

Universidad Autónoma Metropolitana
Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería
Licenciatura en Ingeniería en Computación

Modalidad:
Proyecto Tecnológico

Sistema para el seguimiento de alumnos basado en Clasificador Bayesiano
Ingenuo

Trimestre 2017 Otoño

José Alonso Chavarría González
2133034903
al2133034903@alumnos.azc.uam.mx

Josué Figueroa González
Asesor
Departamento de Sistemas
jfgo@correo.azc.uam.mx

14 de diciembre de 2017

Declaratoria

Yo, Josué Figueroa González, *declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.*



Josué Figueroa González

Yo, José Alonso Chavarría González, *doy mi autorización a la Coordinación de Servicios de Información de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, para publicar el presente documento en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.*



José Alonso Chavarría González

Resumen

En el presente proyecto de integración se diseñó e implementó una aplicación que permite dar seguimiento a alumnos de nivel medio superior para reducir el riesgo de que abandonen sus estudios, basándose en un Clasificador Bayesiano Ingenuo para calcular la probabilidad de deserción a partir de diversos factores académicos y personales, los cuales se tomaron de una base de datos.

El programa resultante cuenta con una interfaz gráfica compuesta de distintas vistas, en la que se permite elegir la operación a realizar, ya sea la carga de los datos, el cálculo de la probabilidad o la visualización de estadísticas mediante gráficas. La primera pantalla permite cargar la información de los alumnos hacia la base de datos, a partir de archivos fuente específicos, para así tenerlos disponibles para consultar durante el cálculo.

En la segunda parte del programa se cuenta con una pantalla para escoger de la lista de atributos, ya sean académicos o personales, los que se toman en cuenta para el cálculo de la probabilidad. Además, desde aquí se permite desplegar otra ventana con recomendaciones para reducir el riesgo de abandono, mostrando las posibles mejoras de forma individual o en conjunto.

La última pantalla muestra gráficas con algunos de los atributos sobresalientes que influyen sobre el resultado según los datos recabados. Todo esto permite tener una mayor comprensión sobre la relación entre el desempeño de los alumnos y las diferentes dificultades que pueden afrontar ya sea en casa o por parte de la escuela.

Índice

ÍNDICE DE FIGURAS.....	1
INTRODUCCIÓN	2
ANTECEDENTES.....	3
JUSTIFICACIÓN.....	5
OBJETIVOS	6
MARCO TEÓRICO	7
DESARROLLO DEL PROYECTO	11
ESTRUCTURA DEL PROYECTO	11
INFORMACIÓN A UTILIZAR.....	12
CASOS DE USO	15
PRESENTACIÓN DE LAS PANTALLAS	18
RESULTADOS.....	23
ANÁLISIS Y DISCUSIÓN DE RESULTADOS	24
CONCLUSIONES.....	25
REFERENCIAS BIBLIOGRÁFICAS	26
APÉNDICES.....	27
CÓDIGO FUENTE DE LA CLASE CONEXION.....	27
CÓDIGO FUENTE DE LA CLASE CARGAR	29
CÓDIGO FUENTE DE LA CLASE SELECCIONAR	33
CÓDIGO FUENTE DE LA CLASE CLASIFICAR	36
CÓDIGO FUENTE DE LA CLASE RECOMENDAR.....	38
CÓDIGO FUENTE DE LA CLASE GRAFICAR.....	58
CÓDIGO FUENTE DE LA CLASE MAIN	60

Índice de Figuras

TABLA 1. MUESTRA EJEMPLO	9
TABLA 2. ALUMNO EJEMPLO	9
TABLA 3. ATRIBUTOS ACADÉMICOS	13
TABLA 4. ATRIBUTOS PERSONALES	15
FIGURA 1. COMPONENTES DEL SISTEMA.....	11
FIGURA 2. DIAGRAMA DE CASOS DE USO	15
FIGURA 3. PANTALLA DE CARGA DE DATOS.....	18
FIGURA 4. PANTALLA PRINCIPAL DEL CLASIFICADOR.....	19
FIGURA 5. PANTALLA ACTUALIZADA	20
FIGURA 6. RESULTADOS DEL CLASIFICADOR.....	20
FIGURA 7. PANTALLA DE RECOMENDACIONES.....	21
FIGURA 8. PANTALLA DE PRESENTACIÓN DE ESTADÍSTICAS	22

Introducción

La educación siempre ha sido, y será, un tema de extrema relevancia para el desarrollo del país y la calidad de vida de sus habitantes. A pesar de los esfuerzos para que la población tenga mayor acceso a una buena y mejor educación, aún hay muchos problemas presentes que impiden grandes avances en este ámbito.

La deserción escolar es resultado de numerosos factores, varios de los cuales son comunes entre grandes grupos de estudiantes, por lo que se requiere que sean ubicados de forma precisa para ser tratados y resueltos, para así evitar futuras ocurrencias que afecten la permanencia de los alumnos en las instituciones.

La Minería de Datos consiste en un conjunto de técnicas utilizadas en el análisis de información para su mejor comprensión, permitiendo incluso predecir comportamientos basados en datos históricos. A su vez, la Minería de Datos Educativa enfoca estos métodos de estudio a la información que surge del ámbito educativo para hacer descubrimientos y poder entender mejor a los alumnos y la forma en que aprenden.

En este proyecto, se aplica la Minería de Datos Educativa para entender los patrones que pueden llevar a que un alumno de nivel medio superior no termine sus estudios, y así poder dar un seguimiento para ayudarlo a finalizarlos. Con este fin se decidió utilizar un Clasificador Bayesiano Ingenuo, que permite predecir la probabilidad de llegar a ciertos posibles resultados y cuya implementación es bastante sencilla a pesar de que su nivel de eficiencia puede competir con la de métodos de minería mucho más avanzados. [1]

Antecedentes

Minería de Datos Educacional en la Identificación de Factores de Deserción Escolar a Nivel Medio Superior. [2]

El objetivo de este artículo es muy similar al trabajo elaborado en este proyecto, analiza las causas de deserción en nivel medio superior, pero utiliza un árbol de decisiones que genera un modelo para identificar los factores que más afectan a los alumnos. A diferencia de este proyecto que utiliza Clasificador Bayesiano Ingenuo para realizar esta tarea.

Deserción Escolar a Nivel Superior: Una Aproximación a un Problema Complejo. [3]

Este trabajo se enfoca en analizar el problema de la deserción escolar desde el punto de vista sociológico. Busca entender la relación que existe entre la deserción escolar y los diversos problemas que tienen los alumnos de Nivel Superior, falta de recursos económicos, reprobación, etc. A diferencia del proyecto que se realizó, se analiza el Nivel Superior y además no pretende encontrar una forma de predecir la probabilidad de deserción de los alumnos.

Aplicación de Distintas Técnicas de Minería de Datos para el Tratamiento de Información. [4]

Un trabajo enfocado en la corrección de datos ausentes o incoherentes en una base de datos, hace énfasis en la limpieza de los datos para tener mejor operatividad con ellos. Contiene diversas bases teóricas relacionadas con el presente proyecto. La diferencia radica en que no está orientado exclusivamente al ámbito educacional.

Lenguaje de Manipulación y Minería de Datos. [5]

La intención principal de este trabajo es la de crear un lenguaje para realizar minería de datos, utilizando dos algoritmos para probar su funcionamiento, uno de ellos siendo el bayesiano ingenuo. La aplicación del clasificador bayesiano ingenuo es donde comparte similitudes con el presente proyecto, pero aquí se aplica el clasificador en la Minería de Datos Educacional y no crea ningún método o lenguaje nuevo para su uso.

Educational Data Mining: A Review of the State of the Art. [6]

Este trabajo trata el tema de la Minería de Datos Educacional desde un amplio espectro, incluye desde su significado, los tipos de usuarios, los ambientes en donde se desarrolla y los datos que se proveen. No se enfoca en un estudio en específico, como es el caso de este proyecto en el nivel medio superior, sino que da ejemplos de los logros que ha tenido esta área de investigación.

Machine Learning Algorithms and Predictive Models for Undergraduate Student Retention. [7]

Un artículo muy apegado al tema de este proyecto, analiza la retención de alumnos de licenciatura usando algoritmos de aprendizaje automático para clasificar los datos de los estudiantes. Lo que se obtiene como resultado son los factores que influyen a los alumnos para terminar sus estudios, además de crear una red neuronal para predecir la retención. La diferencia está en el método de clasificación para obtener la probabilidad de deserción, además de que el enfoque es a nivel medio superior.

Justificación

El problema de la deserción escolar a niveles medio superior y superior es de suma relevancia para el futuro del país, la persistencia estudiantil ha ido reduciéndose en los últimos años por lo que se deben buscar formas para que estudiantes con situaciones desfavorables puedan seguir acudiendo a clases, recibiendo el apoyo que necesiten.

A través del uso de Minería de Datos en una base de datos con información recopilada de alumnos, tanto de quienes desertaron como de quienes concluyeron sus estudios, se analizan estos datos históricos para generar una forma de predecir el resultado más probable para nuevos alumnos según sus circunstancias particulares.

Hallar los factores más importantes que provocan que un alumno termine desertando de sus estudios y a la vez qué cambios se necesitaría realizar para que el alumno tome una ruta diferente y logre continuar estudiando es donde radica la importancia de este trabajo.

Objetivos

Objetivo General

Diseñar e implementar un sistema basado en un Clasificador Bayesiano Ingenuo que ayude al seguimiento de alumnos para reducir el riesgo de abandono de estudios.

Objetivos Específicos

- Elaborar un módulo para la limpieza de los datos de los alumnos, y cargarlos a una base de datos.
- Diseñar e implementar un módulo de selección de atributos a tomarse en cuenta para la clasificación.
- Elaborar un módulo con un clasificador bayesiano a partir de los atributos elegidos.
- Desarrollar un módulo de presentación de las estadísticas obtenidas de manera gráfica.
- Diseñar y desarrollar un módulo de seguimiento para mostrar la probabilidad de un alumno de abandonar sus estudios y los factores que se deben trabajar para evitarlo.

Marco Teórico

Minería de Datos

El termino Minería de Datos es relativamente moderno, está integrado de diversas técnicas de análisis de datos y extracción de modelos. Aunque toma como base distintas disciplinas, entre ellas algunas que son más tradicionales, se distingue por su orientación más hacia el fin que el medio. El ser capaces de extraer patrones, describir tendencias, predecir comportamientos y, en general, sacar partido a toda la información que está a nuestro alcance hoy en día, permite comprender y modelar de una manera más eficiente y precisa el contexto en el que se debe actuar y tomar decisiones. [8]

La Minería de Datos no surgió debido al desarrollo de nuevas tecnologías, sino en realidad por el surgimiento de nuevas necesidades, por la importancia que representa el valor de los datos almacenados por las instituciones, empresas, gobiernos y particulares. Los datos ahora son una materia prima explotable y la Minería de Datos es la etapa en la cual se extrae información útil y comprensible, previamente desconocida, a partir de estos. [1]

Los avances en la computación han ido facilitando cada vez más este proceso, permitiendo conducir análisis estadísticos mucho más complejos que los permitidos en el pasado, con métodos cada vez más eficaces y simples. Junto con esto se ha ido desarrollando la necesidad de poder visualizar los resultados de dichos análisis de forma más intuitiva, permitiendo así interpretar lo que sucede con los datos y sugerir posibles direcciones para estudios posteriores.

La presentación de resultados en el pasado se ha basado principalmente en la generación de estadísticas como medias, medianas, desviaciones y rangos, atractivas debido a su naturaleza objetiva. Sin embargo, esto suele ocultar otras características interesantes que pueden ser vistas por investigadores al analizarla desde puntos de vista diferentes. [9]

Visualización

Las técnicas de visualización de datos tienen dos objetivos: aprovechar la capacidad humana para encontrar patrones, anomalías y tendencias a partir de imágenes y, ayudar al usuario a comprender más rápidamente los patrones descubiertos automáticamente. Estos objetivos marcan dos momentos diferentes en el uso de la visualización.

La visualización previa, para entender mejor los datos y sugerir patrones o las herramientas a utilizar. La visualización posterior, muestra los patrones para entenderlos mejor y así poder validarlos y mostrar los resultados de la Minería de Datos.

En este proyecto se plantea el simple uso de visualización previa, mediante gráficas de barras, para poder detectar algunos patrones en los datos que respalden los resultados obtenidos al utilizar el clasificador.

Teorema de Bayes

El Teorema de Bayes, en teoría de la probabilidad, es una regla para realizar inferencias. Es decir, permite calcular la probabilidad de un evento A dado el conocimiento previo de un evento B.

Nos permite pasar de la probabilidad *a priori* $P(\text{suceso})$ a la *a posteriori* $P(\text{suceso}|\text{observaciones})$. La probabilidad a priori se puede ver como la inicial, fijada sin tener más conocimiento. La a posteriori es la que se obtiene tras conocer más información, es decir, refinando el conocimiento. [8]

El Teorema de Bayes viene representado por la expresión:

$$P(h|O) = \frac{P(O|h) \cdot P(h)}{P(O)}$$

Dónde:

$P(h)$ = probabilidad a priori de la hipótesis

$P(O)$ = probabilidad a priori de las observaciones

$P(O|h)$ = probabilidad de O dado H

$P(h|O)$ = probabilidad de h dado O

En Minería de datos, el uso de métodos bayesianos es una forma práctica para realizar inferencias a partir de los datos, induciendo modelos probabilísticos que después se utilizan para analizar nuevos valores.

Clasificador Bayesiano Ingenuo

Basado en el Teorema de Bayes, es una técnica de clasificación que, dado un conjunto de objetos, cada uno de los cuales pertenece a una clase y además tiene un vector de variables conocidos, permite asignar objetos futuros a una clase según el vector de variables que los describen. Es uno de los métodos de clasificación más fáciles de implementar y de interpretar, y que además tiene un alto desempeño, permitiendo aplicarlo a conjuntos de datos de gran tamaño. [10]

La base principal del Bayesiano Ingenuo es la suposición de que todos los atributos son independientes uno de otro dentro de una misma clase, por lo que la fórmula para el cálculo de la probabilidad se modifica quitando el denominador, debido a que este sería el mismo par todas las clases. Lo que nos da como resultado:

$$P(h|O) = P(O|h) \cdot P(h)$$

En este proyecto, el objetivo es predecir la probabilidad de que un alumno abandone o no sus estudios en base a los datos que se proporcionan, comparándolos con la información previamente almacenada en la base de datos de alumnos obtenida de la Encuesta Nacional de Deserción en la Educación Media Superior [11].

Para ejemplificar el funcionamiento del clasificador se toma una muestra de diez alumnos con información de tipo académico, en la que el número de factores que se toman para el cálculo del abandono a cinco, con lo que se obtiene la Tabla 1.

Promedio asistencia	Promedio último año	Porcentaje aprobación	Beca	Problemas compañeros	Deserto
BAJO	BAJO	BAJO	SI	NO	SI
BAJO	BAJO	ALTO	NO	NO	SI
MEDIO	MEDIO	MEDIO	NO	NO	SI
ALTO	ALTO	ALTO	SI	NO	NO
MEDIO	ALTO	ALTO	SI	SI	NO
ALTO	MEDIO	ALTO	NO	NO	NO
ALTO	BAJO	MEDIO	NO	NO	NO
MEDIO	MEDIO	ALTO	SI	NO	NO
MEDIO	MEDIO	MEDIO	NO	SI	NO
ALTO	ALTO	MEDIO	SI	NO	NO

Tabla 1. Muestra Ejemplo

Suponiendo que se tiene un nuevo alumno con las siguientes características:

Promedio asistencia	Promedio último año	Porcentaje aprobación	Beca	Problemas compañeros
MEDIO	MEDIO	ALTO	SI	NO

Tabla 2. Alumno Ejemplo

Las fórmulas que se utilizan para calcular la probabilidad de desertar y no desertar son:

Probabilidad de desertar

$$P(si|E) = \frac{P(asistencia|si)P(promedio|si)P(aprobacion|si)P(beca|si)P(problemas|si) * P(si)}{P(E)}$$

Probabilidad de no desertar

$$P(\text{no}|E) = \frac{P(\text{asistencia}|\text{no})P(\text{promedio}|\text{no})P(\text{aprobacion}|\text{no})P(\text{beca}|\text{no})P(\text{problemas}|\text{no}) * P(\text{no})}{P(E)}$$

Dónde:

- $P(\text{si})$ = cantidad de alumnos que DESERTARON (3/10)
- $P(\text{no})$ = cantidad de alumnos que NO DESERTARON (7/10)
- $P(\text{asistencia}|\text{si})$ = alumnos que DESERTARON con prom. de asistencia ALTO (1/3)
- $P(\text{promedio}|\text{si})$ = alumnos que DESERTARON con prom. del último año MEDIO (1/3)
- $P(\text{aprobacion}|\text{si})$ = alumnos que DESERTARON con prom. de aprobación ALTO (1/3)
- $P(\text{beca}|\text{si})$ = alumnos que DESERTARON que SI tienen beca (1/3)
- $P(\text{problemas}|\text{si})$ = alumnos que DESERTARON que NO tienen problemas con compañeros (3/3)
- $P(\text{asistencia}|\text{no})$ = alumnos que NO DESERTARON con prom. de asistencia MEDIO (3/7)
- $P(\text{promedio}|\text{no})$ = alumnos que NO DESERTARON con prom. del último año MEDIO (3/7)
- $P(\text{aprobacion}|\text{no})$ = alumnos que NO DESERTARON con prom. de aprobación ALTO (4/7)
- $P(\text{beca}|\text{no})$ = alumnos que NO DESERTARON que SI tienen beca (4/7)
- $P(\text{problemas}|\text{no})$ = alumnos que NO DESERTARON que NO tienen problemas con compañeros (5/7)

De nuevo, $P(E)$ se va a descartar por el principio del Bayesiano Ingenuo.

Por lo tanto:

Probabilidad de desertar:

$$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{3} \cdot \frac{3}{3} \cdot \frac{3}{10} = 0.0037$$

Probabilidad de no desertar:

$$\frac{3}{7} \cdot \frac{3}{7} \cdot \frac{4}{7} \cdot \frac{4}{7} \cdot \frac{5}{7} \cdot \frac{7}{10} = 0.0299$$

Como porcentaje:

Deserto:

$$\frac{p(\text{si})}{p(\text{si}) + p(\text{no})} * 100 = 11.012\%$$

No deserto:

$$\frac{p(\text{no})}{p(\text{si}) + p(\text{no})} * 100 = 88.988\%$$

Desarrollo del Proyecto

Estructura del Proyecto

El proyecto está desarrollado en lenguaje Java, la interfaz gráfica se diseñó utilizando las bibliotecas de Java Swing apoyándose en el constructor de interfaces de Netbeans 8.2. Para la base de datos se utiliza JDBC con el servidor MySQL.

El funcionamiento del sistema está basado en cinco módulos que cumplen distintas funciones. La forma en que interactúan estos módulos se presenta en la Figura 1. Los módulos son:

- Módulo de Carga
- Módulo de Selección de Atributos
- Módulo de Clasificación
- Módulo de Presentación de Estadísticas
- Módulo de Seguimiento

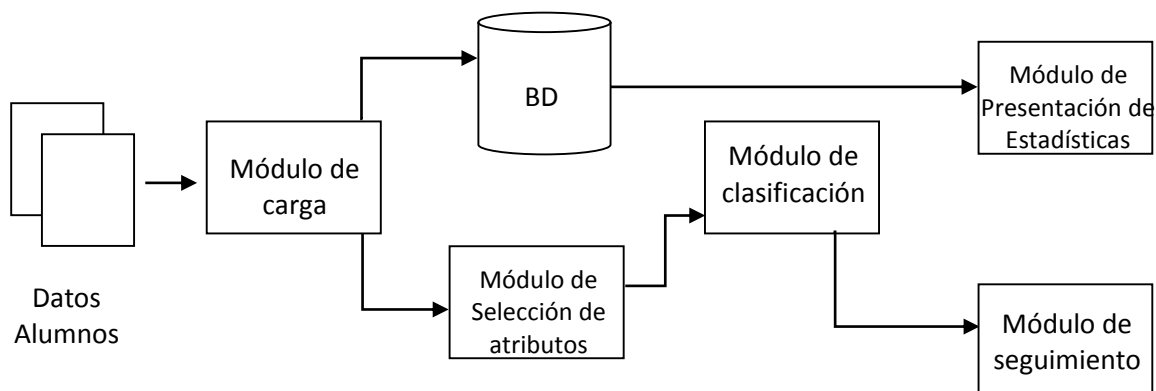


Figura 1. Componentes del Sistema

Módulo de Carga

Se encarga de tomar los datos de los alumnos que se tienen en un archivo base, y los coloca en una tabla de la Base de Datos.

Módulo de Selección de Atributos

Selecciona los atributos almacenados en la base de datos que se utilizarán en el clasificador bayesiano. Los atributos se muestran en pantalla y el usuario selecciona los que desea utilizar para el proceso de clasificación de un nuevo alumno no clasificado previamente.

Módulo de Clasificación

Aplica el clasificador bayesiano ingenuo sobre los datos que se escogieron en el módulo anterior. Es la parte central del sistema, devuelve la probabilidad de que el alumno finalice o no sus estudios.

Módulo de Presentación de Estadísticas

Presenta estadísticas de los datos de alumnos almacenados en la base de datos. La forma de visualizarlas es a través de gráficas de barras de algunos de los atributos más importantes en el cálculo de la probabilidad de desertar.

Módulo de Seguimiento

Muestra sugerencias sobre qué atributos se pueden mejorar y a que valores llegar para aumentar la probabilidad de que el alumno no abandone sus estudios.

Información a Utilizar

La información de los alumnos se obtiene de la Encuesta Nacional de Deserción en la Educación Media Superior [2]. Se toman en cuenta atributos Académicos y Personales, de los cuales los criterios para el clasificador se presentan en las tablas 3 y 4.

Criterio	Categorías	Incluye los casos
Género	HOMBRE MUJER	
Edad de ingreso	NORMAL ATRASO	14 – 16 años Más de 16 años
Examen de Admisión	ALTO MEDIO BAJO	
Promedio de secundaria	ALTO MEDIO BAJO	Más de 8.7 Más de 7.6 – 8.7 Menos de 7.5
Tipo de Secundaria	PUBLICA PRIVADA AMBAS	

Cambios de Escuela	DOS_MAS UNA NINGUNA	
Tiempo de Traslado	ALTO MEDIO BAJO	2 horas o más Media hora a 2 horas Menos de media hora
Razón de Ingreso	ACADEMICA ASIGNACION ECONOMICA PERSONAL	
Promedio de Asistencia a Clases	ALTO MEDIO BAJO	Siempre o casi siempre En ocasiones faltaba Faltaba mucho a clases
Promedio del Último Año	ALTO MEDIO BAJO	8.5 – 10 7.5 – 8.5 6 – 7.5
Índice de Aprobación de Materias	ALTO MEDIO BAJO	Siempre aprobaba En ocasiones reprobaba Casi siempre reprobaba
Aprovechamiento	ALTO MEDIO BAJO	
Beca	SI NO	
Apoyo de la Escuela	SI NO	
Problemas con Compañeros	SI NO	
Problemas Escolares	SI NO	

Tabla 3. Atributos Académicos

Criterio	Categorías	Incluye los casos
Género	HOMBRE MUJER	
Vivía Con	AMIGOS DIRECTOS PAREJA SOLO OTROS	
Apoyo de Familiares	SI NO	
Amigos Abandonaron la escuela	SI NO	
Hermanos Abandonaron la Escuela	SI NO	
Consumo de Cigarro	ALTO MEDIO BAJO NO	
Consumo de Alcohol	ALTO MEDIO BAJO NO	
Consumo de Drogas	ALTO MEDIO BAJO NO	Incluye cualquier tipo de droga
Problemas de Acoso Escolar	SI NO	Incluye acoso escolar y discriminación
Problemas en Casa	SI NO	
Familia Propia	MATRIMONIO EMBARAZO AMBOS NINGUNO	

Falta de Dinero en Casa	SI NO	
Baja Autoestima	SI NO	
Enfermedad Grave o Fallecimiento de un Familiar	SI NO	
Poco Gusto por Estudios	SI NO	Incluye problemas de disciplina y considerar poco importante al estudio
Otros Familiares Estudiaron	SI NO	Considera el caso en donde se le dio prioridad al estudio de otros familiares
Padres finalizaron nivel medio superior	PADRE MADRE AMBOS NINGUNO	

Tabla 4. Atributos Personales

Casos de Uso

Para una mejor comprensión de la funcionalidad del sistema se presentan a continuación los casos de uso.

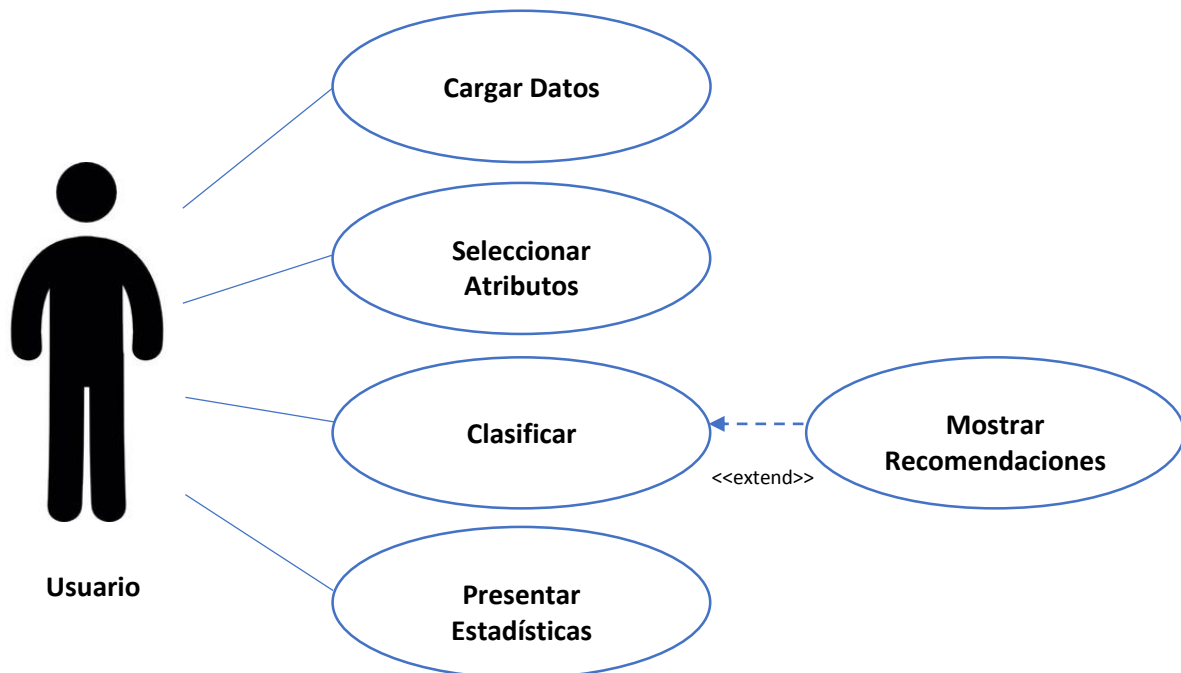


Figura 2. Diagrama de Casos de Uso

Documentación de caso de uso **Cargar Datos**

Caso de Uso	Cargar Datos
Actor	Usuario
Tipo	Básico
Propósito	Cargar los datos de los alumnos a la base de datos
Pre Condiciones	El programa deberá contar con la información del servidor de base de datos
Post Condiciones	Se habrá creado y llenado la tabla del tipo de datos seleccionado
Escenario de Éxito	<ol style="list-style-type: none"> 1. El sistema se conecta al servidor de base de datos 2. El sistema permite explorar los directorios para seleccionar los archivos de encabezados y datos 3. Se crea la tabla del tipo de atributos seleccionado utilizando el archivo de encabezados para las columnas 4. Se llena dicha tabla con los datos de los alumnos
Extensiones	1.1 En caso de que no se pueda conectar al servidor de base de datos se indicara un mensaje en pantalla indicando el error
Pantallas	Pantalla con campos para seleccionar la ruta de los archivos de encabezados y datos, junto con dos botones, uno para crear encabezados de la tabla y otro para llenar la tabla

Documentación de caso de uso **Seleccionar Datos**

Caso de Uso	Seleccionar Datos
Actor	Usuario
Tipo	Básico
Propósito	Seleccionar los atributos a utilizar por el clasificador
Pre Condiciones	Contar con los datos de los alumnos en la base de datos
Post Condiciones	Se tendrán los atributos a enviar para clasificar un nuevo alumno
Escenario de Éxito	<ol style="list-style-type: none"> 1. El sistema permite elegir entre datos académicos o personales 2. Se presentan los diversos atributos del tipo seleccionado con una lista para escoger su valor
Extensiones	1.1 En caso de que no exista la tabla del tipo de atributo seleccionado se mostrará un mensaje de error en pantalla
Pantallas	Pantalla la opción de seleccionar el tipo de atributos de una lista desplegable, y los atributos con sus valores en listas desplegables del tipo de atributo seleccionado. Un botón para limpiar la selección, un botón para realizar la clasificación, y un último para mostrar recomendaciones. Dos campos de texto para mostrar los resultados de la clasificación.

Documentación de caso de uso **Clasificar**

Caso de Uso	Clasificar
Actor	Usuario
Tipo	Básico
Propósito	Obtener la probabilidad de que un nuevo alumno finalice o no sus estudios
Pre Condiciones	Se habrán seleccionado los valores de los atributos de un nuevo alumno
Post Condiciones	Se mostrará la probabilidad de desertar y la de finalizar del alumno
Escenario de Éxito	1. El sistema calcula la probabilidad de deserción del alumno y la muestra en pantalla
Extensiones	1.1 Si no se selecciona ningún valor a clasificar, simplemente se muestra el porcentaje de alumnos que desertaron y que finalizaron
Pantallas	Pantalla del caso Seleccionar Datos, mostrando en los campos de texto la probabilidad de finalizar y de desertar

Documentación de caso de uso **Mostrar Recomendaciones**

Caso de Uso	Mostrar Recomendaciones
Actor	Usuario
Tipo	Extensión
Propósito	Mostrar posibles cambios para mejorar la probabilidad de finalizar estudios
Pre Condiciones	Se habrá hecho una clasificación de un alumno nuevo
Post Condiciones	Se mostrarán diversas sugerencias para reducir la probabilidad de desertar
Escenario de Éxito	1. El sistema despliega una nueva pantalla con las recomendaciones
Extensiones	1.1 En caso de no haber realizado ninguna clasificación, el botón para mostrar recomendaciones permanecerá deshabilitado
Pantallas	Pantalla con 7 posibles cambios para reducir la probabilidad de desertar. Se mostrarán nuevos porcentajes por cada cambio, además de un porcentaje total si se logran cumplir todos los cambios

Documentación de caso de uso **Presentar Estadísticas**

Caso de Uso	Presentar Estadísticas
Actor	Usuario
Tipo	Básico
Propósito	Permitir la visualización de estadísticas sobre los datos almacenados en la base de datos

Pre Condiciones	Contar con los datos de los alumnos en la base de datos
Post Condiciones	Se mostrarán diversas gráficas de barras con información relevante
Escenario de Éxito	<ol style="list-style-type: none"> 1. El sistema permite elegir el tipo de atributos para presentar, así como de que clase (desertó o finalizo) 2. El sistema mostrará una gráfica de barras apiladas con el tipo de información seleccionada
Extensiones	1.1 En caso de no existir datos en la base de datos se mostrará en pantalla un mensaje de error
Pantallas	Pantalla con listas desplegables para seleccionar el tipo de atributos y la clase, junto con un botón que mostrará la gráfica de la información seleccionada

Presentación de las Pantallas

En la Figura 3 se muestra la primera pantalla que aparece ante el usuario, desde está se pueden utilizar los botones de Seleccionar para navegar entre los directorios del sistema para seleccionar los archivos de encabezados y de datos para la base de datos. La tabla se crea con el botón de Crear Encabezados y esta se llena con el de Llenar Tabla.

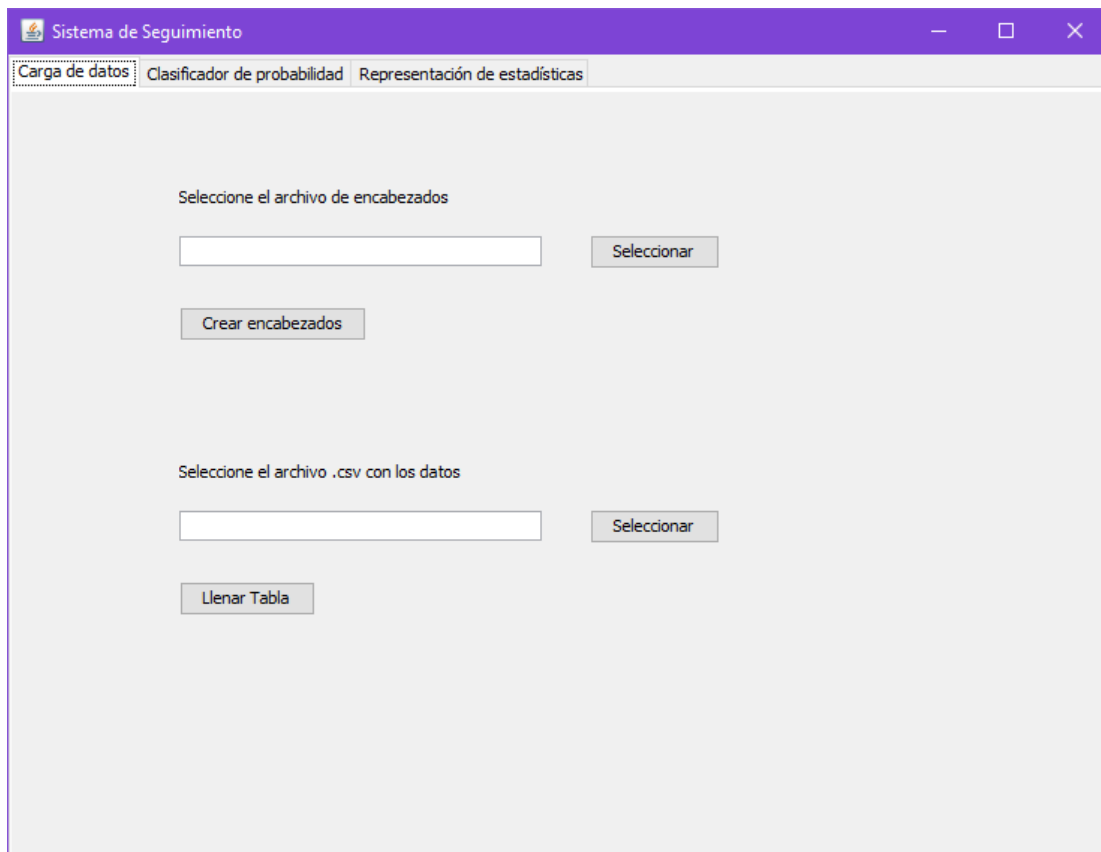


Figura 3. Pantalla de Carga de Datos

Después de haber cargado los datos en la pantalla anterior, o al ya tener una base de datos precargada, se muestra en la Figura 4 la siguiente pantalla para seleccionar los atributos y clasificar un nuevo alumno. Lo primero a realizar en esta pantalla es seleccionar el tipo de atributos que se considerarán (académicos o personales), lo que actualizará las etiquetas para mostrar los atributos con los que cuenta el tipo de clase seleccionado (Figura 5).

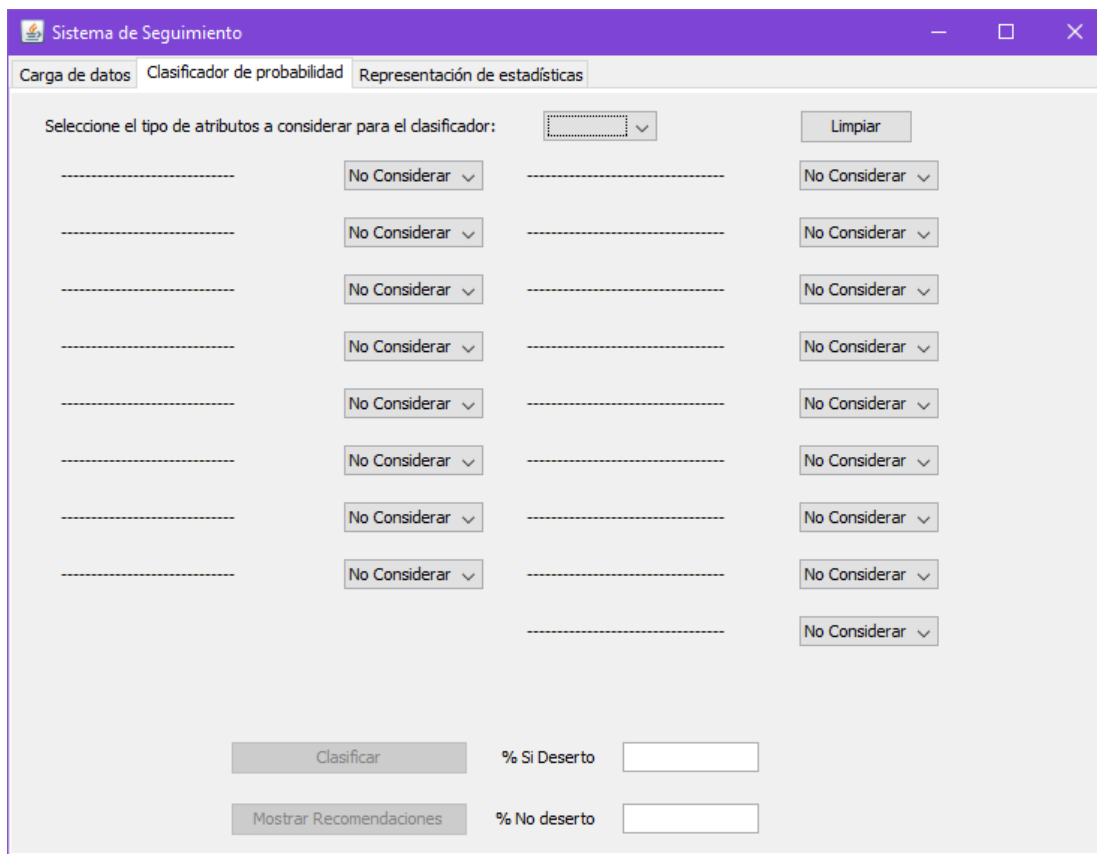


Figura 4. Pantalla Principal del Clasificador

Junto con la actualización de las etiquetas, las listas desplegables añaden los valores que cada uno de los atributos puede manejar, permitiendo al usuario elegir uno para cada atributo o dejarlo en la opción de No Considerar. Es en esta etapa en la que se presenta un nuevo alumno para ser clasificado, tomando como sus datos los valores seleccionados de las listas. Si no se seleccionará ningún dato, el clasificador solo regresará el porcentaje de alumnos que desertaron y el de los que finalizaron.

Figura 5. Pantalla Actualizada

Al oprimir el botón de clasificar, la probabilidad de desertar y no desertar se muestra en los campos de texto que aparecen en la parte inferior de la pantalla (Figura 6).

Figura 6. Resultados del Clasificador

Si se oprime el botón de Mostrar Recomendaciones, una pantalla nueva (Figura 7) se abrirá mostrando distintas opciones con las que se puede buscar trabajar para mejorar la probabilidad de no desertar. De las siete opciones que se presentan, cada una tiene consigo el nuevo porcentaje para la deserción y finalización de estudios si se consigue alcanzar el valor, además de un valor total para finalizar si se siguen todas las recomendaciones.

		Nuevo % Desertar	Nuevo % Finalizar
Promedio de asistencia	ALTO	56.048508	43.951492
Porcentaje de aprobación	ALTO	24.234768	75.765236
Aprovechamiento	ALTO	27.655363	72.344635
Beca	SI	64.428345	35.57165
Apoyo de la escuela	SI	60.697903	39.302097
Problemas con compañeros	NO	62.12862	37.87138
Problemas escolares	NO	64.3524	35.647594

Si sigue todas las recomendaciones anteriores, la probabilidad de finalizar cambia de 35.57165% a 96.47033%

Figura 7. Pantalla de Recomendaciones

Finalmente, en la última pantalla del proyecto (Figura 8.) se presenta la opción de mostrar gráficas con estadísticas obtenidas de la base de datos de los alumnos que se cargó en la primera pantalla. Dos listas desplegables permiten elegir que clase y de que tipos de atributos (de nuevo académicos o personales) serán los datos que se utilizarán para realizar las gráficas. Estas serán gráficas de barras apiladas con 5 de los atributos más influyentes a la hora de clasificar.

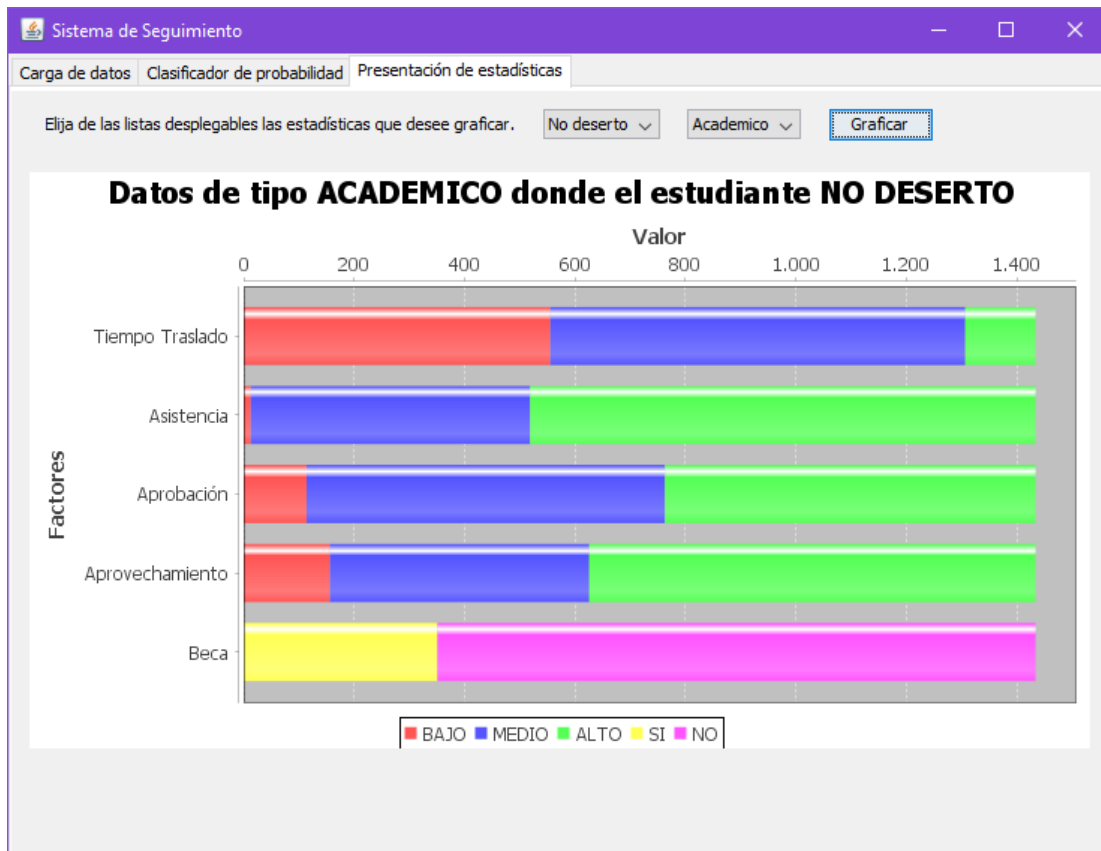


Figura 8. Pantalla de Presentación de Estadísticas

Resultados

Se logró cumplir con los objetivos específicos del proyecto, implementando los cinco módulos que se plantearon al inicio.

Módulo de Carga: Almacena satisfactoriamente la información de los alumnos en la base de datos, crea una tabla por cada tipo de atributo a utilizar en el sistema.

Módulo de Selección de Atributos: Permite seleccionar un nuevo conjunto de valores del atributo seleccionado para representar un nuevo alumno, permitiendo utilizar el clasificador sobre los datos.

Módulo de Clasificación: Aplica el Clasificador Bayesiano Ingenuo sobre los datos de un nuevo alumno, devolviendo su probabilidad de desertar o finalizar como un porcentaje.

Módulo de Presentación de Estadísticas: Muestra, mediante el uso de gráficas de barras apiladas, estadísticas de los alumnos almacenados en la base de datos, dependiendo de la clase y tipo de atributo seleccionados.

Módulo de Seguimiento: Presenta nuevos valores para algunos atributos como sugerencias para la mejora de la probabilidad de finalizar los estudios del nuevo alumno que se acaba de clasificar.

Análisis y Discusión de Resultados

Después de realizar diversas pruebas aprovechando la funcionalidad del sistema para la clasificación y muestra de recomendaciones, se ha podido llegar a algunos resultados que permiten comprender más la forma en que influye cada atributo en el cálculo de la probabilidad de desertar. Analizando en conjunto los números arrojados por el clasificador y las recomendaciones, sobresalen 5 atributos de tipo académico y 2 de personales que influyen fuertemente sobre el futuro del alumno.

Para los atributos de tipo académico se pudo observar que el Promedio de Asistencia, Porcentaje de Aprobación y el Aprovechamiento eran los campos que más influían en el porcentaje, individualmente al seleccionar entre sus valores el cambio entre desertar o no podía dar grandes saltos. También se observó que el Examen de Admisión y el Promedio del Último Año tenían una influencia similar a los tres atributos anteriores. Sin embargo, estos últimos atributos son variables que ya no se pueden cambiar a futuro, por lo que no habría razón para tomarlos en cuenta como recomendaciones. No obstante, su importancia no debe descartarse ya que podrían usarse para algún otro tipo de estudio previo al ingreso a la Educación Media Superior.

En el caso de los atributos personales, se notó que un atributo en específico influía más que la mayoría. El Poco Gusto por Estudiar tenía fluctuaciones mucho más grandes que la mayoría de los atributos. Y al igual que en académicos, existe un atributo que no se considera como recomendación directa, la Familia Propia que generaba cambios similares. Los otros atributos. Los demás valores no generaban cambios tan bruscos, pero caben resaltar como importantes el Consumo de Cigarro, Alcohol, Drogas, Problemas en Casa y Baja Autoestima.

Conclusiones

A partir de los resultados se consigue notar el impacto que podría tener la aplicación de este proyecto en la Educación Media Superior, ya que como se tenía planteado en el objetivo, se logró crear un sistema que permite dar seguimiento a los alumnos para reducir el riesgo de que abandonen sus estudios a partir de la identificación de los factores que más influyen en este aspecto, permitiendo trabajar sobre ellos basándose en las recomendaciones que arroja.

Hace posible identificar alumnos que están en un posible riesgo de abandono o, inclusive, aquellos que no se encuentran en peligro eminente pero que si no se les presta atención podrían llegar a estarlo. De esta forma se podría ir mejorando poco a poco el porcentaje de alumnos que finalizan sus estudios, dado que se les da atención más personalizada que atiende a sus necesidades particulares.

Todo esto se logra gracias a la correcta aplicación del Clasificador Bayesiano Ingenuo para poder calcular la probabilidad de deserción de un alumno, apoyándose en una vasta Base de Datos, aumenta la exactitud de los cálculos y reduce la posibilidad de errores, garantizando la integridad del sistema.

Referencias Bibliográficas

- [1] Witten, I. y Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations* (1a ed.). San Francisco, USA: Morgan Kaufmann,
- [2] J. Figueroa y S. B. González, “Minería de Datos en la Identificación de Factores de Deserción Escolar a Nivel Medio Superior”, Libro Electrónico de las Memorias “*Avances en las Tecnologías de la Información*”, ANIEI (2016)
- [3] M. Guevara Orive, “Deserción Escolar a Nivel Superior: Una Aproximación a un Problema Complejo”, Proyecto terminal, División de Ciencias Sociales y Humanidades, Universidad Autónoma Metropolitana Azcapotzalco, 1999
- [4] N. Guzmán González, “Aplicación de Distintas Técnicas de Minería de Datos para el Tratamiento de Información”, Proyecto terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, 2011
- [5] C. A. Díaz Jiménez, “Lenguaje de Manipulación y Minería de Datos”, Proyecto terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, 2011
- [6] C. Romero y S. Ventura, “Educational Data Mining: A Review of the State of the Art”, *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and reviews*, Vol. 40, Núm. 6, 601-618, 2010
- [7] J. Jia, M. Mareboyana, “Machine Learning Algorithms and Predictive Models for Undergraduate Student Retention”, *Proceedings of the World Congress on Engineering and Computer Science 2013*, Vol 1, 2013. [En línea]. Disponible: http://www.iaeng.org/publication/WCECS2013/WCECS2013_pp222-227.pdf
- [8] Hernández Orallo, J., Ramírez Quintana, M.J., Ferri Ramírez, C. (2004). *Introducción a la Minería de Datos* (1a ed.). Madrid, España: Pearson Prentice Hall
- [9] B. Shneiderman, “Inventing Discovery Tools: Combining Information Visualization with Data Mining”, *Information Visualization (2002) 1*, 5-12, 2002
- [10] Hand, D.J. (2009), Naïve Bayes. En Wu, X. y Kumar, V. (Eds.) *The Top Ten Algorithms in Data Mining* (pp. 163-178) Boca Raton, USA: CRC Press
- [11] Subsecretaría de Educación Media Superior, “Encuesta Nacional de Deserción en la Educación Media Superior”. [En línea]. Disponible: http://www.sems.gob.mx/es_mx/sems/encuesta_nacional_desercion_ems, 2017

Apéndices

Código Fuente de la Clase Conexion

```
package main;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
/**
 * Esta clase se encarga de manejar la conexion al servidor de MySQL,
 * tiene dos metodos, uno para la conexion el inicio del programa
 * y otro para operaciones de consulta durante la ejecución.
 */
public class Conexion {
    /**
     * Configuracion de los datos de la BD
     */
    private final String usuario = "root";
    private final String pass = "root";
    private final String host = "localhost";
    private final String nombre_BD = "alumnos_ms";
    private Statement st;

    private Connection con = null;

    public Conexion() {
    }
    /**
     * Metodo que devuelve una conexion para realizar consultas a la BD.
     */
    public Connection getConexionMYSQL() {
        try
        {
            Class.forName("com.mysql.jdbc.Driver").newInstance( );
            String servidor = "jdbc:mysql://" + host + "/" + nombre_BD;
            con = DriverManager.getConnection(servidor, usuario, pass);

            return con;
        } catch (Exception e) {
            e.printStackTrace();
            return con;
        }
    }

    /**
     * Primer metodo para conectarse al servidor de base de datos,
     * si no existe la base de datos la crea.
     */
    public Connection getConexionMYSQL2 ()
    {
        try
        {
            Class.forName("com.mysql.jdbc.Driver").newInstance( );
```

```

String servidor = "jdbc:mysql://" + host + "/";
con = DriverManager.getConnection(servidor, usuario, pass);
st = con.createStatement();
st.executeUpdate("CREATE DATABASE IF NOT EXISTS " + nombre_BD);
return con;
}catch (Exception e) {
    e.printStackTrace();
    return con;
}finally {
    try {
        if (st != null)
            st.close();
    } catch (Exception se) {
        se.printStackTrace();
    }
    try {
        if (con != null)
            con.close();
    } catch (Exception se2) {
        se2.printStackTrace();
    }
}
}
}
}

```


Código Fuente de la Clase Cargar

```
package main;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.sql.Connection;
import java.sql.Statement;
import javax.swing.JOptionPane;
/**
 * La clase se encarga de la lectura de los archivos que contienen
 * la información de los alumnos, tiene dos métodos: uno para la
 * creación de la tabla de los atributos que se envíen como parametro,
 * y otro para el llenado de dicha tabla a partir del archivo de datos.
 */
public class Cargar {
    private Conexion conexion = new Conexion();
    private Connection con;
    private Statement st;
    private File ar = null;
    private File ar1 = new File("encabezados_tipos.txt");
    private FileReader fr = null;
    private FileReader fr1 = null;
    private FileWriter fw = null;
    private BufferedReader br = null;
    private BufferedReader br1 = null;
    private BufferedWriter bw = null;
    private String linea;
    private String [] headers;//GENERO VARCHAR(15),EDAD_INGRESO
    VARCHAR(15)...
    private String [] cols;//GENERO,EDAD_INGESO....

    /**
     * El constructor intenta obtener una conexion a la BD para
     * poder realizar consultas con los demás métodos.
     */
    public Cargar()
    {
        try{
            if((con = conexion.getConexionMYSQL())==null){
                JOptionPane.showMessageDialog(null,"Error con la conexion
a la BD");
                return;
            }
            st = con.createStatement();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
    /**
     * Lee el archivo de encabezados tomando los nombres de las columnas
     * y luego crea un archivo encabezados_tipos.txt donde almacena el
     * nombre de las columnas y su tipo de variable, y a partir de este
     * crea la tabla del tipo de atributo table.
     */
}
```

```

*/
public void encabezados(String filex, String table){
    try
    {
        ar = new File(filex);
        fr = new FileReader(ar);
        br = new BufferedReader(fr);

        ar1.createNewFile();
        fw = new FileWriter(ar1);
        bw = new BufferedWriter(fw);

        linea = br.readLine();

        cols = linea.split(",");

        for (int j=0;j<cols.length;j++){
            bw.write(cols[j]+" VARCHAR(15)");
            bw.newLine();
        }
        bw.close();

        fr1 = new FileReader(ar1);
        br1 = new BufferedReader(fr1);

        StringBuilder str = new StringBuilder();
        int q=0;

        while((linea = br1.readLine()) != null)
        {
            if(q==1)
                str.append(", ");
            headers = linea.split(",");
            str.append(headers[0]).append(" ").append(headers[1]);
            q=1;
        }
        String tables = str.toString();

        String query = "CREATE TABLE IF NOT EXISTS "+table+" (AA INT
        AUTO_INCREMENT PRIMARY KEY, "+tables+")ENGINE = InnoDB;";
        st.executeUpdate(query);
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(null,"Error con la conexion a
la BD");
        e.printStackTrace();
    }
    finally
    {
        try
        {
            if(null != fr)
                fr.close();
            if(null != fr1)
                fr1.close();
        }
    }
}

```

```

    }
    catch(Exception e1)
    {
        e1.printStackTrace();
    }
}

/**
 * Lee el archivo de datos y va insertando a la tabla
 * los valores que se encuentran en este, al finalizar
 * muestra un mensaje de exito.
 */
public void llenado(String filey, String table){
    try
    {
        ar = new File(filey);
        fr = new FileReader(ar);
        br = new BufferedReader(fr);
        String [] data;
        String columns, content;
        StringBuilder str = new StringBuilder();

        int m=0;
        int n;
        int b=0;
        for (int i=0;i<cols.length;i++)
        {
            if (m==1)
                str.append(",");
            str.append(cols[i]);
            m=1;
        }

        columns = str.toString().toUpperCase();
        linea = br.readLine();
        StringBuilder str2 = new StringBuilder();

        while((linea = br.readLine()) != null)
        {
            data = linea.split(",");
            n=0;
            for (int j=0;j<data.length;j++){
                if (n==1)
                    str2.append(", ");
                str2.append("").append(data[j]).append("");
                n=1;
            }
            content = str2.toString().toUpperCase();
            String query = "INSERT INTO "+table+" ("+columns+"
VALUES ("+content+");";

            st.executeUpdate(query);
            str2.delete(0, 500);
            b++;
        }
        JOptionPane.showMessageDialog(null, "Carga Finalizada");
    }
}

```

```
    } catch (Exception e) {
        e.printStackTrace();
        JOptionPane.showMessageDialog(null, "Error con la conexion a
la BD");
    }
    finally
    {
        try
        {
            if(null != fr)
                fr.close();
        }
        catch(Exception e1)
        {
            e1.printStackTrace();
        }
    }
}
}
```

Código Fuente de la Clase Seleccionar

```
package main;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.JOptionPane;
/**
 * La clase se encarga de realizar las diversas consultas
 * a la BD usando tres métodos para: obtener el conteo del
 * total de datos, total de alumnos que desertaron y de los
 * que no, y un conteo dependiente del valor que reciba.
 */
public class Seleccionar {
    private Conexion conexion = new Conexion();
    private Connection con;
    private Statement st;
    private ResultSet rs;
    private String [] aca_colsname=
{"GENERO","EDAD_INGRESO","EXAMEN_ADM","PROMEDIO_SECU","TIPO_SECU","VECES_
CAMBIO",

"TIEMPO_TRAS","RAZON_ING","ASISTENCIA","PROMEDIO_ULTIMO","APROBACION","AP
ROVECHAMIENTO","BECA","APOYO_ESCUELA",
    "PROB_COMP","PROB_ESC","DESERTO"};
    private String []
pers_colsname={"GENERO","VIVIA_CON","APOYO_FAMILIA","AMIGOS_ABANDONAN","H
ERMANOS_ABANDONAN",

"CIGARRO","ALCOHOL","DROGAS","ACOSO","PROBLEMAS_CASA","FAMILIA_PROPIA","F
ALTA_DINERO","BAJA_AUTO","ENFERMEDAD",
    "POCO_GUSTO","ESTUDIOS_OTROS","FINALIZARON_MS","DESERTO"};

    /**
     * El constructor intenta obtener una conexion a la BD para
     * poder realizar consultas con los demás métodos.
     */
    public Seleccionar() {
        try{
            if((con = conexion.getConnectionMySQL())==null){
                JOptionPane.showMessageDialog(null,"Error con la conexion
a la BD");
                return;
            }
            st = con.createStatement();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
    /**
     * Regresa el total de datos que se encuentran en la
     * tabla que se pasa como parametro.
     */
    public int total_datos(String table){
        int res=0;
        try
```

```

    {
        String query = "SELECT COUNT(*) FROM "+table;
        rs = st.executeQuery(query);
        rs.next();
        res = rs.getInt("COUNT(*)");

        return res;

    }catch(Exception e)
    {
        JOptionPane.showMessageDialog(null,"No hay datos en la tabla
"+table);
        //e.printStackTrace();
        return res;
    }
}

/**
 * Regresa el total de alumnos de la tabla que se recibe como
 * parametro, tomando como condición si desertaron o no.
 */
public int total_sino(String sino,String table){
    int res=0;
    try
    {
        String query;
        if(table.equalsIgnoreCase("personal")){
            query = "SELECT COUNT(*) FROM "+table+" WHERE
"+pers_colname[pers_colname.length-1]+" = '" +sino+ "'";
        }else
        {
            query = "SELECT COUNT(*) FROM "+table+" WHERE
"+aca_colname[aca_colname.length-1]+" = '" +sino+ "'";
        }

        rs = st.executeQuery(query);
        rs.next();
        res = rs.getInt("COUNT(*)");

        return res;

    }catch(Exception e)
    {
        //JOptionPane.showMessageDialog(null,"No hay datos en la
tabla "+table);
        //e.printStackTrace();
        return res;
    }
}

/**
 * Regresa la cuenta de los alumnos que cumplen las condiciones
 * que se reciben, del tipo de atributo que se especifica con
 * table y si deserto o no deserto con sino.
 */
public int conteo(int x, String cond, String sino,String table){

```

```

int res=0;
try
{
    String query;
    if(table.equalsIgnoreCase("personal")){
        query ="SELECT COUNT(*) FROM "+table+" WHERE
"+pers_colsname[x]+
        " = '" +cond + "' AND
"+pers_colsname[pers_colsname.length-1]+" = '" +sino +"'";
    }else
    {
        query ="SELECT COUNT(*) FROM "+table+" WHERE
"+aca_colsname[x]+
        " = '" +cond + "' AND
"+aca_colsname[aca_colsname.length-1]+" = '" +sino +"'";
    }
    rs = st. executeQuery(query);
    rs.next();
    res = rs.getInt("COUNT(*)");

    return res;
}catch(Exception e)
{
    //JOptionPane.showMessageDialog(null,"No hay datos en la
tabla "+table);
    //e.printStackTrace();
    return res;
}
}
}

```

Código Fuente de la Clase Clasificar

```
package main;
/**
 * Se encarga de realizar el cálculo de probabilidad de que
 * el alumno finalice o no, aplica la fórmula del Clasificador
 * Bayesiano Ingenuo.
 */
public class Clasificar {
    /**
     * Método específico encargado de clasificar datos de
     * tipo personal.
     */
    public float c_pers(float[] si,float[] no,int total_datos,float
total_deserto,float total_finalizo,String yn)
    {
        float res;
        float p_no_finalizo, p_si_finalizo;
        float p_deserto,p_finalizo;
        float p_total_deserto;
        float p_total_finalizo;

        p_deserto = total_deserto / total_datos;
        p_finalizo = total_finalizo / total_datos;

        p_no_finalizo =
si[0]*si[1]*si[2]*si[3]*si[4]*si[5]*si[6]*si[7]*si[8]*si[9]*
si[10]*si[11]*si[12]*si[13]*si[14]*si[15]*si[16]*p_deserto;

        p_si_finalizo =
no[0]*no[1]*no[2]*no[3]*no[4]*no[5]*no[6]*no[7]*no[8]*no[9]*
no[10]*no[11]*no[12]*no[13]*no[14]*no[15]*no[16]*p_finalizo;

        p_total_deserto = p_no_finalizo / (p_no_finalizo+p_si_finalizo);
        p_total_finalizo = 1 - p_total_deserto;

        if(yn.equalsIgnoreCase("si"))
        {
            res=p_total_deserto;
        }else
            res=p_total_finalizo;

        return res;
    }
    /**
     * Método específico encargado de clasificar datos de
     * tipo académico.
     */
    public float c_acad(float[] si,float[] no,int total_datos,float
total_deserto,float total_finalizo,String yn)
    {
        float res;
        float p_no_finalizo, p_si_finalizo;
        float p_deserto,p_finalizo;
        float p_total_deserto;
```



```

float p_total_finalizo;

p_deserto = total_deserto / total_datos;
p_finalizo = total_finalizo / total_datos;

p_no_finalizo =
si[0]*si[1]*si[2]*si[3]*si[4]*si[5]*si[6]*si[7]*si[8]*si[9]*
    si[10]*si[11]*si[12]*si[13]*si[14]*si[15]*p_deserto;
p_si_finalizo =
no[0]*no[1]*no[2]*no[3]*no[4]*no[5]*no[6]*no[7]*no[8]*no[9]*
    no[10]*no[11]*no[12]*no[13]*no[14]*no[15]*p_finalizo;

p_total_deserto = p_no_finalizo / (p_no_finalizo+p_si_finalizo);
p_total_finalizo = 1 - p_total_deserto;

if(yn.equalsIgnoreCase("si"))
{
    res=p_total_deserto;
}else
    res=p_total_finalizo;

return res;
}
}

```

Código Fuente de la Clase Recomendar

```
package main;
/**
 * Clase que se encarga de mostrar las recomendaciones,
 * crea una nueva interfaz gráfica que se muestra en una
 * ventana emergente, utiliza instancias de las clases
 * Seleccionar y Clasificar.
 */
public class Recomendar extends javax.swing.JFrame {

    private final String [] asistencia = {"Bajo","Medio","Alto"};
    private final String [] aprobacion = {"Bajo","Medio","Alto"};
    private final String [] aprovechamiento = {"Bajo","Medio","Alto"};
    private final String [] beca = {"Si","No"};
    private final String [] apoyo = {"Si","No"};
    private final String [] problemas_comp = {"No","Si"};
    private final String [] problemas_esc = {"No","Si"};

    private final String [] cigarro = {"Alto","Medio","Bajo","No"};
    private final String [] alcohol = {"Alto","Medio","Bajo","No"};
    private final String [] drogas = {"Bajo","No"};
    private final String [] acoso = {"No","Si"};
    private final String [] problemas_casa = {"No","Si"};
    private final String [] autoestima = {"No","Si"};
    private final String [] gusto_est = {"No","Si"};

    private final Seleccionar selection;
    private final Clasificar clasific;

    /**
     * Constructor de la clase que inicializa los componentes.
     */
    public Recomendar()
    {
        initComponents();
        selection = new Seleccionar();
        clasific = new Clasificar();
        jTextField1.setEditable(false);
        jTextField2.setEditable(false);
        jTextField3.setEditable(false);
        jTextField4.setEditable(false);
        jTextField5.setEditable(false);
        jTextField6.setEditable(false);
        jTextField7.setEditable(false);
    }

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

        jPanel4 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
    }
}
```

```
jLabel5 = new javax.swing.JLabel ();
jLabel6 = new javax.swing.JLabel ();
jLabel7 = new javax.swing.JLabel ();
jLabel8 = new javax.swing.JLabel ();
jTextField1 = new javax.swing.JTextField ();
jTextField2 = new javax.swing.JTextField ();
jTextField3 = new javax.swing.JTextField ();
jTextField4 = new javax.swing.JTextField ();
jTextField5 = new javax.swing.JTextField ();
jTextField6 = new javax.swing.JTextField ();
jTextField7 = new javax.swing.JTextField ();
jLabel9 = new javax.swing.JLabel ();
jLabel10 = new javax.swing.JLabel ();
jLabel11 = new javax.swing.JLabel ();
jLabel12 = new javax.swing.JLabel ();
jLabel13 = new javax.swing.JLabel ();
jLabel14 = new javax.swing.JLabel ();
jLabel15 = new javax.swing.JLabel ();
jLabel16 = new javax.swing.JLabel ();
jLabel17 = new javax.swing.JLabel ();
jLabel18 = new javax.swing.JLabel ();
jLabel19 = new javax.swing.JLabel ();
jLabel20 = new javax.swing.JLabel ();
jLabel21 = new javax.swing.JLabel ();
jLabel22 = new javax.swing.JLabel ();
jLabel23 = new javax.swing.JLabel ();
jLabel24 = new javax.swing.JLabel ();
jLabel25 = new javax.swing.JLabel ();
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
setTitle("Recomendaciones");
```

```
jLabel11.setText("Los cambios que se pueden realizar para mejorar  
la probabilidad de no desertar son los siguientes:");
```

```
jLabel2.setText("-----");
```

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

```
jLabel4.setText("-----");
```

```
jLabel5.setText("-----");
```

```
jLabel6.setText("-----");
```

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

```
jLabel8.setText("-----");
```

```
jLabel9.setText("Nuevo % Desertar");
```

```
jLabel10.setText("Nuevo % Finalizar");
```

```
jLabel11.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
```

```

jLabel12.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel13.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel14.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel15.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel16.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
    jLabel17.setFocusable(false);

jLabel17.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel18.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel19.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel20.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel21.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel22.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel23.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

jLabel24.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);

    javax.swing.GroupLayout jPanel4Layout = new
javax.swing.GroupLayout(jPanel4);
    jPanel4.setLayout(jPanel4Layout);
    jPanel4Layout.setHorizontalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel4Layout.createSequentialGroup()
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addGroup(jPanel4Layout.createParallelGroup(javax.
swing.
GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel4Layout.createSequentialGroup()
                        .addGap(77, 77, 77)
                        .addComponent(jLabel11))
                    .addGroup(jPanel4Layout.createSequentialGroup()
                        .addGap(53, 53, 53)
                        .addGroup(jPanel4Layout.createParallelGroup(
javax.
swing.
GroupLayout.Alignment.LEADING)

```

```

        .addGroup(javax.swing.GroupLayout.Alignment.T
RAILING, jPanel4Layout.createSequentialGroup())
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED, 337, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel9,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel10,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel4Layout.createSequentialGroup
())
        .addGroup(jPanel4Layout.createParallelGro
up(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel12,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel13,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel14,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel15,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel16,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel17,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel18,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(50, 50, 50)
        .addGroup(jPanel4Layout.createParallelGro
up(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(jTextField1)
        .addComponent(jTextField2)
        .addComponent(jTextField3)
        .addComponent(jTextField4)
        .addComponent(jTextField5)
        .addComponent(jTextField6)
        .addComponent(jTextField7,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel4Layout.createParallelGro
up(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel4Layout.createSequent
ialGroup())
        .addGap(28, 28, 28)
        .addComponent(jLabel15,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE))

```

```

        .addGroup(javax.swing.GroupLayout.Ali
gment.TRAILING, jPanel4Layout.createSequentialGroup()
        .addPreferredGap(javax.swing.Layo
utStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel4Layout.createPar
allelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel116,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel117,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel113,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel112,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel111,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel114,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGro
up(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel118,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel119,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel120,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel122,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel123,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel124,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel121,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addComponent(jLabel125,
javax.swing.GroupLayout.PREFERRED_SIZE, 555,
javax.swing.GroupLayout.PREFERRED_SIZE))))

```

```

        .addContainerGap(55, Short.MAX_VALUE))
    );
    jPanel4Layout.setVerticalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addGap(29, 29, 29)
        .addComponent(jLabel1)
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel9,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel10))
        .addGap(22, 22, 22)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel18,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel11,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel2)
                .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel12,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel19,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel3)
                .addComponent(jTextField2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel13,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(jLabel20,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.
ng.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel4)
            .addComponent(jTextField3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.G
roupLayout.Alignment.TRAILING)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel5)
                .addComponent(jTextField4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addComponent(jLabel14,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel21,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.G
roupLayout.Alignment.LEADING)
            .addComponent(jLabel15,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel22,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel6)
                .addComponent(jTextField5,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.G
roupLayout.Alignment.LEADING)
            .addComponent(jLabel16,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel23,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel7)

```



```

        .addComponent(jTextField6,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.G
roupLayout.Alignment.LEADING)
        .addComponent(jLabel17,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel24,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel4Layout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel8)
        .addComponent(jTextField7,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(35, 35, 35)
        .addComponent(jLabel25,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(56, Short.MAX_VALUE))
    );

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

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );

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

/**
 * Dependiendo del valor de table actualiza las etiquetas
 * de texto con los atributos usados para la recomendación.
 */
public void startUp(String table) {
    if(table.equalsIgnoreCase("academico"))
    {
        jLabel2.setText("Promedio de asistencia");
        jLabel3.setText("Porcentaje de aprobación");
    }
}

```

```

        jLabel4.setText("Aprovechamiento");
        jLabel5.setText("Beca");
        jLabel6.setText("Apoyo de la escuela");
        jLabel7.setText("Problemas con compañeros");
        jLabel8.setText("Problemas escolares");
    }else
    {
        jLabel2.setText("Cigarro");
        jLabel3.setText("Alcohol");
        jLabel4.setText("Drogas");
        jLabel5.setText("Acoso");
        jLabel6.setText("Problemas en casa");
        jLabel7.setText("Baja autoestima");
        jLabel8.setText("Poco gusto por estudiar");
    }
}

/**
 * Método específico para recomendar los atributos de tipo
 * académico, recibe como parametro las variables usadas para
 * el cálculo de la probabilidad, además de los valores que
 * están seleccionados en la pantalla de selección de atributos,
 * actualiza las etiquetas al encontrar mejores valores para que
 * el alumno finalice sus estudios.
 */
public void Recommendation_acad (float[] si,float[] no,int
total_datos,float total_desert,float total_finalizo,
        float deserto_acad,float finalizo_acad,String asist,String
aprob,String aprov,String bec,String apo,
        String prob_comp,String prob_esc)
{
    float[] si_final = new
float[] {si[8],si[10],si[11],si[12],si[13],si[14],si[15]};
    float[] no_final = new
float[] {no[8],no[10],no[11],no[12],no[13],no[14],no[15]};

    String mejora = asist;
    float mejora_des = deserto_acad;
    float mejora_fin = finalizo_acad;
    float temp_des,temp_fin;
    float temp_si, temp_no;
    temp_si = si[8];
    temp_no = no[8];
    for (int i=0;i<asistencia.length;i++){
        if(!asist.equalsIgnoreCase(asistencia[i]))
        {
            si[8] = selection.conteo(8, asistencia[i].toUpperCase(),
"SI", "academico")/total_desert;
            no[8] = selection.conteo(8, asistencia[i].toUpperCase(),
"NO", "academico")/total_finalizo;
            temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
            temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
            if(temp_fin>mejora_fin)
            {

```

```

        mejora_fin = temp_fin;
        mejora_des = temp_des;
        mejora = asistencia[i];
        si_final[0] = si[8];
        no_final[0] = no[8];
    }
}
jTextField1.setText(mejora.toUpperCase());
jLabel18.setText(""+mejora_fin*100);
jLabel11.setText(""+mejora_des*100);

si[8] = temp_si;
no[8] = temp_no;
mejora = aprob;
mejora_des = deserto_acad;
mejora_fin = finalizo_acad;
temp_si = si[10];
temp_no = no[10];
for (int i=0;i<aprobacion.length;i++){
    if(!aprob.equalsIgnoreCase(aprobacion[i]))
    {
        si[10] = selection.conteo(10,
aprobacion[i].toUpperCase(), "SI", "academico")/total_desert;
        no[10] = selection.conteo(10,
aprobacion[i].toUpperCase(), "NO", "academico")/total_finalizo;
        temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
        temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = aprobacion[i];
            si_final[1] = si[10];
            no_final[1] = no[10];
        }
    }
}
jTextField2.setText(mejora.toUpperCase());
jLabel19.setText(""+mejora_fin*100);
jLabel12.setText(""+mejora_des*100);

si[10] = temp_si;
no[10] = temp_no;
mejora = aprov;
mejora_des = deserto_acad;
mejora_fin = finalizo_acad;
temp_si = si[11];
temp_no = no[11];
for (int i=0;i<aprovechamiento.length;i++){
    if(!aprov.equalsIgnoreCase(aprovechamiento[i]))
    {
        si[11] = selection.conteo(11,
aprovechamiento[i].toUpperCase(), "SI", "academico")/total_desert;

```

```

        no[11] = selection.conteo(11,
aprovechamiento[i].toUpperCase(), "NO", "academico")/total_finalizo;
        temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
        temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = aprovechamiento[i];
            si_final[2] = si[11];
            no_final[2] = no[11];
        }
    }
}
jTextField3.setText(mejora.toUpperCase());
jLabel20.setText(""+mejora_fin*100);
jLabel13.setText(""+mejora_des*100);

si[11] = temp_si;
no[11] = temp_no;
mejora = bec;
mejora_des = deserto_acad;
mejora_fin = finalizo_acad;
temp_si = si[12];
temp_no = no[12];
for (int i=0;i<beca.length;i++){
    if(!bec.equalsIgnoreCase(beca[i]))
    {
        si[12] = selection.conteo(12, beca[i].toUpperCase(),
"SI", "academico")/total_desert;
        no[12] = selection.conteo(12, beca[i].toUpperCase(),
"NO", "academico")/total_finalizo;
        temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
        temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = beca[i];
            si_final[3] = si[12];
            no_final[3] = no[12];
        }
    }
}
jTextField4.setText(mejora.toUpperCase());
jLabel21.setText(""+mejora_fin*100);
jLabel14.setText(""+mejora_des*100);

si[12] = temp_si;
no[12] = temp_no;
mejora = apo;
mejora_des = deserto_acad;

```

```

mejora_fin = finalizo_acad;
temp_si = si[13];
temp_no = no[13];
for (int i=0;i<apoyo.length;i++){
    if(!apo.equalsIgnoreCase(apoyo[i]))
    {
        si[13] = selection.conteo(13, apoyo[i].toUpperCase(),
"SI", "academico")/total_desert;
        no[13] = selection.conteo(13, apoyo[i].toUpperCase(),
"NO", "academico")/total_finalizo;
        temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
        temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = apoyo[i];
            si_final[4] = si[13];
            no_final[4] = no[13];
        }
    }
}
jTextField5.setText(mejora.toUpperCase());
jLabel22.setText(""+mejora_fin*100);
jLabel15.setText(""+mejora_des*100);

si[13] = temp_si;
no[13] = temp_no;
mejora = prob_comp;
mejora_des = deserto_acad;
mejora_fin = finalizo_acad;
temp_si = si[14];
temp_no = no[14];
for (int i=0;i<problemas_comp.length;i++){
    if(!prob_comp.equalsIgnoreCase(problemas_comp[i]))
    {
        si[14] = selection.conteo(14,
problemas_comp[i].toUpperCase(), "SI", "academico")/total_desert;
        no[14] = selection.conteo(14,
problemas_comp[i].toUpperCase(), "NO", "academico")/total_finalizo;
        temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
        temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = problemas_comp[i];
            si_final[5] = si[14];
            no_final[5] = no[14];
        }
    }
}

```

```

    }
    jTextField6.setText(mejora.toUpperCase());
    jLabel23.setText(""+mejora_fin*100);
    jLabel16.setText(""+mejora_des*100);

    si[14] = temp_si;
    no[14] = temp_no;
    mejora = prob_esc;
    mejora_des = deserto_acad;
    mejora_fin = finalizo_acad;
    temp_si = si[15];
    temp_no = no[15];
    for (int i=0;i<problemas_esc.length;i++){
        if(!prob_esc.equalsIgnoreCase(problemas_esc[i]))
        {

            si[15] = selection.conteo(15,
problemas_esc[i].toUpperCase(), "SI", "academico")/total_desert;
            no[15] = selection.conteo(15,
problemas_esc[i].toUpperCase(), "NO", "academico")/total_finalizo;
            temp_des = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "si");
            temp_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
            if(temp_fin>mejora_fin)
            {
                mejora_fin = temp_fin;
                mejora_des = temp_des;
                mejora = problemas_esc[i];
                si_final[6] = si[15];
                no_final[6] = no[15];
            }
        }
    }
    jTextField7.setText(mejora.toUpperCase());
    jLabel24.setText(""+mejora_fin*100);
    jLabel17.setText(""+mejora_des*100);
    si[15] = temp_si;
    no[15] = temp_no;

    si[8] = si_final[0];
    no[8] = no_final[0];

    si[10] = si_final[1];
    no[10] = no_final[1];

    si[11] = si_final[2];
    no[11] = no_final[2];

    si[12] = si_final[3];
    no[12] = no_final[3];

    si[13] = si_final[4];
    no[13] = no_final[4];

    si[14] = si_final[5];
    no[14] = no_final[5];

```

```

        si[15] = si_final[6];
        no[15] = no_final[6];
        float next_fin = clasific.c_acad(si, no, total_datos,
total_desert, total_finalizo, "no");
        jLabel25.setText("Si sigue todas las recomendaciones anteriores,
la probabilidad de finalizar"
        + " cambia de "+finalizo_acad*100+"% a
"+next_fin*100+"%");
    }

    /**
     * Método específico para recomendar los atributos de tipo
     * personal, recibe como parametro las variables usadas para
     * el cálculo de la probabilidad, además de los valores que
     * están seleccionados en la pantalla de selección de atributos,
     * actualiza las etiquetas al encontrar mejores valores para que
     * el alumno finalice sus estudios.
     */
    public void Recommendation_pers (float[] si,float[] no,int
total_datos,float total_desert,float total_finalizo,
        float deserto_pers,float finalizo_pers,String cig,String
alc,String drog,String acos,String prob_casa,
        String autoest,String gusto)
    {
        float[] si_final = new
float[]{si[5],si[6],si[7],si[8],si[9],si[12],si[14]};
        float[] no_final = new
float[]{no[5],no[7],no[7],no[8],no[9],no[12],no[14]};

        String mejora = cig;
        float mejora_des = deserto_pers;
        float mejora_fin = finalizo_pers;
        float temp_des,temp_fin;
        float temp_si, temp_no;
        temp_si = si[5];
        temp_no = no[5];
        for (int i=0;i<cigarro.length;i++){
            if(!cig.equalsIgnoreCase(cigarro[i]))
            {

                si[5] = selection.conteo(5, cigarro[i].toUpperCase(),
"SI", "personal")/total_desert;
                no[5] = selection.conteo(5, cigarro[i].toUpperCase(),
"NO", "personal")/total_finalizo;
                temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");
                temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
                if(temp_fin>mejora_fin)
                {
                    mejora_fin = temp_fin;
                    mejora_des = temp_des;
                    mejora = cigarro[i];
                    si_final[0] = si[5];
                    no_final[0] = no[5];

```

```

        }
    }
}
jTextField1.setText(mejora.toUpperCase());
jLabel18.setText(""+mejora_fin*100);
jLabel11.setText(""+mejora_des*100);

si[5] = temp_si;
no[5] = temp_no;
mejora = alc;
mejora_des = deserto_pers;
mejora_fin = finalizo_pers;
temp_si = si[6];
temp_no = no[6];
for (int i=0;i<alcohol.length;i++){
    if(!alc.equalsIgnoreCase(alcohol[i]))
    {
        si[6] = selection.conteo(6, alcohol[i].toUpperCase(),
"SI", "personal")/total_desert;
        no[6] = selection.conteo(6, alcohol[i].toUpperCase(),
"NO", "personal")/total_finalizo;
        temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");
        temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = alcohol[i];
            si_final[1] = si[6];
            no_final[1] = no[6];
        }
    }
}
jTextField2.setText(mejora.toUpperCase());
jLabel19.setText(""+mejora_fin*100);
jLabel12.setText(""+mejora_des*100);

si[6] = temp_si;
no[6] = temp_no;
mejora = drog;
mejora_des = deserto_pers;
mejora_fin = finalizo_pers;
temp_si = si[7];
temp_no = no[7];
for (int i=0;i<drogas.length;i++){
    if(!drog.equalsIgnoreCase(drogas[i]))
    {
        si[7] = selection.conteo(7, drogas[i].toUpperCase(),
"SI", "personal")/total_desert;
        no[7] = selection.conteo(7, drogas[i].toUpperCase(),
"NO", "personal")/total_finalizo;
        temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");

```



```

        temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
        if(temp_fin>mejora_fin)
        {
            mejora_fin = temp_fin;
            mejora_des = temp_des;
            mejora = drogas[i];
            si_final[2] = si[7];
            no_final[2] = no[7];
        }
    }
    jTextField3.setText(mejora.toUpperCase());
    jLabel20.setText(""+mejora_fin*100);
    jLabel13.setText(""+mejora_des*100);

    si[7] = temp_si;
    no[7] = temp_no;
    mejora = acos;
    mejora_des = deserto_pers;
    mejora_fin = finalizo_pers;
    temp_si = si[8];
    temp_no = no[8];
    for (int i=0;i<acos.length;i++){
        if(!acos.equalsIgnoreCase(acos[i]))
        {
            si[8] = selection.conteo(8, acos[i].toUpperCase(), "SI",
"personal")/total_desert;
            no[8] = selection.conteo(8, acos[i].toUpperCase(), "NO",
"personal")/total_finalizo;
            temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");
            temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
            if(temp_fin>mejora_fin)
            {
                mejora_fin = temp_fin;
                mejora_des = temp_des;
                mejora = acos[i];
                si_final[3] = si[8];
                no_final[3] = no[8];
            }
        }
    }
    jTextField4.setText(mejora.toUpperCase());
    jLabel21.setText(""+mejora_fin*100);
    jLabel14.setText(""+mejora_des*100);

    si[8] = temp_si;
    no[8] = temp_no;
    mejora = prob_casa;
    mejora_des = deserto_pers;
    mejora_fin = finalizo_pers;
    temp_si = si[9];
    temp_no = no[9];
    for (int i=0;i<problemas_casa.length;i++){

```

```

        if(!prob_casa.equalsIgnoreCase(problemas_casa[i]))
        {
            si[9] = selection.conteo(9,
problemas_casa[i].toUpperCase(), "SI", "personal")/total_desert;
            no[9] = selection.conteo(9,
problemas_casa[i].toUpperCase(), "NO", "personal")/total_finalizo;
            temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");
            temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
            if(temp_fin>mejora_fin)
            {
                mejora_fin = temp_fin;
                mejora_des = temp_des;
                mejora = problemas_casa[i];
                si_final[4] = si[9];
                no_final[4] = no[9];
            }
        }
    }
    jTextField5.setText(mejora.toUpperCase());
    jLabel22.setText(""+mejora_fin*100);
    jLabel15.setText(""+mejora_des*100);

    si[9] = temp_si;
    no[9] = temp_no;
    mejora = autoest;
    mejora_des = deserto_pers;
    mejora_fin = finalizo_pers;
    temp_si = si[12];
    temp_no = no[12];
    for (int i=0;i<autoestima.length;i++){
        if(!autoest.equalsIgnoreCase(autoestima[i]))
        {
            si[12] = selection.conteo(12,
autoestima[i].toUpperCase(), "SI", "personal")/total_desert;
            no[12] = selection.conteo(12,
autoestima[i].toUpperCase(), "NO", "personal")/total_finalizo;
            temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");
            temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
            if(temp_fin>mejora_fin)
            {
                mejora_fin = temp_fin;
                mejora_des = temp_des;
                mejora = autoestima[i];
                si_final[5] = si[12];
                no_final[5] = no[12];
            }
        }
    }
    jTextField6.setText(mejora.toUpperCase());
    jLabel23.setText(""+mejora_fin*100);
    jLabel16.setText(""+mejora_des*100);

```

```

    si[12] = temp_si;
    no[12] = temp_no;
    mejora = gusto;
    mejora_des = deserto_pers;
    mejora_fin = finalizo_pers;
    temp_si = si[14];
    temp_no = no[14];
    for (int i=0;i<gusto_est.length;i++){
        if(!gusto.equalsIgnoreCase(gusto_est[i]))
        {
            si[14] = selection.conteo(14, gusto_est[i].toUpperCase(),
"SI", "personal")/total_desert;
            no[14] = selection.conteo(14, gusto_est[i].toUpperCase(),
"NO", "personal")/total_finalizo;
            temp_des = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "si");
            temp_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
            if(temp_fin>mejora_fin)
            {
                mejora_fin = temp_fin;
                mejora_des = temp_des;
                mejora = gusto_est[i];
                si_final[6] = si[14];
                no_final[6] = no[14];
            }
        }
    }
    jTextField7.setText(mejora.toUpperCase());
    jLabel24.setText(""+mejora_fin*100);
    jLabel17.setText(""+mejora_des*100);
    si[14] = temp_si;
    no[14] = temp_no;

    si[5] = si_final[0];
    no[5] = no_final[0];

    si[6] = si_final[1];
    no[6] = no_final[1];

    si[7] = si_final[2];
    no[7] = no_final[2];

    si[8] = si_final[3];
    no[8] = no_final[3];

    si[9] = si_final[4];
    no[9] = no_final[4];

    si[12] = si_final[5];
    no[12] = no_final[5];

    si[14] = si_final[6];
    no[14] = no_final[6];

```

```

        float next_fin = clasific.c_pers(si, no, total_datos,
total_desert, total_finalizo, "no");
        jLabel25.setText("Si sigue todas las recomendaciones anteriores,
la probabilidad de finalizar"
        + " cambia de "+finalizo_pers*100+"% a
"+next_fin*100+"%");
    }

    public static void main(String args[]) {
        /* Crea y muestra la interfaz */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Recomendar().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;
    private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
    private javax.swing.JLabel jLabel13;
    private javax.swing.JLabel jLabel14;
    private javax.swing.JLabel jLabel15;
    private javax.swing.JLabel jLabel16;
    private javax.swing.JLabel jLabel17;
    private javax.swing.JLabel jLabel18;
    private javax.swing.JLabel jLabel19;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel20;
    private javax.swing.JLabel jLabel21;
    private javax.swing.JLabel jLabel22;
    private javax.swing.JLabel jLabel23;
    private javax.swing.JLabel jLabel24;
    private javax.swing.JLabel jLabel25;
    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.JLabel jLabel9;
    private javax.swing.JPanel jPanel4;
    private javax.swing.JTextField jTextField1;
    private javax.swing.JTextField jTextField2;
    private javax.swing.JTextField jTextField3;
    private javax.swing.JTextField jTextField4;
    private javax.swing.JTextField jTextField5;
    private javax.swing.JTextField jTextField6;
    private javax.swing.JTextField jTextField7;
    // End of variables declaration
}

```


Código Fuente de la Clase Graficar

```
package main;

import org.jfree.data.category.DefaultCategoryDataset;
/**
 * Clase encargada de obtener los valores que se utilizarán
 * para las gráficas de barras.
 */
public class Graficar {
    private final String[] traslado = {"BAJO","MEDIO","ALTO"};
    private final String[] asistencia = {"BAJO","MEDIO","ALTO"};
    private final String[] aprobacion = {"BAJO","MEDIO","ALTO"};
    private final String[] aprovechamiento = {"BAJO","MEDIO","ALTO"};
    private final String[] beca = {"SI","NO"};

    private final String[] genero = {"HOMBRE","MUJER"};
    private final String[] cigarro = {"NO","BAJO","MEDIO","ALTO"};
    private final String[] alcohol = {"NO","BAJO","MEDIO","ALTO"};
    private final String[] autoestima = {"NO","SI"};
    private final String[] padres = {"NINGUNO","MADRE","PADRE","AMBOS"};
    private final Seleccionar select = new Seleccionar();
    /**
     * El método regresa conjunto de datos para crear la gráfica
     * dependiendo del tipo de atributo que se reciba en la variable
     * table, y de si desertaron o no según el valor de sino.
     */
    public DefaultCategoryDataset datosBarra(String table,String sino){

        DefaultCategoryDataset result = new DefaultCategoryDataset();
        select.total_datos(table);
        if (table.equalsIgnoreCase("personal"))
        {
            result.addValue(select.conteo(0, genero[0], sino, table),
genero[0], "Genero");
            result.addValue(select.conteo(0, genero[1], sino, table),
genero[1], "Genero");
            result.addValue(select.conteo(6, cigarro[0],sino, table),
cigarro[0], "Cigarro");
            result.addValue(select.conteo(6, cigarro[1],sino, table),
cigarro[1], "Cigarro");
            result.addValue(select.conteo(6, cigarro[2],sino, table),
cigarro[2], "Cigarro");
            result.addValue(select.conteo(6, cigarro[3],sino, table),
cigarro[3], "Cigarro");
            result.addValue(select.conteo(7, alcohol[0],sino, table),
alcohol[0], "Alcohol");
            result.addValue(select.conteo(7, alcohol[1],sino, table),
alcohol[1], "Alcohol");
            result.addValue(select.conteo(7, alcohol[2],sino, table),
alcohol[2], "Alcohol");
            result.addValue(select.conteo(7, alcohol[3],sino, table),
alcohol[3], "Alcohol");

            result.addValue(select.conteo(13,autoestima[0],sino,table),autoestima[0],
"Baja Autoestima");
        }
    }
}
```

```

result.addValue(select.conteo(13,autoestima[1],sino,table),autoestima[1],
"Baja Autoestima");
    result.addValue(select.conteo(16,padres[0],sino,table),
padres[0], "Padres con estudios");
    result.addValue(select.conteo(16,padres[1],sino,table),
padres[1], "Padres con estudios");
    result.addValue(select.conteo(16,padres[2],sino,table),
padres[2], "Padres con estudios");
    result.addValue(select.conteo(16,padres[3],sino,table),
padres[3], "Padres con estudios");
    }else
    {
        result.addValue(select.conteo(6, traslado[0], sino,
table),traslado[0],"Tiempo Traslado");
        result.addValue(select.conteo(6, traslado[1], sino,
table),traslado[1],"Tiempo Traslado");
        result.addValue(select.conteo(6, traslado[2], sino,
table),traslado[2],"Tiempo Traslado");
        result.addValue(select.conteo(8, asistencia[0], sino, table),
asistencia[0], "Asistencia");
        result.addValue(select.conteo(8, asistencia[1], sino, table),
asistencia[1], "Asistencia");
        result.addValue(select.conteo(8, asistencia[2], sino, table),
asistencia[2], "Asistencia");
        result.addValue(select.conteo(10, aprobacion[0], sino,
table), aprobacion[0], "Aprobación");
        result.addValue(select.conteo(10, aprobacion[1], sino,
table), aprobacion[1], "Aprobación");
        result.addValue(select.conteo(10, aprobacion[2], sino,
table), aprobacion[2], "Aprobación");
        result.addValue(select.conteo(11, aprovechamiento[0], sino,
table), aprovechamiento[0], "Aprovechamiento");
        result.addValue(select.conteo(11, aprovechamiento[1], sino,
table), aprovechamiento[1], "Aprovechamiento");
        result.addValue(select.conteo(11, aprovechamiento[2], sino,
table), aprovechamiento[2], "Aprovechamiento");
        result.addValue(select.conteo(12, beca[0], sino,
table),beca[0],"Beca");
        result.addValue(select.conteo(12, beca[1], sino,
table),beca[1],"Beca");
    }
    return result;
}
}

```

Código Fuente de la Clase Main

```
package main;

import java.awt.Dimension;
import java.sql.Connection;
import javax.swing.JComboBox;
import javax.swing.JFileChooser;
import javax.swing.JOptionPane;
import javax.swing.SwingUtilities;
import javax.swing.UIManager;
import org.jfree.chart.ChartFactory;
import org.jfree.chart.ChartPanel;
import org.jfree.chart.JFreeChart;
import org.jfree.chart.plot.PlotOrientation;
import org.jfree.data.category.DefaultCategoryDataset;

/**
 * La clase Main es donde se muestra la mayor parte de la interfaz
 gráfica,
 * a excepción de las recomendaciones, el diseño se realizó utilizando
 * NetBeans IDE 8.2 y cuenta con un panel con tres pestañas para la
 función
 * de cargar archivos, clasificar y presentar estadísticas.
 */
public class Main extends javax.swing.JFrame {
    private UIManager.LookAndFeelInfo apariencias[];
    private Conexion conexion = new Conexion();
    private Connection con;
    private Seleccionar selection;
    private Clasificar clasification;
    private Recomendar recomnd;
    private Cargar load;
    private String filex = "";
    private String filey = "";
    private String table = "";
    private float deserto_pers=0;
    private float finalizo_pers=0;
    private float [] si_pers = new float[17];
    private float [] no_pers = new float[17];
    private int total_datos_pers =0;
    private float total_desert_pers = 0,total_final_pers = 0;
    private float finalizo_acad = 0;
    private float deserto_acad = 0;
    private float [] si_acad = new float[16];
    private float [] no_acad = new float[16];
    private int total_datos_acad =0;
    private float total_desert_acad = 0,total_final_acad = 0;

    /**
     * El constructor intenta obtener una conexion al servidor para
     * poder verificar que existe la base de datos, además de cargar
     * el estilo de la interfaz que se utilizará, también crea estancias
     * de las clases Seleccionar, Cargar, Clasificar y Recomendar.
     */
    public Main() {
        initComponents();
    }
}
```



```

apariencias=UIManager.getInstalledLookAndFeels();
try {
    UIManager.setLookAndFeel(apariencias[3].getClassName());
    SwingUtilities.updateComponentTreeUI(this);

    if((con = conexion.getConexionMYSQL2())==null){
        JOptionPane.showMessageDialog(null,"Error al conectarse
con el servidor");
    }
    selection = new Seleccionar();
    load = new Cargar();
    clasification = new Clasificar();
    recomnd = new Recomendar();
} catch (Exception e) {
    e.printStackTrace();
}finally{
    try{
        if(con!=null)
            con.close();
    }catch(Exception se){
        se.printStackTrace();
    }//end finally try
} //end try

}

/**
 * Código generado por el editor de interfaces de NetBeans.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    PanelPrincipal = new javax.swing.JTabbedPane();
    loadPanel = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    Field1 = new javax.swing.JTextField();
    Bot1 = new javax.swing.JButton();
    jLabel2 = new javax.swing.JLabel();
    Bot3 = new javax.swing.JButton();
    Field2 = new javax.swing.JTextField();
    Bot2 = new javax.swing.JButton();
    jButtonon1 = new javax.swing.JButton();
    clasifPanel = new javax.swing.JPanel();
    jLabel3 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox<>();
    jComboBox2 = new javax.swing.JComboBox<>();
    jComboBox3 = new javax.swing.JComboBox<>();
    jComboBox4 = new javax.swing.JComboBox<>();
    jComboBox5 = new javax.swing.JComboBox<>();
    jComboBox6 = new javax.swing.JComboBox<>();
    jComboBox7 = new javax.swing.JComboBox<>();
    jComboBox8 = new javax.swing.JComboBox<>();
    jComboBox9 = new javax.swing.JComboBox<>();
    jComboBox10 = new javax.swing.JComboBox<>();
    jComboBox11 = new javax.swing.JComboBox<>();

```

```

jComboBox12 = new javax.swing.JComboBox<>();
jComboBox13 = new javax.swing.JComboBox<>();
jComboBox14 = new javax.swing.JComboBox<>();
jComboBox15 = new javax.swing.JComboBox<>();
jComboBox16 = new javax.swing.JComboBox<>();
jButton2 = new javax.swing.JButton();
Field3 = new javax.swing.JTextField();
Field4 = new javax.swing.JTextField();
jLabel4 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
jLabel9 = new javax.swing.JLabel();
jLabel10 = new javax.swing.JLabel();
jLabel11 = new javax.swing.JLabel();
jLabel12 = new javax.swing.JLabel();
jLabel13 = new javax.swing.JLabel();
jLabel14 = new javax.swing.JLabel();
jLabel15 = new javax.swing.JLabel();
jLabel16 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
jLabel18 = new javax.swing.JLabel();
jLabel19 = new javax.swing.JLabel();
jLabel20 = new javax.swing.JLabel();
jLabel21 = new javax.swing.JLabel();
jButton3 = new javax.swing.JButton();
jComboBox17 = new javax.swing.JComboBox<>();
jLabel22 = new javax.swing.JLabel();
jComboBox18 = new javax.swing.JComboBox<>();
jButton4 = new javax.swing.JButton();
graphPanel = new javax.swing.JPanel();
jLabel24 = new javax.swing.JLabel();
jComboBox21 = new javax.swing.JComboBox<>();
jComboBox22 = new javax.swing.JComboBox<>();
jButton5 = new javax.swing.JButton();
drawPanel = new javax.swing.JPanel();

```

```

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("Sistema de Seguimiento");
setName("ProjectX"); // NOI18N

```

```

PanelPrincipal.setPreferredSize(new java.awt.Dimension(640,
500));

```

```

jLabel1.setText("Seleccione el archivo de encabezados");

```

```

Bot1.setText("Seleccionar");
Bot1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Seleccionar(evt);
    }
});

```

```

jLabel2.setText("Seleccione el archivo .csv con los datos");

```

```

Bot3.setText("Crear encabezados");
Bot3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        CrearEncabezados(evt);
    }
});

Bot2.setText("Seleccionar");
Bot2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Seleccionar2(evt);
    }
});

jButton1.setText("Llenar Tabla");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        LlenarTabla(evt);
    }
});

javax.swing.GroupLayout loadPanelLayout = new
javax.swing.GroupLayout(loadPanel);
loadPanel.setLayout(loadPanelLayout);
loadPanelLayout.setHorizontalGroup(

loadPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(loadPanelLayout.createSequentialGroup()
        .addGap(112, 112, 112)
        .addGroup(loadPanelLayout.createParallelGroup(javax.swing
.GroupLayout.Alignment.LEADING)
            .addComponent(jButton1)
            .addComponent(Bot3)
            .addComponent(jLabel2)
            .addComponent(jLabel1)
            .addGroup(loadPanelLayout.createSequentialGroup()
                .addGroup(loadPanelLayout.createParallelGroup(jav
ax.swing.GroupLayout.Alignment.TRAILING, false)
                    .addComponent(Field2,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(Field1,
javax.swing.GroupLayout.Alignment.LEADING))
                .addGap(32, 32, 32)
                .addGroup(loadPanelLayout.createParallelGroup(jav
ax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(Bot1)
                    .addComponent(Bot2))))
            .addGap(162, Short.MAX_VALUE))
    );
loadPanelLayout.setVerticalGroup(

loadPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(loadPanelLayout.createSequentialGroup()
        .addGap(63, 63, 63)

```

```

        .addComponent(jLabel1)
        .addGap(18, 18, 18)
        .addGroup(loadPanelLayout.createParallelGroup(javax.swing
.GroupLayout.Alignment.BASELINE)
            .addComponent(Field1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(Bot1))
        .addGap(25, 25, 25)
        .addComponent(Bot3)
        .addGap(80, 80, 80)
        .addComponent(jLabel2)
        .addGap(18, 18, 18)
        .addGroup(loadPanelLayout.createParallelGroup(javax.swing
.GroupLayout.Alignment.BASELINE)
            .addComponent(Field2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(Bot2))
        .addGap(25, 25, 25)
        .addComponent(jButton1)
        .addContainerGap(123, Short.MAX_VALUE))
);

PanelPrincipal.addTab("Carga de datos", loadPanel);

jLabel3.setText("Seleccione el tipo de atributos a considerar
para el clasificador:");

jComboBox1.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox2.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox3.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox4.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox5.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox6.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox7.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox8.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jComboBox9.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

```

```

        jComboBox10.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jComboBox11.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jComboBox12.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jComboBox13.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jComboBox14.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jComboBox15.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jComboBox16.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

        jButton2.setText("Clasificar");
        jButton2.setEnabled(false);
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                Clasificar(evt);
            }
        });

        jLabel4.setText("% Si Deserto");
        jLabel5.setText("% No deserto");

        jLabel6.setText("-----");
        jLabel7.setText("-----");
        jLabel8.setText("-----");
        jLabel9.setText("-----");
        jLabel10.setText("-----");
        jLabel11.setText("-----");
        jLabel12.setText("-----");
        jLabel13.setText("-----");
        jLabel14.setText("-----");
        jLabel15.setText("-----");
        jLabel16.setText("-----");
        jLabel17.setText("-----");

```

```

jLabel18.setText("-----");
jLabel19.setText("-----");
jLabel20.setText("-----");
jLabel21.setText("-----");

jButton3.setText("Limpiar");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Limpiar(evt);
    }
});

jComboBox17.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Academico", "Personal" }));
jComboBox17.setSelectedIndex(-1);
jComboBox17.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(java.awt.event.ItemEvent evt) {
        Seleccion(evt);
    }
});

jLabel22.setText("-----");

jComboBox18.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "No Considerar" }));

jButton4.setText("Mostrar Recomendaciones");
jButton4.setEnabled(false);
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Recomendacion(evt);
    }
});

javax.swing.GroupLayout clasifPanelLayout = new
javax.swing.GroupLayout(clasifPanel);
clasifPanel.setLayout(clasifPanelLayout);
clasifPanelLayout.setHorizontalGroup(

clasifPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(clasifPanelLayout.createSequentialGroup()
        .addGap(23, 23, 23)
        .addGroup(clasifPanelLayout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
            .addGroup(clasifPanelLayout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addGroup(clasifPanelLayout.createParallelGroup(javax.
avax.swing.GroupLayout.Alignment.LEADING, false)
                    .addComponent(jLabel6,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)

```

```

        .addComponent(jLabel7,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)
        .addComponent(jLabel8,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)
        .addComponent(jLabel9,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)
        .addComponent(jLabel10,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)
        .addComponent(jLabel11,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)
        .addComponent(jLabel12,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE)
        .addComponent(jLabel13,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 185, Short.MAX_VALUE))
        .addPreferredGap(javax.swing.LayoutStyle.Component
Placement.RELATED)
        .addGroup(classifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(jComboBox6,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox5,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox8,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox7,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

```

```

        .addGap(29, 29, 29)
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel14,
javax.swing.GroupLayout.PREFERRED_SIZE, 155,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel15,
javax.swing.GroupLayout.PREFERRED_SIZE, 155,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(clasifPanelLayout.createParallelGro
up(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jLabel17,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(jLabel18,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                        .addComponent(jLabel19,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel22,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(jLabel20,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                    .addComponent(jLabel16,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                        .addComponent(jLabel21,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 155,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addGroup(clasifPanelLayout.createSequentialGroup())
            .addComponent(jLabel3)
            .addGap(31, 31, 31)
            .addComponent(jComboBox17,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
clasifPanelLayout.createSequentialGroup())
            .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(jButton2,
javax.swing.GroupLayout.PREFERRED_SIZE, 159,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jButton4))
            .addGap(18, 18, 18)
            .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.LEADING)

```



```

        .addGroup(javax.swing.GroupLayout.Alignment.T
RAILING, clasifPanelLayout.createSequentialGroup())
        .addComponent(jLabel5)
        .addGap(18, 18, 18)
        .addComponent(Field4,
javax.swing.GroupLayout.PREFERRED_SIZE, 91,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.T
RAILING, clasifPanelLayout.createSequentialGroup())
        .addComponent(jLabel4)
        .addGap(18, 18, 18)
        .addComponent(Field3,
javax.swing.GroupLayout.PREFERRED_SIZE, 91,
javax.swing.GroupLayout.PREFERRED_SIZE))))))
        .addGap(27, 27, 27)
        .addGroup(clasifPanelLayout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.LEADING)
        .addComponent(jButton3,
javax.swing.GroupLayout.PREFERRED_SIZE, 76,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(clasifPanelLayout.createParallelGroup(javax
.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(jComboBox16,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox9,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox10,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox11,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox12,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox13,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox14,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox15,
javax.swing.GroupLayout.Alignment.TRAILING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jComboBox18,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(16, Short.MAX_VALUE))
    );
    clasifPanelLayout.setVerticalGroup(

clasifPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(clasifPanelLayout.createSequentialGroup())

```

```

        .addContainerGap ()
        .addGroup (clasifPanelLayout.createParallelGroup (javax.swing.
GroupLayout.Alignment.BASELINE)
            .addComponent (jLabel3)
            .addComponent (jComboBox17,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent (jButton3))
        .addPreferredGap (javax.swing.LayoutStyle.ComponentPlaceme
nt.UNRELATED)
        .addGroup (clasifPanelLayout.createParallelGroup (javax.swing.
GroupLayout.Alignment.LEADING)
            .addGroup (clasifPanelLayout.createSequentialGroup ()
                .addGroup (clasifPanelLayout.createParallelGroup (j
avax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent (jComboBox9,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent (jLabel14))
                    .addGap (18, 18, 18)
                    .addGroup (clasifPanelLayout.createParallelGroup (j
avax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent (jComboBox10,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent (jLabel15))
                        .addGap (18, 18, 18)
                        .addGroup (clasifPanelLayout.createParallelGroup (j
avax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent (jComboBox11,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent (jLabel16))
                            .addGap (18, 18, 18)
                            .addGroup (clasifPanelLayout.createParallelGroup (j
avax.swing.GroupLayout.Alignment.BASELINE)
                                .addComponent (jComboBox12,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addComponent (jLabel17))
                                .addGap (18, 18, 18)
                                .addGroup (clasifPanelLayout.createParallelGroup (j
avax.swing.GroupLayout.Alignment.BASELINE)
                                    .addComponent (jComboBox13,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                    .addComponent (jLabel18))
                                    .addGap (18, 18, 18)
                                    .addGroup (clasifPanelLayout.createParallelGroup (j
avax.swing.GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(jComboBox14,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel19))
        .addGap(18, 18, 18)
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBox15,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel20))
        .addGap(18, 18, 18)
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBox16,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel21))
        .addGap(18, 18, 18)
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBox18,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel22)))
        .addGroup(clasifPanelLayout.createSequentialGroup())
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel6))
        .addGap(18, 18, 18)
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBox2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel7,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBox3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel8))
        .addGap(18, 18, 18)

```

```

        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel9))
            .addGap(18, 18, 18)
            .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jComboBox5,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel10))
                .addGap(18, 18, 18)
                .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jComboBox6,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(jLabel11))
                    .addGap(18, 18, 18)
                    .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jComboBox7,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(jLabel12))
                        .addGap(18, 18, 18)
                        .addGroup(clasifPanelLayout.createParallelGroup(j
avax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(jComboBox8,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addComponent(jLabel13))))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlaceme
nt.RELATED, 28, Short.MAX_VALUE)
            .addGroup(clasifPanelLayout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.BASELINE)
                .addComponent(Field3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel4)
                    .addComponent(jButton2))
                .addGap(18, 18, 18)
                .addGroup(clasifPanelLayout.createParallelGroup(javax.swi
ng.GroupLayout.Alignment.BASELINE)
                    .addComponent(Field4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(jLabel5)

```

```

        .addComponent(jButton4))
        .addContainerGap()
    );

    PanelPrincipal.addTab("Clasificador de probabilidad",
clasifPanel);

    jLabel24.setText("Elija de las listas desplegadas las
estadísticas que desee graficar.");

    jComboBox21.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Deserto", "No deserto" }));

    jComboBox22.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Academico", "Personal" }));

    jButton5.setText("Graficar");
    jButton5.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Graficar(evt);
        }
    });

    drawPanel.setToolTipText("");

    javax.swing.GroupLayout drawPanelLayout = new
javax.swing.GroupLayout(drawPanel);
    drawPanel.setLayout(drawPanelLayout);
    drawPanelLayout.setHorizontalGroup(

drawPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(
            .addGap(0, 0, Short.MAX_VALUE)
        );
    drawPanelLayout.setVerticalGroup(

drawPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(
            .addGap(0, 386, Short.MAX_VALUE)
        );

    javax.swing.GroupLayout graphPanelLayout = new
javax.swing.GroupLayout(graphPanel);
    graphPanel.setLayout(graphPanelLayout);
    graphPanelLayout.setHorizontalGroup(

graphPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(
            .addGroup(graphPanelLayout.createSequentialGroup()
                .addGap(23, 23, 23)
                .addComponent(jLabel24)
                .addGap(18, 18, 18)
                .addComponent(jComboBox21,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)

```

```

        .addComponent (jComboBox22,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap (18, 18, 18)
        .addComponent (jButton5)
        .addContainerGap (19, Short.MAX_VALUE))
    .addGroup (graphPanelLayout.createSequentialGroup ())
        .addContainerGap ()
        .addComponent (drawPanel,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap ())
    );
    graphPanelLayout.setVerticalGroup (

graphPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup (graphPanelLayout.createSequentialGroup ())
            .addContainerGap ()
            .addGroup (graphPanelLayout.createParallelGroup (javax.swing.
GroupLayout.Alignment.BASELINE)
                .addComponent (jLabel24)
                .addComponent (jComboBox21,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent (jComboBox22,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent (jButton5))
            .addGap (18, 18, 18)
            .addComponent (drawPanel,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap (34, Short.MAX_VALUE))
    );

    PanelPrincipal.addTab ("Presentación de estadísticas",
graphPanel);

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

layout.createParallelGroup (javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent (PanelPrincipal,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );
    layout.setVerticalGroup (

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

```

```

        .addComponent(PanelPrincipal,
javafx.swing.GroupLayout.DEFAULT_SIZE,
javafx.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );

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

/**
 * Selecciona y obtiene la ruta del archivo de encabezados.
 */
private void Seleccionar(java.awt.event.ActionEvent evt) {
    JFileChooser fileChooser = new JFileChooser();
    fileChooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
    fileChooser.setAcceptAllFileFilterUsed(true);

    int rval = fileChooser.showOpenDialog(null);
    if (rval == JFileChooser.APPROVE_OPTION) {

Field1.setText(fileChooser.getSelectedFile().getAbsolutePath());
        filex = fileChooser.getSelectedFile().getAbsolutePath();

    }
}

/**
 * Selecciona y obtiene la ruta del archivo de datos.
 */
private void Seleccionar2(java.awt.event.ActionEvent evt) {
    JFileChooser fileChooser = new JFileChooser();
    fileChooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
    fileChooser.setAcceptAllFileFilterUsed(true);

    int rval = fileChooser.showOpenDialog(null);
    if (rval == JFileChooser.APPROVE_OPTION) {

Field2.setText(fileChooser.getSelectedFile().getAbsolutePath());
        filey = fileChooser.getSelectedFile().getAbsolutePath();
        table =
fileChooser.getSelectedFile().getName().split("\\.")[0];
    }
}

/**
 * Llama a la ejecución del método para crear encabezados de Cargar.
 */
private void CrearEncabezados(java.awt.event.ActionEvent evt) {
    load.encabezados(filex,table);
}

/**
 * Llama a la ejecución del método para llenar la tabla de Cargar.
 */
private void LlenarTabla(java.awt.event.ActionEvent evt) {
    load.llenado(filey,table);
}

```

```

}

/**
 * Obtiene el tipo de atributo seleccionado de la lista
 * desplegable principal y actualiza las etiquetas y las
 * listas para mostrar la información de los atributos.
 */
private void Seleccion(java.awt.event.ItemEvent evt) {
    JComboBox cb =(JComboBox)evt.getSource();
    String select = (String)cb.getSelectedItem();

    if (select.equalsIgnoreCase("Academico"))
    {
        jLabel16.setText("Género");
        jLabel17.setText("Edad de ingreso");
        jLabel18.setText("Examen de admisión");
        jLabel19.setText("Promedio de secundaria");
        jLabel110.setText("Tipo de secundaria");
        jLabel111.setText("Cambios de escuela");
        jLabel112.setText("Tiempos de traslado");
        jLabel113.setText("Razón de ingreso");
        jLabel114.setText("Promedio de asistencia");
        jLabel115.setText("Promedio de último año");
        jLabel116.setText("Porcentaje de aprobación");
        jLabel117.setText("Aprovechamiento");
        jLabel118.setText("Beca");
        jLabel119.setText("Apoyo de la escuela");
        jLabel120.setText("Problemas con compañeros");
        jLabel121.setText("Problemas escolares");
        Field3.setText("");
        Field4.setText("");
        jLabel122.setVisible(false);
        JComboBox18.setVisible(false);

        JComboBox1.removeAllItems();
        JComboBox2.removeAllItems();
        JComboBox3.removeAllItems();
        JComboBox4.removeAllItems();
        JComboBox5.removeAllItems();
        JComboBox6.removeAllItems();
        JComboBox7.removeAllItems();
        JComboBox8.removeAllItems();
        JComboBox9.removeAllItems();
        JComboBox10.removeAllItems();
        JComboBox11.removeAllItems();
        JComboBox12.removeAllItems();
        JComboBox13.removeAllItems();
        JComboBox14.removeAllItems();
        JComboBox15.removeAllItems();
        JComboBox16.removeAllItems();

        JComboBox1.addItem("No Considerar");
        JComboBox1.addItem("Mujer");
        JComboBox1.addItem("Hombre");

        JComboBox2.addItem("No Considerar");
        JComboBox2.addItem("Atraso");
    }
}

```



```
jComboBox2.addItem("Normal");

jComboBox3.addItem("No Considerar");
jComboBox3.addItem("Bajo");
jComboBox3.addItem("Medio");
jComboBox3.addItem("Alto");

jComboBox4.addItem("No Considerar");
jComboBox4.addItem("Bajo");
jComboBox4.addItem("Medio");
jComboBox4.addItem("Alto");

jComboBox5.addItem("No Considerar");
jComboBox5.addItem("Ambas");
jComboBox5.addItem("Privada");
jComboBox5.addItem("Publica");

jComboBox6.addItem("No Considerar");
jComboBox6.addItem("Ninguna");
jComboBox6.addItem("Una");
jComboBox6.addItem("Dos_mas");

jComboBox7.addItem("No Considerar");
jComboBox7.addItem("Bajo");
jComboBox7.addItem("Medio");
jComboBox7.addItem("Alto");

jComboBox8.addItem("No Considerar");
jComboBox8.addItem("Academica");
jComboBox8.addItem("Asignacion");
jComboBox8.addItem("Economica");
jComboBox8.addItem("Personal");

jComboBox9.addItem("No Considerar");
jComboBox9.addItem("Bajo");
jComboBox9.addItem("Medio");
jComboBox9.addItem("Alto");

jComboBox10.addItem("No Considerar");
jComboBox10.addItem("Bajo");
jComboBox10.addItem("Medio");
jComboBox10.addItem("Alto");

jComboBox11.addItem("No Considerar");
jComboBox11.addItem("Bajo");
jComboBox11.addItem("Medio");
jComboBox11.addItem("Alto");

jComboBox12.addItem("No Considerar");
jComboBox12.addItem("Bajo");
jComboBox12.addItem("Medio");
jComboBox12.addItem("Alto");

jComboBox13.addItem("No Considerar");
jComboBox13.addItem("No");
jComboBox13.addItem("Si");
```

```

jComboBox14.addItem("No Considerar");
jComboBox14.addItem("No");
jComboBox14.addItem("Si");

jComboBox15.addItem("No Considerar");
jComboBox15.addItem("No");
jComboBox15.addItem("Si");

jComboBox16.addItem("No Considerar");
jComboBox16.addItem("No");
jComboBox16.addItem("Si");

}else if(select.equalsIgnoreCase("Personal"))
{
jLabel16.setText("Género");
jLabel17.setText("Vivía con");
jLabel18.setText("Apoyo de familiares");
jLabel19.setText("Amigos abandonaron");
jLabel110.setText("Hermanos abandonaron");
jLabel111.setText("Consumo de cigarro");
jLabel112.setText("Consumo de alcohol");
jLabel113.setText("Consumo de drogas");
jLabel114.setText("Acoso");
jLabel115.setText("Problemas en casa");
jLabel116.setText("Tiene familia propia");
jLabel117.setText("Falta de dinero");
jLabel118.setText("Baja autoestima");
jLabel119.setText("Enfermedad grave");
jLabel20.setText("Poco gusto por estudiar");
jLabel21.setText("Otros familiares estudiaron");
jLabel22.setText("Padres con preparatoria");
Field3.setText("");
Field4.setText("");
jLabel22.setVisible(true);
jComboBox18.setVisible(true);

jComboBox1.removeAllItems();
jComboBox2.removeAllItems();
jComboBox3.removeAllItems();
jComboBox4.removeAllItems();
jComboBox5.removeAllItems();
jComboBox6.removeAllItems();
jComboBox7.removeAllItems();
jComboBox8.removeAllItems();
jComboBox9.removeAllItems();
jComboBox10.removeAllItems();
jComboBox11.removeAllItems();
jComboBox12.removeAllItems();
jComboBox13.removeAllItems();
jComboBox14.removeAllItems();
jComboBox15.removeAllItems();
jComboBox16.removeAllItems();
jComboBox18.removeAllItems();

jComboBox1.addItem("No Considerar");
jComboBox1.addItem("Mujer");
jComboBox1.addItem("Hombre");

```

```
jComboBox2.addItem("No Considerar");
jComboBox2.addItem("Solo");
jComboBox2.addItem("Directos");
jComboBox2.addItem("Pareja");
jComboBox2.addItem("Amigos");
jComboBox2.addItem("Otros");

jComboBox3.addItem("No Considerar");
jComboBox3.addItem("No");
jComboBox3.addItem("Si");

jComboBox4.addItem("No Considerar");
jComboBox4.addItem("No");
jComboBox4.addItem("Si");

jComboBox5.addItem("No Considerar");
jComboBox5.addItem("No");
jComboBox5.addItem("Si");

jComboBox6.addItem("No Considerar");
jComboBox6.addItem("No");
jComboBox6.addItem("Bajo");
jComboBox6.addItem("Medio");
jComboBox6.addItem("Alto");

jComboBox7.addItem("No Considerar");
jComboBox7.addItem("No");
jComboBox7.addItem("Bajo");
jComboBox7.addItem("Medio");
jComboBox7.addItem("Alto");

jComboBox8.addItem("No Considerar");
jComboBox8.addItem("No");
jComboBox8.addItem("Bajo");

jComboBox9.addItem("No Considerar");
jComboBox9.addItem("No");
jComboBox9.addItem("Si");

jComboBox10.addItem("No Considerar");
jComboBox10.addItem("No");
jComboBox10.addItem("Si");

jComboBox11.addItem("No Considerar");
jComboBox11.addItem("Ninguno");
jComboBox11.addItem("Matrimonio");
jComboBox11.addItem("Embarazo");
jComboBox11.addItem("Ambos");

jComboBox12.addItem("No Considerar");
jComboBox12.addItem("No");
jComboBox12.addItem("Si");

jComboBox13.addItem("No Considerar");
jComboBox13.addItem("No");
jComboBox13.addItem("Si");
```

```

        jComboBox14.addItem("No Considerar");
        jComboBox14.addItem("No");
        jComboBox14.addItem("Si");

        jComboBox15.addItem("No Considerar");
        jComboBox15.addItem("No");
        jComboBox15.addItem("Si");

        jComboBox16.addItem("No Considerar");
        jComboBox16.addItem("No");
        jComboBox16.addItem("Si");

        jComboBox18.addItem("No Considerar");
        jComboBox18.addItem("Ninguno");
        jComboBox18.addItem("Madre");
        jComboBox18.addItem("Padre");
        jComboBox18.addItem("Ambos");
    }
    jButton2.setEnabled(true);
}

/**
 * Limpia todos los valores de los atributos y los
 * regresa a No Considerar, también borra los campos
 * de texto.
 */
private void Limpiar(java.awt.event.ActionEvent evt) {
    jComboBox1.setSelectedIndex(0);
    jComboBox2.setSelectedIndex(0);
    jComboBox3.setSelectedIndex(0);
    jComboBox4.setSelectedIndex(0);
    jComboBox5.setSelectedIndex(0);
    jComboBox6.setSelectedIndex(0);
    jComboBox7.setSelectedIndex(0);
    jComboBox8.setSelectedIndex(0);
    jComboBox9.setSelectedIndex(0);
    jComboBox10.setSelectedIndex(0);
    jComboBox11.setSelectedIndex(0);
    jComboBox12.setSelectedIndex(0);
    jComboBox13.setSelectedIndex(0);
    jComboBox14.setSelectedIndex(0);
    jComboBox15.setSelectedIndex(0);
    jComboBox16.setSelectedIndex(0);
    jComboBox18.setSelectedIndex(0);
    Field3.setText("");
    Field4.setText("");
}

/**
 * Manda a llamar el método para clasificar de la clase
 * Clasificar, distingue entre académico y personal.
 */
private void Clasificar(java.awt.event.ActionEvent evt) {
    table = jComboBox17.getSelectedItem().toString().toLowerCase();
    if(table.equalsIgnoreCase("Academico"))
    {

```

```

        clasificar_acad();
    }else if(table.equalsIgnoreCase("Personal"))
    {
        clasificar_pers();
    }
    jButton4.setEnabled(true);
}

/**
 * Manda a llamar el método para recomendar de la clase
 * Recomendar, distingue entre académico y personal,
 * también selecciona los valores de los atributos que
 * se utilizaran para la recomendación y los envia como
 * parametro.
 */
private void Recomendacion(java.awt.event.ActionEvent evt) {
    String select = jComboBox17.getSelectedItem().toString();
    recomnd.startUp(select);
    if(select.equalsIgnoreCase("academico"))
    {
        String asistencia =
jComboBox9.getSelectedItem().toString().toUpperCase();
        String aprobacion =
jComboBox11.getSelectedItem().toString().toUpperCase();
        String aprovechamiento =
jComboBox12.getSelectedItem().toString().toUpperCase();
        String beca =
jComboBox13.getSelectedItem().toString().toUpperCase();
        String apoyo =
jComboBox14.getSelectedItem().toString().toUpperCase();
        String problemas_comp =
jComboBox15.getSelectedItem().toString().toUpperCase();
        String problemas_esc =
jComboBox16.getSelectedItem().toString().toUpperCase();

        recomnd.Recommendation_acad(si_acad, no_acad,
total_datos_acad, total_desert_acad, total_final_acad,
deserto_acad,finalizo_acad,asistencia,aprobacion,aprovechamiento,beca,apo
yo,problemas_comp,problemas_esc);
    }else
    {
        String cigarro =
jComboBox6.getSelectedItem().toString().toUpperCase();
        String alcohol =
jComboBox7.getSelectedItem().toString().toUpperCase();
        String drogas =
jComboBox8.getSelectedItem().toString().toUpperCase();
        String acoso =
jComboBox9.getSelectedItem().toString().toUpperCase();
        String problemas_casa =
jComboBox10.getSelectedItem().toString().toUpperCase();
        String autoestima =
jComboBox13.getSelectedItem().toString().toUpperCase();
        String gusto_est =
jComboBox15.getSelectedItem().toString().toUpperCase();
    }
}

```

```

        recomnd.Recommendation_pers(si_pers, no_pers,
total_datos_pers, total_desert_pers, total_final_pers,
deserto_pers,finalizo_pers,cigarro,alcohol,drogas,acoso,problemas_casa,au
toestima,gusto_est);
    }
    jButton4.setEnabled(false);
    recomnd.setVisible(true);

}

/**
 * Llama al método para obtener los valores que se utilizaran
 * para la gráfica de barras y después crea dicha gráfica.
 */
private void Graficar(java.awt.event.ActionEvent evt) {
    String pers_acad =
jComboBox22.getSelectedItem().toString().toLowerCase();
    String des_fin =
jComboBox21.getSelectedItem().toString().toLowerCase();
    String si_no;
    if(des_fin.equalsIgnoreCase("deserto"))
    {
        si_no = "SI";
    }else
        si_no = "NO";

    drawPanel.removeAll();
    Graficar datos = new Graficar();
    DefaultCategoryDataset dataset =
datos.datosBarra(pers_acad,si_no);

    JFreeChart barChart = ChartFactory.createStackedBarChart(
"Datos de tipo "+pers_acad.toUpperCase()+" donde el estudiante "
+des_fin.toUpperCase(),
"Factores", "Valor",
dataset,PlotOrientation.HORIZONTAL,true,true,true);

    ChartPanel cp = new ChartPanel(barChart);
    cp.setVisible(true);
    Dimension dim = drawPanel.getSize();

    cp.setBounds(2, 2, dim.width, dim.height);
    drawPanel.add(cp);
    repaint();
}

/**
 * Método que realiza el conteo de todos los valores necesarios
 * para el clasificador, después manda a llamar al método para
 * clasificar de la clase Clasificar de tipo académico.
 */
private void clasificar_acad()
{
    total_datos_acad = selection.total_datos(table);
    total_desert_acad = selection.total_sino("SI",table);
    total_final_acad = selection.total_sino("NO",table);
}

```

```

        if(jComboBox1.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[0]=1;
            no_acad[0]=1;
        }else{
            si_acad[0] =
selection.conteo(0,jComboBox1.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[0] =
selection.conteo(0,jComboBox1.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox2.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[1]=1;
            no_acad[1]=1;
        }else{
            si_acad[1] =
selection.conteo(1,jComboBox2.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[1] =
selection.conteo(1,jComboBox2.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox3.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[2]=1;
            no_acad[2]=1;
        }else{
            si_acad[2] =
selection.conteo(2,jComboBox3.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[2] =
selection.conteo(2,jComboBox3.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox4.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[3]=1;
            no_acad[3]=1;
        }else{
            si_acad[3] =
selection.conteo(3,jComboBox4.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[3] =
selection.conteo(3,jComboBox4.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

```

```

        if(jComboBox5.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[4]=1;
            no_acad[4]=1;
        }else{
            si_acad[4] =
selection.conteo(4,jComboBox5.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[4] =
selection.conteo(4,jComboBox5.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox6.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[5]=1;
            no_acad[5]=1;
        }else{
            si_acad[5] =
selection.conteo(5,jComboBox6.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[5] =
selection.conteo(5,jComboBox6.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox7.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[6]=1;
            no_acad[6]=1;
        }else{
            si_acad[6] =
selection.conteo(6,jComboBox7.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[6] =
selection.conteo(6,jComboBox7.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox8.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[7]=1;
            no_acad[7]=1;
        }else{
            si_acad[7] =
selection.conteo(7,jComboBox8.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[7] =
selection.conteo(7,jComboBox8.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

```



```

        if(jComboBox9.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[8]=1;
            no_acad[8]=1;
        }else{
            si_acad[8] =
selection.conteo(8,jComboBox9.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_acad;
            no_acad[8] =
selection.conteo(8,jComboBox9.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_acad;
        }

        if(jComboBox10.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[9]=1;
            no_acad[9]=1;
        }else{
            si_acad[9] =
selection.conteo(9,jComboBox10.getSelectedItem().toString().toUpperCase()
,"SI",table)/total_desert_acad;
            no_acad[9] =
selection.conteo(9,jComboBox10.getSelectedItem().toString().toUpperCase()
,"NO",table)/total_final_acad;
        }

        if(jComboBox11.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[10]=1;
            no_acad[10]=1;
        }else{
            si_acad[10] =
selection.conteo(10,jComboBox11.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_acad;
            no_acad[10] =
selection.conteo(10,jComboBox11.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_acad;
        }

        if(jComboBox12.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[11]=1;
            no_acad[11]=1;
        }else{
            si_acad[11] =
selection.conteo(11,jComboBox12.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_acad;
            no_acad[11] =
selection.conteo(11,jComboBox12.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_acad;
        }

```

```

        if(jComboBox13.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[12]=1;
            no_acad[12]=1;
        }else{
            si_acad[12] =
selection.conteo(12,jComboBox13.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_acad;
            no_acad[12] =
selection.conteo(12,jComboBox13.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_acad;
        }

        if(jComboBox14.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[13]=1;
            no_acad[13]=1;
        }else{
            si_acad[13] =
selection.conteo(13,jComboBox14.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_acad;
            no_acad[13] =
selection.conteo(13,jComboBox14.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_acad;
        }

        if(jComboBox15.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[14]=1;
            no_acad[14]=1;
        }else{
            si_acad[14] =
selection.conteo(14,jComboBox15.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_acad;
            no_acad[14] =
selection.conteo(14,jComboBox15.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_acad;
        }

        if(jComboBox16.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
        {
            si_acad[15]=1;
            no_acad[15]=1;
        }else{
            si_acad[15] =
selection.conteo(15,jComboBox16.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_acad;
            no_acad[15] =
selection.conteo(15,jComboBox16.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_acad;
        }

```

```

        deserto_acad = clasificacion.c_acad(si_acad, no_acad,
total_datos_acad, total_desert_acad, total_final_acad, "si");
        finalizo_acad = clasificacion.c_acad(si_acad, no_acad,
total_datos_acad, total_desert_acad, total_final_acad, "no");
        Field3.setText(deserto_acad*100+"%");
        Field4.setText(finalizo_acad*100+"%");
    }

/**
 * Método que realiza el conteo de todos los valores necesarios
 * para el clasificador, después manda a llamar al método para
 * clasificar de la clase Clasificar de tipo personal.
 */
private void clasificar_pers ()
{

    total_datos_pers = selection.total_datos(table);
    total_desert_pers = selection.total_sino("SI",table);
    total_final_pers = selection.total_sino("NO",table);

    if(jComboBox1.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[0]=1;
        no_pers[0]=1;
    }else{
        si_pers[0] =
selection.conteo(0,jComboBox1.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[0] =
selection.conteo(0,jComboBox1.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox2.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[1]=1;
        no_pers[1]=1;
    }else{
        si_pers[1] =
selection.conteo(1,jComboBox2.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[1] =
selection.conteo(1,jComboBox2.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox3.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[2]=1;
        no_pers[2]=1;
    }else{
        si_pers[2] =
selection.conteo(2,jComboBox3.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;

```

```

        no_pers[2] =
selection.conteo(2,jComboBox3.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox4.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[3]=1;
        no_pers[3]=1;
    }else{
        si_pers[3] =
selection.conteo(3,jComboBox4.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[3] =
selection.conteo(3,jComboBox4.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox5.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[4]=1;
        no_pers[4]=1;
    }else{
        si_pers[4] =
selection.conteo(4,jComboBox5.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[4] =
selection.conteo(4,jComboBox5.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox6.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[5]=1;
        no_pers[5]=1;
    }else{
        si_pers[5] =
selection.conteo(5,jComboBox6.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[5] =
selection.conteo(5,jComboBox6.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox7.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[6]=1;
        no_pers[6]=1;
    }else{
        si_pers[6] =
selection.conteo(6,jComboBox7.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;

```

```

        no_pers[6] =
selection.conteo(6,jComboBox7.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox8.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[7]=1;
        no_pers[7]=1;
    }else{
        si_pers[7] =
selection.conteo(7,jComboBox8.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[7] =
selection.conteo(7,jComboBox8.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox9.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[8]=1;
        no_pers[8]=1;
    }else{
        si_pers[8] =
selection.conteo(8,jComboBox9.getSelectedItem().toString().toUpperCase(),
"SI",table)/total_desert_pers;
        no_pers[8] =
selection.conteo(8,jComboBox9.getSelectedItem().toString().toUpperCase(),
"NO",table)/total_final_pers;
    }

    if(jComboBox10.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[9]=1;
        no_pers[9]=1;
    }else{
        si_pers[9] =
selection.conteo(9,jComboBox10.getSelectedItem().toString().toUpperCase(),
,"SI",table)/total_desert_pers;
        no_pers[9] =
selection.conteo(9,jComboBox10.getSelectedItem().toString().toUpperCase(),
,"NO",table)/total_final_pers;
    }

    if(jComboBox11.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[10]=1;
        no_pers[10]=1;
    }else{
        si_pers[10] =
selection.conteo(10,jComboBox11.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;

```

```

        no_pers[10] =
selection.conteo(10,jComboBox11.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    if(jComboBox12.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[11]=1;
        no_pers[11]=1;
    }else{
        si_pers[11] =
selection.conteo(11,jComboBox12.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;
        no_pers[11] =
selection.conteo(11,jComboBox12.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    if(jComboBox13.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[12]=1;
        no_pers[12]=1;
    }else{
        si_pers[12] =
selection.conteo(12,jComboBox13.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;
        no_pers[12] =
selection.conteo(12,jComboBox13.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    if(jComboBox14.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[13]=1;
        no_pers[13]=1;
    }else{
        si_pers[13] =
selection.conteo(13,jComboBox14.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;
        no_pers[13] =
selection.conteo(13,jComboBox14.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    if(jComboBox15.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[14]=1;
        no_pers[14]=1;
    }else{
        si_pers[14] =
selection.conteo(14,jComboBox15.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;

```

```

        no_pers[14] =
selection.conteo(14,jComboBox15.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    if(jComboBox16.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[15]=1;
        no_pers[15]=1;
    }else{
        si_pers[15] =
selection.conteo(15,jComboBox16.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;
        no_pers[15] =
selection.conteo(15,jComboBox16.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    if(jComboBox18.getSelectedItem().toString().equalsIgnoreCase("No
considerar"))
    {
        si_pers[16]=1;
        no_pers[16]=1;
    }else{
        si_pers[16] =
selection.conteo(16,jComboBox18.getSelectedItem().toString().toUpperCase(
),"SI",table)/total_desert_pers;
        no_pers[16] =
selection.conteo(16,jComboBox18.getSelectedItem().toString().toUpperCase(
),"NO",table)/total_final_pers;
    }

    deserto_pers = clasification.c_pers(si_pers, no_pers,
total_datos_pers, total_desert_pers, total_final_pers,"si");
    finalizo_pers = clasification.c_pers(si_pers, no_pers,
total_datos_pers, total_desert_pers, total_final_pers, "no");
    Field3.setText(deserto_pers*100+"%");
    Field4.setText(finalizo_pers*100+"%");
}

public static void main(String args[]) {
    /* Crea y muestra la interfaz */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Main().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton Bot1;
private javax.swing.JButton Bot2;
private javax.swing.JButton Bot3;
private javax.swing.JTextField Field1;
private javax.swing.JTextField Field2;

```

```

private javax.swing.JTextField Field3;
private javax.swing.JTextField Field4;
private javax.swing.JTabbedPane PanelPrincipal;
private javax.swing.JPanel clasifPanel;
private javax.swing.JPanel drawPanel;
private javax.swing.JPanel graphPanel;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JComboBox<String> jComboBox1;
private javax.swing.JComboBox<String> jComboBox10;
private javax.swing.JComboBox<String> jComboBox11;
private javax.swing.JComboBox<String> jComboBox12;
private javax.swing.JComboBox<String> jComboBox13;
private javax.swing.JComboBox<String> jComboBox14;
private javax.swing.JComboBox<String> jComboBox15;
private javax.swing.JComboBox<String> jComboBox16;
private javax.swing.JComboBox<String> jComboBox17;
private javax.swing.JComboBox<String> jComboBox18;
private javax.swing.JComboBox<String> jComboBox2;
private javax.swing.JComboBox<String> jComboBox21;
private javax.swing.JComboBox<String> jComboBox22;
private javax.swing.JComboBox<String> jComboBox3;
private javax.swing.JComboBox<String> jComboBox4;
private javax.swing.JComboBox<String> jComboBox5;
private javax.swing.JComboBox<String> jComboBox6;
private javax.swing.JComboBox<String> jComboBox7;
private javax.swing.JComboBox<String> jComboBox8;
private javax.swing.JComboBox<String> jComboBox9;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel19;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel20;
private javax.swing.JLabel jLabel21;
private javax.swing.JLabel jLabel22;
private javax.swing.JLabel jLabel24;
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.JLabel jLabel9;
private javax.swing.JPanel loadPanel;
// End of variables declaration
}

```