorCreditoUT.usrActivo = this.usrActivo;

miControladorCreditoUT.cargaEstadoCreditoUT();

//Desde el controlador se hace visible el buzón

miControladorCreditoUT.miCreditoUT.setVisible(true);

this.miCaja.dispose();
}

//Al clic en la opción Loboletos del menú

if (ae.getSource() == miCaja.jMenuItem_servLoboletos) {

    //Instancia de una nueva caja

    servCaja miCaja = new servCaja();

    modelo_servCaja modeloCaja = new modelo_servCaja();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

    controlador_servCaja      miControladorCaja      =      new      controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

    //Paso los datos de usuario

    miControladorCaja.flagRol = this.flagRol;

    miControladorCaja.flagmatricula = this.flagmatricula;

    miControladorCaja.usrActivo = this.usrActivo;

    miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

```

```

miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

//Desde el controlador se hace visible la caja
miControladorCaja.miCaja.setVisible(true);

this.miCaja.dispose();
}

//Al clic en la opción Evaluación del menú
if (ae.getSource() == miCaja.jMenuItem_servEvaluacion) {

    //Instancia de buzón
    modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

    ServEvaluacion vistaEvaluacion = new ServEvaluacion();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
    controlador_evaluacion miControladorEvaluacion = new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

    //Paso los datos de usuario
    miControladorEvaluacion.flagRol = this.flagRol;

    miControladorEvaluacion.flagmatricula = this.flagmatricula;

    miControladorEvaluacion.usrActivo = this.usrActivo;

    //Desde el controlador se hace visible el buzón
    miControladorEvaluacion.miEvaluacion.setVisible(true);

    this.miCaja.dispose();
}

//Al clic en la opción Buzón del menú
if (ae.getSource() == miCaja.jMenuItem_servBuzon) {

    //Instancia de buzón
    modelo_buzon modeloBuzon = new modelo_buzon();

    ServBuzon vistaBuzon = new ServBuzon();

```

```

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
        controlador_buzon  miControladorBuzon  =  new  controlador_buzon(modeloBuzon,
vistaBuzon, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorBuzon.flagRol = this.flagRol;

        miControladorBuzon.flagmatricula = this.flagmatricula;

        miControladorBuzon.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón

        miControladorBuzon.miBuzon.setVisible(true);

        this.miCaja.dispose();
    }

    //Si se da clic en el menú a la opción Calificaciones
    if (ae.getSource() == miCaja.jMenuItem_consCalificaciones) {

        File archivo = new File(miModelo.abreCalif(flagmatricula));

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menú a la opción Horario

    if (ae.getSource() == miCaja.jMenuItem_consHorario) {

        miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

        //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
dic-2019/TICASI-7-E-9.pdf");

    }

    //Si se da clic en el menú a la opción Estado de cuenta

    if (ae.getSource() == miCaja.jMenuItem_consEdoCuenta) {

        File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));

        miModelo.abreArchivo(archivo);
    }

```

```

    }

    //Si se da clic en el menu a la opción de ver mapa
    if (ae.getSource() == miCaja.jMenuItem_ConstMapa) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/mapaUT.jpg");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción de ver calendario
    if (ae.getSource() == miCaja.jMenuItem_calendario) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/calendario.pdf");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción becas
    if (ae.getSource() == miCaja.jMenuItem_becas) {

        miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

    }

    //Al clic en la opción Constancias del menú
    if (ae.getSource() == miCaja.jMenuItem_constancias) {

        //Instancia de buzón

        modelo_constancia modeloConstancia = new modelo_constancia();

        TramConstancia vistaConstancia = new TramConstancia();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
    }

```

```

        controlador_constancia        miControladorConstancia        =        new
controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorConstancia.flagRol = this.flagRol;

        miControladorConstancia.flagmatricula = this.flagmatricula;

        miControladorConstancia.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón

        miControladorConstancia.miConstancia.setVisible(true);

        this.miCaja.dispose();
    }

    //Si se da clic en el menú a la opción correo
    if (ae.getSource() == miCaja.jMenuItem_correo) {

        miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");

    }

    //Si se da clic en el menú a la opción navegar
    if (ae.getSource() == miCaja.jMenuItem_navega) {

        miModelo.abreNavegador("http://www.google.com.mx");

    }

    //Al clic en la opción Soporte del menú
    if (ae.getSource() == miCaja.jMenuItem_soporte) {

        //Instancia de buzón

        modelo_soporte modeloSoporte = new modelo_soporte();

        ServSoporte vistaSoporte = new ServSoporte();

        //se pasa la nueva caja como parametro para el constructor en el controlador de buzón

        controlador_soporte miControladorSoporte = new controlador_soporte(modeloSoporte,
vistaSoporte, miMenu, SalidaLogin);

```

```

//Paso los datos de usuario

miControladorSoporte.flagRol = this.flagRol;

miControladorSoporte.flagmatricula = this.flagmatricula;

miControladorSoporte.usrActivo = this.usrActivo;

miControladorSoporte.mensajeUsuarioSoporte();

//Desde el controlador se hace visible el buzón

miControladorSoporte.miSoporte.setVisible(true);

this.miCaja.dispose();

}

}

//Método que valida y cambia el estado de diversos componentes según el tipo de pago

public void radioPagoValida(){

    if(miCaja.rd_pagoCreditoUT.isSelected()) {

        miCaja.cb_vencimientoMes.setEnabled(false);

        miCaja.cb_vencimientoAnio.setEnabled(false);

        miCaja.tBtn_tecladoTarjeta.setEnabled(false);

        miCaja.tBtn_tecladoCVE.setEnabled(false);

    }

    else if (miCaja.rd_pagoTarjeta.isSelected()) {

        miCaja.cb_vencimientoMes.setEnabled(true);

        miCaja.cb_vencimientoAnio.setEnabled(true);

        miCaja.tBtn_tecladoTarjeta.setEnabled(true);

        miCaja.tBtn_tecladoCVE.setEnabled(true);

    }

}

}

```



```
@Override
```

```
public void windowOpened(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowClosing(WindowEvent we) {  
    miCaja.tBtn_tecladoTarjeta.setSelected(false);  
    this.miCaja.tBtn_tecladoCVE.setSelected(false);  
}
```

```
@Override
```

```
public void windowClosed(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowIconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowDeiconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowActivated(WindowEvent we) {
```

```

    }

    @Override

    public void windowDeactivated(WindowEvent we) {

    }

    @Override

    //Si se preciona en el menú cerrar sesión

    public void menuSelected(MenuEvent e) {

        int reply = JOptionPane.showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar
sesión", JOptionPane.YES_NO_CANCEL_OPTION);

        if(reply == JOptionPane.YES_OPTION) {

            //Carga el método que incia el login desde la clase main y cierra la ventana acual

            SalidaLogin.cargaLogin();

            this.miCaja.dispose();

        }

        else {

        }

    }

    @Override

    public void menuDeselected(MenuEvent e) {

    }

    @Override

    public void menuCanceled(MenuEvent e) {

```

```
}  
}
```

Controlador_soporte

```
package Controlador;  
  
import Modelo.*;  
  
import Vista.*;  
  
import java.awt.Desktop;  
  
import java.awt.event.ActionEvent;  
  
import java.awt.event.ActionListener;  
  
import java.io.File;  
  
import java.awt.event.WindowEvent;  
  
import java.awt.event.WindowListener;  
  
import java.io.File;  
  
import java.io.IOException;  
  
import java.util.logging.Level;  
  
import java.util.logging.Logger;  
  
import javax.swing.BorderFactory;  
  
import javax.swing.JOptionPane;  
  
import javax.swing.*.*;  
  
import javax.swing.event.*;  
  
import javax.swing.event.MenuEvent;  
  
import javax.swing.event.MenuListener;  
  
public class controlador_soporte implements ActionListener, WindowListener, MenuListener{  
  
    public modelo_soporte miModelo;  
  
    public ServSoporte miSoporte;
```

```

public principalMenu miMenu;

public mainInicial SalidaLogin;

public int flagRol;

public int flagmatricula;

public String usrActivo;


public controlador_soporte(modelo_soporte miModelo, ServSoporte miSoporte, principalMenu
miMenu, mainInicial SalidaLogin) {

    this.miModelo = miModelo;

    this.miSoporte = miSoporte;

    this.miMenu = miMenu;

    this.SalidaLogin = SalidaLogin;

    this.miSoporte.btn_enviarSoporte.addActionListener(this);

    this.miSoporte.jMenuItem_servCaja.addActionListener(this);

    this.miSoporte.jMenuItem_servCreditoUT.addActionListener(this);

    this.miSoporte.jMenuItem_servLoboletos.addActionListener(this);

    this.miSoporte.jMenuItem_servEvaluacion.addActionListener(this);

    this.miSoporte.jMenuItem_consCalificaciones.addActionListener(this);

    this.miSoporte.jMenuItem_consHorario.addActionListener(this);

    this.miSoporte.jMenuItem_consEdoCuenta.addActionListener(this);

    this.miSoporte.jMenuItem_ConsMapa.addActionListener(this);

    this.miSoporte.jMenuItem_calendario.addActionListener(this);

    this.miSoporte.jMenuItem_becas.addActionListener(this);

    this.miSoporte.jMenuItem_constancias.addActionListener(this);

    this.miSoporte.jMenuItem_correo.addActionListener(this);

    this.miSoporte.jMenuItem_navega.addActionListener(this);

```

```

this.miSoporte.jMenu_cerrarSesion.addMenuListener(this);

this.miSoporte.jMenuItem_servBuzon.addActionListener(this);

flagRol = 0;

flagmatricula = 0;

usrActivo = "";
}

@Override

//Método que recibe los eventos de acciones
public void actionPerformed(ActionEvent ae) {

    if (ae.getSource() == miSoporte.btn_enviarSoporte) {

        if(miSoporte.txtArea_soporte.getText().equals("")) {

            JOptionPane.showMessageDialog(null, "No se detecto mensaje. Escriba el mensaje que desea enviar.");

        }

        else {

            JOptionPane.showMessageDialog(null, "Tu mensaje ha sido enviado correctamente, espera pronto un correo con la respuesta.");

            miSoporte.txtArea_soporte.setText("");

        }

    }

    //Al clic en la opción caja del menú

    if (ae.getSource() == miSoporte.jMenuItem_servCaja) {

        //Instancia de una nueva caja

        servCaja miCaja = new servCaja();

        modelo_servCaja modeloCaja = new modelo_servCaja();

```

```

        //se pasa la nueva caja coómo parametro paara el constructor en el controlador de caja
        controlador_servCaja    miControladorCaja    =    new    controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

        //Desde el controlador se hace visible la caja
        miControladorCaja.miCaja.setVisible(true);

        this.miSoporte.dispose();
    }

    //Al clic en la opción CreditoUT del menú
    if (ae.getSource() == miSoporte.jMenuItem_servCreditoUT) {

        //Instancia de CreditoUT
        modelo_creditoUT modeloCreditoUT = new modelo_creditoUT();

        ServCreditoUT vistaCreditoUt = new ServCreditoUT();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
        controlador_creditoUT    miControladorCreditoUT    =    new
controlador_creditoUT(modeloCreditoUT, vistaCreditoUt, miMenu, SalidaLogin);

        //Paso los datos de usuario
        miControladorCreditoUT.flagRol = this.flagRol;

        miControladorCreditoUT.flagmatricula = this.flagmatricula;

        miControladorCreditoUT.usrActivo = this.usrActivo;

        miControladorCreditoUT.cargaEstadoCreditoUT();

        //Desde el controlador se hace visible el buzón
        miControladorCreditoUT.miCreditoUT.setVisible(true);

        this.miSoporte.dispose();
    }

    //Al clic en la opción Loboletos del menú
    if (ae.getSource() == miSoporte.jMenuItem_servLoboletos) {

```

```

//Instancia de una nueva caja

servCaja miCaja = new servCaja();

modelo_servCaja modeloCaja = new modelo_servCaja();

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de caja

controlador_servCaja    miControladorCaja    =    new    controlador_servCaja(miCaja,
modeloCaja,miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorCaja.flagRol = this.flagRol;

miControladorCaja.flagmatricula = this.flagmatricula;

miControladorCaja.usrActivo = this.usrActivo;

miControladorCaja.miCaja.cb_tipoPago.setSelectedIndex(4);

miControladorCaja.miCaja.cb_tipoPago.setEnabled(false);

//Desde el controlador se hace visible la caja

miControladorCaja.miCaja.setVisible(true);

this.miSoporte.dispose();

}

//Al clic en la opción Evaluación del menú

if (ae.getSource() == miSoporte.jMenuItem_servEvaluacion) {

//Instancia de buzón

modelo_evaluacion modeloEvaluacion = new modelo_evaluacion();

ServEvaluacion vistaEvaluacion = new ServEvaluacion();

//se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

controlador_evaluacion    miControladorEvaluacion    =    new
controlador_evaluacion(modeloEvaluacion, vistaEvaluacion, miMenu, SalidaLogin);

//Paso los datos de usuario

miControladorEvaluacion.flagRol = this.flagRol;

```

```

miControladorEvaluacion.flagmatricula = this.flagmatricula;

miControladorEvaluacion.usrActivo = this.usrActivo;

//Desde el controlador se hace visible el buzón
miControladorEvaluacion.miEvaluacion.setVisible(true);

this.miSoporte.dispose();
}

//Al clic en la opción Buzón del menú
if (ae.getSource() == miSoporte.jMenuItem_servBuzon) {

    //Instancia de buzón
    modelo_buzon modeloBuzon = new modelo_buzon();

    ServBuzon vistaBuzon = new ServBuzon();

    //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón
    controlador_buzon miControladorBuzon = new controlador_buzon(modeloBuzon,
vistaBuzon, miMenu, SalidaLogin);

    //Paso los datos de usuario
    miControladorBuzon.flagRol = this.flagRol;

    miControladorBuzon.flagmatricula = this.flagmatricula;

    miControladorBuzon.usrActivo = this.usrActivo;

    if(miControladorBuzon.flagRol == 1) {

        miControladorBuzon.miBuzon.jMenuItem_servCaja.setVisible(false);

        miControladorBuzon.miBuzon.jMenuItem_servCreditoUT.setVisible(false);

        miControladorBuzon.miBuzon.jMenuItem_servLoboletos.setVisible(false);

        miControladorBuzon.miBuzon.jMenuItem_servEvaluacion.setVisible(false);

        miControladorBuzon.miBuzon.jMenuItem_consCalificaciones.setVisible(false);

        miControladorBuzon.miBuzon.jMenuItem_consEdoCuenta.setVisible(false);

        miControladorBuzon.miBuzon.jMenuItem_tramites.setVisible(false);
    }
}

```



```

    }

    else {

    }

    //Desde el controlador se hace visible el buzón
    miControladorBuzon.miBuzon.setVisible(true);

    this.miSoporte.dispose();
}

//Si se da clic en el menú a la opción Calificaciones
if (ae.getSource() == miSoporte.jMenuItem_consCalificaciones) {
    File archivo = new File(miModelo.abreCalif(flagmatricula));
    miModelo.abreArchivo(archivo);
}

//Si se da clic en el menú a la opción Horario
if (ae.getSource() == miSoporte.jMenuItem_consHorario) {
    miModelo.abreNavegador(miModelo.abreHorario(flagmatricula));

    //miModelo.abreNavegador("http://www.utags.edu.mx/images/Archivos/ti/horarios/sep-
dic-2019/TICASI-7-E-9.pdf");
}

//Si se da clic en el menú a la opción Estado de cuenta
if (ae.getSource() == miSoporte.jMenuItem_consEdoCuenta) {
    File archivo = new File(miModelo.abreEdoCuenta(flagmatricula));
    miModelo.abreArchivo(archivo);
}

//Si se da clic en el menú a la opción de ver mapa
if (ae.getSource() == miSoporte.jMenuItem_ConsMapa) {

```

```

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/mapaUT.jpg");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción de ver calendario

    if (ae.getSource() == miSoporte.jMenuItem_calendario) {

        //Se mandará a traer un documento local anexo en la carpeta documentos en el propio
        proyecto

        File archivo = new File("src/Documentos/calendario.pdf");

        miModelo.abreArchivo(archivo);

    }

    //Si se da clic en el menu a la opción becas

    if (ae.getSource() == miSoporte.jMenuItem_becas) {

        miModelo.abreNavegador("http://www.utags.edu.mx/index.php/servicios/becas");

    }

    //Al clic en la opción Constancias del menú

    if (ae.getSource() == miSoporte.jMenuItem_constancias) {

        //Instancia de buzón

        modelo_constancia modeloConstancia = new modelo_constancia();

        TramConstancia vistaConstancia = new TramConstancia();

        //se pasa la nueva caja cómo parametro paara el constructor en el controlador de buzón

        controlador_constancia          miControladorConstancia          =          new
        controlador_constancia(modeloConstancia, vistaConstancia, miMenu, SalidaLogin);

        //Paso los datos de usuario

        miControladorConstancia.flagRol = this.flagRol;
    }

```

```

        miControladorConstancia.flagmatricula = this.flagmatricula;

        miControladorConstancia.usrActivo = this.usrActivo;

        //Desde el controlador se hace visible el buzón
        miControladorConstancia.miConstancia.setVisible(true);

        this.miSoporte.dispose();
    }

    //Si se da clic en el menú a la opción correo
    if (ae.getSource() == miSoporte.jMenuItem_correo) {

        miModelo.abreNavegador("https://outlook.office.com/owa/utags.edu.mx");

    }

    //Si se da clic en el menú a la opción navegar
    if (ae.getSource() == miSoporte.jMenuItem_navega) {

        miModelo.abreNavegador("http://www.google.com.mx");

    }

}

@Override

public void windowOpened(WindowEvent we) {

}

@Override

public void windowClosing(WindowEvent we) {

}

```

```
@Override
```

```
public void windowClosed(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowIconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowDeiconified(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowActivated(WindowEvent we) {  
}
```

```
@Override
```

```
public void windowDeactivated(WindowEvent we) {  
}
```

```
@Override
```

```
//Si se preciona en el menú cerrar sesión
```

```
public void menuSelected(MenuEvent e) {  
  
    int reply = JOptionPane.showConfirmDialog(null, "Seguro de que desea cerrar sesión", "Cerrar  
sesión", JOptionPane.YES_NO_CANCEL_OPTION);  
  
    if(reply == JOptionPane.YES_OPTION) {
```

```

        //Carga el método que incia el login desde la clase main y cierra la ventana acual
        SalidaLogin.cargaLogin();

        this.miSoporte.dispose();
    }

    else {

    }

}

@Override

public void menuDeselected(MenuEvent e) {

}

@Override

public void menuCanceled(MenuEvent e) {

}

public void mensajeUsuarioSoporte() {

    miSoporte.lbl_mensajeSoporte.setText("Hola "+usrActivo+", ¿Tienes algún problema con CajeroUT?");

}

}

```