

Universidad Autónoma Metropolitana
Unidad Azcapotzalco
División de Ciencias Básicas e Ingeniería
Licenciatura en Ingeniería en Computación

Presentación de estadísticas académicas con base en criterios de ingreso utilizando técnicas de visualización de información.

Modalidad: Proyecto Tecnológico.

Trimestre 2020-Invierno

Brenda Rodríguez Sánchez
Matrícula: 2143002128
al2143002128@azc.uam.mx

Mtro. Josué Figueroa González
Profesor asociado
Departamento de Sistemas
jfgo@azc.uam.mx

17 de agosto de 2020

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, Brenda Rodríguez Sánchez, 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.

Resumen

El desempeño de los alumnos a lo largo de su vida universitaria depende de muchos factores, como pueden ser la edad o la escuela de nivel medio superior de donde egresaron, entre otros. Cada alumno tiene un perfil de ingreso a la universidad diferente, sin embargo, encontrar posibles patrones en sus características de ingreso y su relación con su integración y desempeño en el ambiente universitario permite a la universidad prever dificultades y ofrecer las herramientas necesarias para que el alumno refuerce o desarrolle ciertas características que le permitan adentrarse al ambiente universitario.

El objetivo de este proyecto es poder analizar la información que se tiene de los alumnos de una manera más fácil, ya que la información esta disponible en la universidad en una base de datos, sin embargo, es difícil analizar y encontrar patrones en una tabla constituida por múltiples filas y columnas, de ahí la importancia de una representación gráfica que permita la visualización de los datos de una manera comprensible.

Para cumplir con el objetivo del proyecto se desarrolló una aplicación web que permite al usuario elegir al menos un criterio de entrada (edad, sexo, escuela de procedencia, promedio de educación media superior y/o puntaje obtenido en el examen de ingreso a la UAM), a partir del cual el sistema nos va a generar una salida grafica en la cual se puede visualizar si el alumno finalizó o no, el tiempo que le llevó finalizar la carrera o el tiempo que lleva activo.

Tabla de contenido

Introducción	1
Antecedentes	2
Proyectos terminales	2
Proyectos externos	2
Justificación	4
Objetivos	5
Objetivo general	5
Objetivos específicos	5
Marco teórico	6
Visualización de datos	6
Chart.js	6
jsPDF	6
Base de datos	7
Desarrollo del proyecto	9
Módulo de selección de criterios de salida y entrada	10
Módulo de obtención y procesamiento de datos	10
Módulo de generación de gráficas	10
Módulo de exportación	
Resultados	14
Conclusiones	17
Referencias bibliográficas	18
Apéndice A. Código fuente módulo de selección de criterios de entrada y salida	19
Apéndice B. Código fuente módulo de obtención y procesamiento de datos	33
Apéndice C. Código fuente módulo de generación de gráficas	55
Apéndice D. Código fuente módulo de exportación de gráficas	6

Índice de figuras

Figura 1. Componentes del sistema desarrollado	9
Figura 2. Interfaz del sistema	11
Figura 3. Selección de criterios	14
Figura 4. Gráfica de barras - Tiempo de finalización	14
Figura 5. Gráfica de área – Tiempo de finalización	15
Figura 6. Gráfica de barras – Tiempo activo	15
Figura 7. Gráfica de área – Tiempo activo	16
Figura 8. Gráfica radar – El alumno finalizó o no	16

Introducción

Tweets, clics, fotos, likes, blogs, transacciones online y mucha más información es generada en la web en tan solo 24 horas, se estima que se superan los cinco exabytes de información por día en el mundo [1], sin embargo, la utilidad de esta puede ser poca e incluso nula si no se logra extraer solo aquella que es de gran importancia o aporta alguna diferencia relevante en aquello que se busca demostrar o analizar.

Esta generación de información se presenta en el ámbito educativo en varios aspectos, el desempeño de los alumnos a lo largo de su vida universitaria depende de muchos factores, como pueden ser la edad o hasta la escuela de nivel medio superior de donde egresaron. Cada alumno tiene un perfil de ingreso a la universidad diferente, sin embargo, el encontrar posibles patrones en las características de ingreso y su relación con su integración y desempeño en el ambiente universitario, permite a la universidad prever dificultades y ofrecer las herramientas necesarias para que el alumno refuerce o desarrolle ciertas características que le permitan adentrarse al ambiente universitario.

Es común que empresas, escuelas, instituciones e incluso personas se encuentren con la necesidad de analizar una gran cantidad de información que la mayoría de las veces se encuentra en una base de datos o en algún formato que no permite un análisis claro y fácil de la información que se tiene a nuestra disposición, lo cual puede provocar que datos relevantes pasen desapercibidos.

De aquí la importancia de herramientas y tecnologías de visualización de datos. La visualización de datos es la representación gráfica de información y datos de una manera comprensible para el usuario permitiendo así la asimilación de información compleja y la percepción de patrones, tendencias y valores atípicos que de otra manera hubieran pasado desapercibidos.

Este proyecto permite visualizar información académica de los alumnos considerando sus características de ingreso a la universidad Autónoma Metropolitana Azcapotzalco (UAM-A) utilizando una aplicación web.

Antecedentes

Proyectos terminales

Sistema para la visualización de factores de abandono escolar a partir de estadísticas educativas [2]

Desarrollo de un sistema web que facilita la consulta a través de representaciones gráficas de información académica, laboral y personal de los estudiantes de Nivel Medio Superior del país, con la intención que pueda servir para la toma de decisiones que lleven a la disminución de la deserción escolar en este nivel de estudios.

Sistema para la visualización de estadísticas relacionadas con la seriación de Unidades de Enseñanza y aprendizaje [3]

Desarrollo de una aplicación Java que, partiendo de una base de datos, facilita el observar la relación entre el desempeño escolar de una UEA y los resultados académicos de la o las UEA seriadas anteriores a través de la visualización de información, en este caso haciendo uso de gráficas de barras.

Aplicación para la consulta de información escolar para asesores y educandos del Instituto Nacional de Educación para los Adultos (AMINEA) [4]

Desarrollo de una aplicación móvil donde el asesor tiene acceso a la información que requiere sobre sus grupos de estudio, horarios, e información de sus educandos, en cualquier momento sin necesidad de estar frente a un monitor.

Factores de Riesgo y Rendimiento Académico en Estudiantes del Nivel Medio Superior (Licenciatura) [5]

Es una tesis de licenciatura de la Universidad Autónoma del Estado de México que consta de una investigación que tiene como objetivo relacionar los factores de riesgo (Dimensión familiar, personal, social, salud y escolar) y el rendimiento académico en estudiantes de Nivel Medio Superior.

Proyectos externos

RAW Graphs [6]

Es un framework de visualización de datos de código abierto que tiene como objetivo hacer que la representación visual de datos complejos sea fácil para todos. Permite pegar la información, subir un documento (TSV, CSV, DSV, JSON y Excel) o introducir una url para obtener los datos a partir de los cuales se generará la gráfica que el usuario seleccione de acuerdo con el filtrado que el mismo haya elegido, dicha gráfica puede ser descargada en archivo tipo svg, png o json.

DataWrapper [7]

Es una herramienta que permite crear gráficas interactivas (barras, líneas, donut, etc.), mapas y tablas a partir de archivos en formato cvs, Excel y Google Sheets. La versión gratuita permite crear las gráficas y descargarlas en ficheros .zip, si se quiere incrustar los gráficos en alguna página web se debe pagar algún plan de los que ofrece.

Justificación

Si bien la UAM-A cuenta con información acerca de las diferentes características de ingreso de los alumnos, no es fácil encontrar patrones, ya que esta se encuentra en un archivo con miles de filas y varias columnas que no son fáciles de procesar, por lo tanto, la creación de un sistema que permita la visualización de estos datos es de suma importancia.

El generar gráficas que representen los datos que son de interés para un estudio como este permite entender mejor las estadísticas con una imagen que intentar analizar un archivo de Excel o una base de datos conformado de múltiples columnas y filas.

Con el proyecto se obtiene una representación visual de los datos de los alumnos, lo cual permite identificar patrones o características de manera rápida y sencilla que podrán llevar a la toma de decisiones.

Objetivos

Objetivo General

Desarrollar un sistema que permita presentar estadísticas académicas con base en criterios de ingreso utilizando técnicas de visualización de información.

Objetivos Específicos

- Diseñar e implementar un módulo que permita la selección de criterios de entrada y salida.
- Diseñar e implementar un módulo de obtención y procesamiento de datos.
- Diseñar e implementar un módulo que permita la generación de gráficas.
- Diseñar e implementar un módulo que permita exportar los datos de salida.

Marco teórico

Visualización de datos

La visualización de datos es la disciplina que estudia cómo utilizar la percepción visual para comunicar y analizar datos [8].

En ocasiones una persona, una empresa o una institución se ve en la necesidad de analizar una inmensa cantidad de información que le permita tomar decisiones o encontrar la naturaleza de algún problema, sin embargo, la información puede estar disponible en hojas de cálculo, interminables tablas o gigantes bases de datos que poco nos pueden decir ya que no permiten encontrar relaciones, sacar conclusiones o mantener nuestro interés en dicha información.

La visualización de datos es una representación gráfica que nos permite observar grandes cantidades de datos de una forma rápida y sencilla identificando así patrones, grupos, brechas o valores atípicos que nos ayudan a la toma de decisiones y a la comprensión y transmisión de nuevos conceptos.

Chart.js

Es una librería en JavaScript que permite crear diferentes tipos de gráficas. Se trata de una librería de código abierto que acepta contribuciones de la comunidad. Su uso permite visualizar los datos en 8 diferentes tipos de gráficas que pueden ser personalizadas y animadas.

Es muy fácil de utilizar, para poder usarla en nuestros proyectos podemos descargar su última versión desde GitHub o incluirla en nuestro archivo HTML utilizando CND.

Cuenta con una gran documentación y ejemplos en su página de internet [9].

jsPDF

Se trata de una librería en JavaScript de código abierto, desarrollada por una agencia digital llamada Parallax. Nos permite crear archivos PDF a partir de HTML, para poder utilizarla podemos utilizar npm o CND, si preferimos descargarla desde la página solo debemos contestar algunas preguntas como nombre y correo electrónico, entre otras, una vez que hemos contestado las preguntas la página nos proporciona un documento zip en el cual podemos encontrar el archivo js que debemos añadir a nuestro proyecto así como diferentes ejemplos que nos ayudan a entender el funcionamiento de la librería.

Base de datos

Una base de datos es una colección de datos lógicamente coherente con algún tipo de significado inherente que están relacionados y representan algún aspecto del mundo real, esta se diseña, construye y rellena para un propósito específico [10].

Para la realización de este proyecto se utilizó una base de datos proporcionada por la UAM-A en la cual se encuentran datos personales y escolares de todos los alumnos.

Dado que el objetivo de este proyecto es poder visualizar estadísticas académicas con base en criterios de ingreso, los campos utilizados fueron los siguientes:

- **Edad:** Años cumplidos al ingreso
- **Esc:** Escuela de procedencia
- **Prom:** Promedio del bachillerato
- **Puntaje:** Puntaje en el examen de ingreso
- **Sexo**
- **Edo:** Estado Académico

Edo = 1: Inscrito a UEA

Alumno o participante que cubrió los requisitos necesarios para quedar inscritos a una o varias unidades de enseñanza-aprendizaje.

Edo = 2: No reinscrito

Alumno sin carga académica.

Edo = 4: Baja definitiva

Recurso que tiene el alumno para solicitar su baja voluntaria en la Universidad.

Edo = 5: Titulado

De acuerdo con el RESUAM ya no tiene la calidad de alumno. Concluyó los créditos del plan de estudios, solicitó su certificado de licenciatura y además, cumplió con todos los requisitos establecidos para obtener el título profesional correspondiente.

Edo = 6: Egresado

Estudiante que ya concluyó sus créditos del plan de estudios y además solicitó su certificado de estudios de licenciatura en la UAM. Todavía no se titula.

Edo = 7: Baja reglamentaria

Es la pérdida de la calidad de alumno. Las causales se dividen en dos grupos: las que permiten reingresar y las que no lo permiten. Entre las causales que permiten reingresar están violación de ciclos y baja definitiva (siempre que ésta se haya realizado dentro de los primeros cinco años de estar como alumno). Si con su primera matrícula un alumno incurrió en violación de ciclos, basta que solicite la baja definitiva para que pueda presentar nuevamente el examen de selección. Dentro de las causales que no permiten reingresar están la baja reglamentaria por presentar un certificado de bachillerato falso, para efectos de ingreso, agotar el plazo máximo para terminar los estudios u obtener 5 NA's en una misma UEA.

Edo = 8: Créditos en revisión

Egresado potencial; es decir, es el alumno activo que está por concluir los créditos del plan de estudios. Se le pone este estado para no afectar su historial académico, cuando su plan de estudios se encuentra en un proceso de conversión.

Edo = 10: Inscrito sin carga académica

Alumno que hizo el trámite de reinscripción sin “carga” de UEA. Lo anterior, para tener derecho a utilizar los servicios de biblioteca, actividades deportivas y salas de cómputo, etc.

Edo = 12: Créditos cubiertos

Estudiante que ya concluyó los créditos del plan de estudios. No ha solicitado su certificado de estudios de licenciatura en la UAM, ni se ha titulado.

- **Edo2:** Segundo estado del aspirante o del alumno

Edo2 = 22: Éxito (Edo = 5, 6, 8 o 12)

Alumno que concluyó con su carrera. En este grupo se encuentran el titulado, egresado, créditos en revisión (potencialmente egresado) y el que tiene créditos cubiertos.

- **Ntri:** Número natural de trimestres en la UAM. Todos los trimestres transcurridos desde que ingresó hasta que tuvo su último trimestre de reinscripción.

Los primeros cinco campos representan los criterios de ingreso de los alumnos tomados en cuenta para este estudio, mientras que los últimos campos nos ayudan a saber si el alumno ya finalizó (Edo2= 22), sigue activo (Edo = 1,2 o 10) o ya no puede terminar la carrera (Edo=4 y 7) y los trimestres que le llevo terminar la carrera, que lleva activo o que duró según corresponda.

Desarrollo del proyecto

Para la creación del proyecto se utilizó el lenguaje de programación PHP para la comunicación con la base de datos, el lenguaje de hojas de estilo CSS, el lenguaje de marcas de hipertexto HTML, el lenguaje de programación JavaScript y la librería Charts.js para el diseño de la interfaz y la librería jsPDF para la exportación a documento PDF y como editor de código Visual Studio Code 1.47.3.

Como gestor de base de datos se ha utilizado MySQL y como servidor web Apache 2.4.43, todo esto a través de la aplicación XAMPP en su versión 7.4.5.

En la Figura 1 se puede observar el diagrama de componentes del sistema desarrollado.

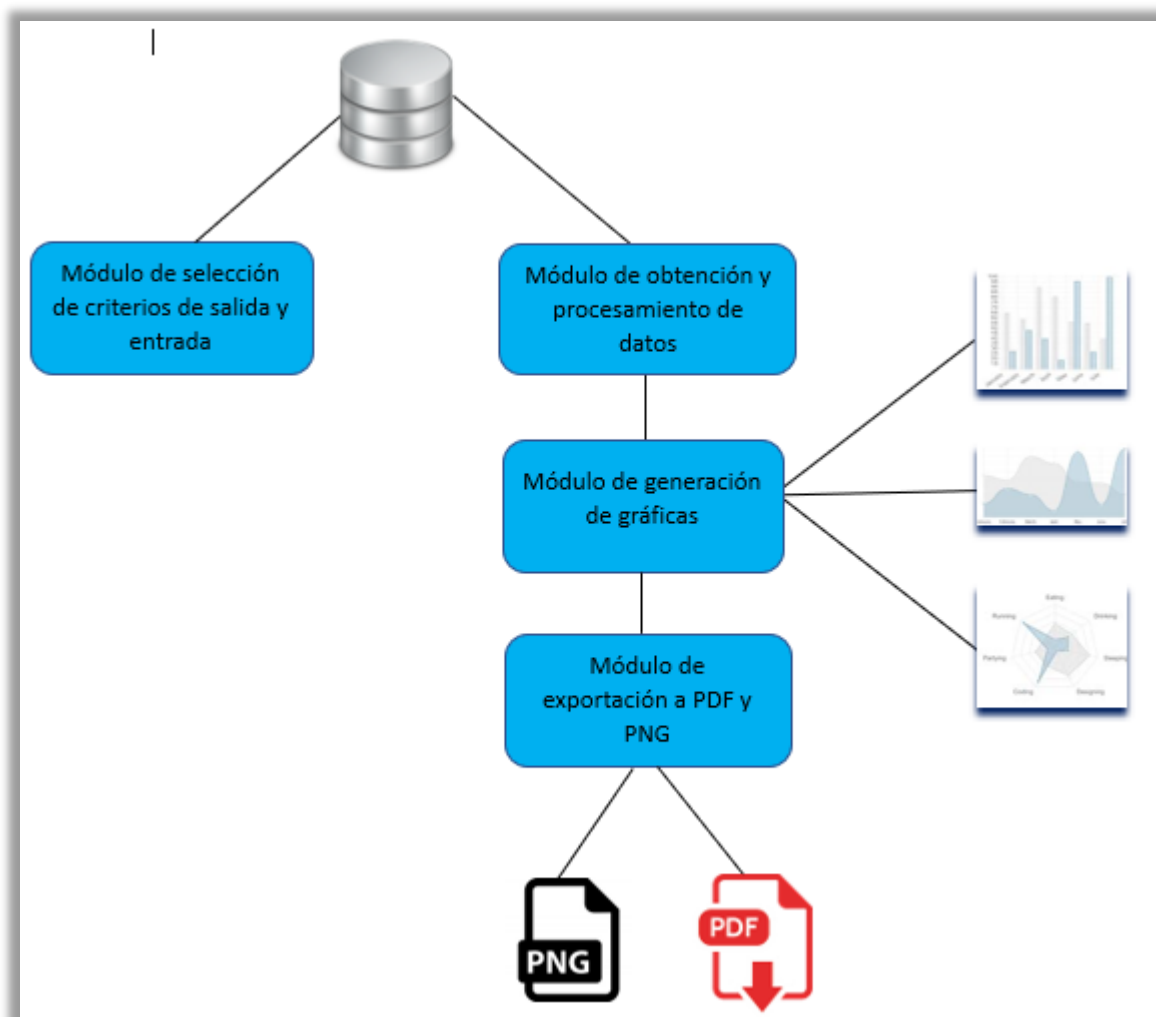


Figura 1 Componentes del sistema desarrollado.

Módulo de selección de criterios de salida y entrada.

Este módulo muestra en pantalla las opciones que se tienen como criterios de entrada y salida y permite que el usuario seleccione aquellos que sean más convenientes según el filtrado que este requiera.

Como entradas, se especificará que criterios o combinación de ellos se usarán para generar una gráfica, los criterios incluyen: edad de ingreso, promedio del nivel medio superior, puntaje en el examen de ingreso, escuela de procedencia y género.

Módulo de obtención y procesamiento de datos.

Este módulo se encarga de hacer la consulta a la base de datos de acuerdo con los criterios de entrada que el usuario haya elegido, de igual forma se encarga de devolver la información en el formato adecuado para la creación de la gráfica.

Módulo de generación de gráficas.

El objetivo de este módulo es mostrar la información de forma visual por medio de una gráfica según lo que haya seleccionado el usuario.

Módulo de exportación.

Este módulo permite exportar la gráfica generada a uno de los siguientes dos formatos, PDF o un formato de imagen, en este caso PNG.

Implementación

Módulo de selección de criterios de salida y entrada.

Este módulo es la interfaz entre el usuario y el sistema, para su creación se utilizó HTML, JavaScript y CSS, este último utilizando el framework w3css (w3.css), los archivos que lo componen son: index.html, validarDatos.js y estadoInicial.js los cuales se pueden ver en el apéndice A.

A través de un formulario, el cual podemos visualizar en la Figura 2, el usuario elige los diferentes criterios que quiere visualizar, una vez que el usuario da clic en el botón Aceptar los datos son validados con JavaScript, el objetivo de esta validación es asegurar que el usuario seleccione al menos un criterio de entrada con al menos una de sus posibles opciones y un criterio de salida, de no ser así se muestra el error correspondiente en la pantalla, en caso contrario el formulario es deshabilitado y los datos son enviados al módulo de obtención y procesamiento de datos utilizando el método POST.

Las consultas, leyenda de la gráfica y el título se construyen de acuerdo con los criterios seleccionados en el siguiente orden: edad, promedio, puntaje, sexo, escuela.

Por ejemplo, si queremos visualizar si el alumno finalizó o no para aquellos hombres menores de 18 años que obtuvieron un promedio mayor a 8.5.

Primero procesamos la edad, se crea un arreglo con las consultas llamado `consultasEdad` y un arreglo llamado `letrerosEdad`, estos arreglos quedan de la siguiente manera:

```
consultasEdad[0] = Edad = 16
```

```
consultasEdad[1] = Edad = 17
```

```
letrerosEdad[0] = 16
```

```
letrerosEdad[1] = 17
```

Después procesamos el promedio casi de la misma forma que la edad:

```
consultasPromedio[0] = Prom between 8.51 and 9
```

```
consultasPromedio[1] = Prom between 8.01 and 9.5
```

```
consultasPromedio[1] = Prom between 9.51 and 10
```

```
letrerosPromedio[0] = [8.51 – 9]
```

```
letrerosPromedio[1] = [9.01 – 9.5]
```

```
letrerosPromedio[2] = [9.51 – 10]
```

Finalmente, procesamos el sexo:

```
consultasSexo[0] = Sexo = 'M'
```

```
letrerosSexo [0] = M
```

Una vez hecho esto, el paso final es concatenar cada elemento del arreglo `consultasEdad` con cada elemento del arreglo `consultasPromedio` y a su vez con los de `consultasSexo`, de esta forma creamos las diferentes consultas y de la misma forma con los arreglos cuyo prefijo es `letreros` para crear la leyenda de la gráfica. Lo anterior daría como resultado las diferentes consultas a realizar a la base de datos, es decir, para la primera consulta requerimos `Edad=16`, `Prom between 8.51 and 9` y `sexo = 'M'` y su letrado correspondiente sería `16 [8.51 – 9] M`.

En este caso el título de la gráfica es `Tiempo de finalización` y la bandera es 0, lo cual indica que el módulo de generación de gráficas debe mostrar una gráfica de barras y una de área, lo mismo sería en el caso de que el criterio de salida seleccionado sea `Tiempo activo del alumno`, en caso de que se seleccione `El alumno finalizó o no`, la bandera es 1, lo que indica al sistema que la gráfica debe ser de radar.

Módulo de generación de gráficas.

Este módulo está constituido por los siguientes archivos: Chart.min.js, utils.js, Chart.min.css y crearGrafica.js, este último puede consultarse en el apéndice C. Utilizando la librería Chart.js este módulo puede generar gráficas de barras, de área o de radar lo cual depende del criterio de salida que se haya seleccionado, si el usuario eligió tiempo de finalización o tiempo activo del alumno las gráficas mostradas serán de barras y de área mientras que si el usuario seleccionó el alumno finalizó o no, la gráfica mostrada será de radar.

Módulo de exportación.

Este módulo está compuesto por los archivos jspdf.min.js (librería jsPDF) y exportarGrafica.js el cual se encuentra en el apéndice D. Para exportar la gráfica a una imagen PNG es suficiente con crear un objeto Blob, sin embargo, para poder exportar a PDF utilizamos la librería jsPDF.

El usuario puede descargar la gráfica en formato de imagen PNG o en formato de archivo PDF para poder usarla en un futuro.

Resultados

El objetivo de poder visualizar estadísticas académicas con base en criterios de ingreso se cumple como se puede ver en los siguientes ejemplos.

Se quiere visualizar el tiempo de finalización de hombres que ingresaron a los 18 años y obtuvieron un promedio mayor a 8.5 en el bachillerato (Figura 3).

Selecciona los criterios de entrada a utilizar:

☒ Edad de ingreso

☐ Menores de 18 años
☒ 18 años
☐ de más de 18 a 20 años
☐ de más de 20 a 23 años
☐ de más de 23 a 26 años
☐ Más de 26 años
☐ Todas las edades

☐ Escuela de procedencia

☐ UNAM ☐ IPN
☐ DGETI ☐ Colegio de Bachilleres
☐ CONALEP ☐ UAEM
☐ Universidad Estatal ☐ Normal Primaria
☐ Incorporada a la SEP ☐ Otra
☐ No definida ☐ Todas las escuelas

☒ Genero

Masculino

☐ Puntaje en el examen de ingreso a la UAM

☐ De 500 a 750
☐ De más de 750 a 850
☐ Más de 850
☐ Todos los puntajes

☒ Promedio de bachillerato

☐ De menos de 7 a 7.5
☐ De más de 7.5 a 8.5
☒ Más de 8.5
☐ Todos los promedios

Selecciona el criterio de salida:

☐ El alumno finalizó o nó
 ☒ Tiempo de finalización del alumno
 ☐ Tiempo activo del alumno

Aceptar

Figura 3 Selección de criterios.

Como resultado podemos visualizar las siguientes gráficas (Figura 4 y 5):

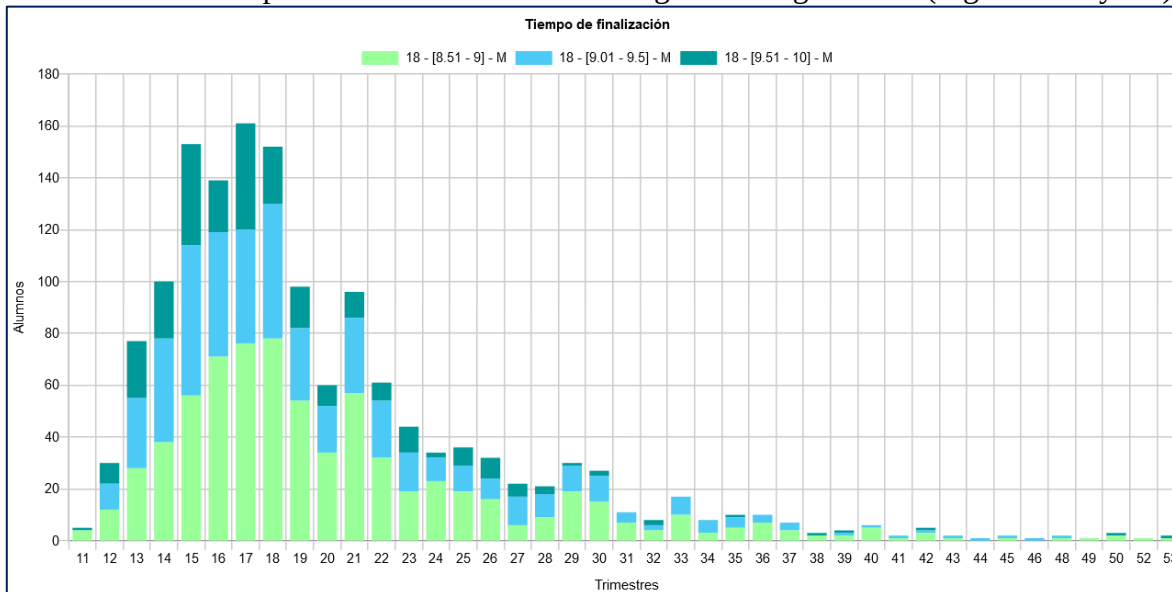


Figura 4 Gráfica de barras – Tiempo de finalización.

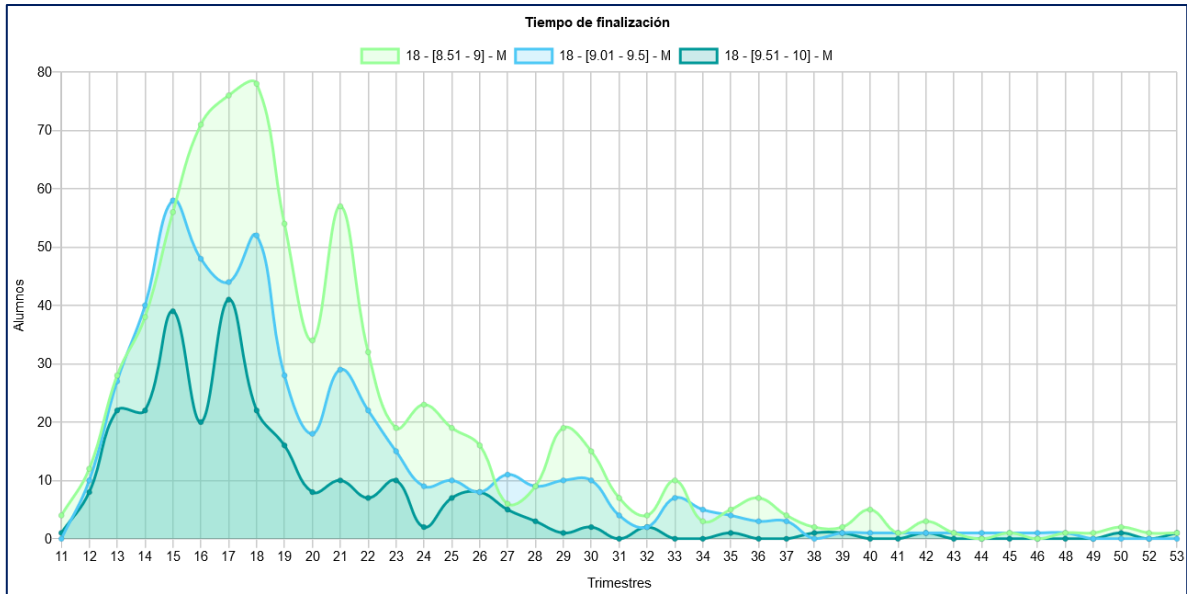


Figura 5 Gráfica de área – Tiempo de finalización.

Si el criterio de salida es tiempo activo en lugar de tiempo de finalización obtenemos el mismo tipo de gráficas (Figuras 6 y 7):

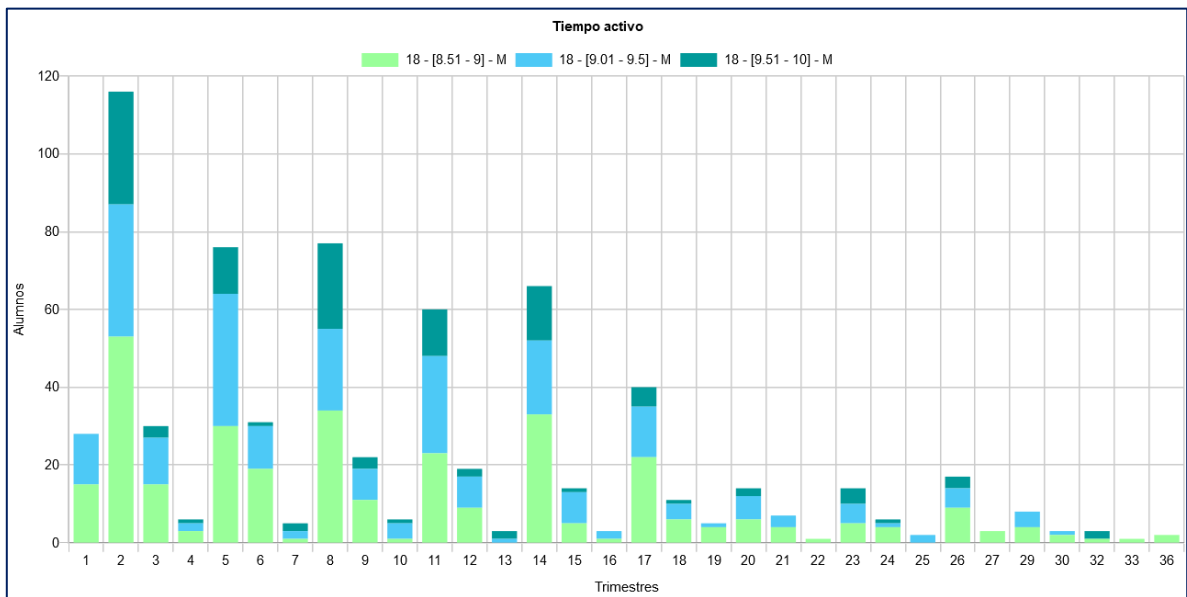


Figura 6 Gráfica de barras - Tiempo activo.

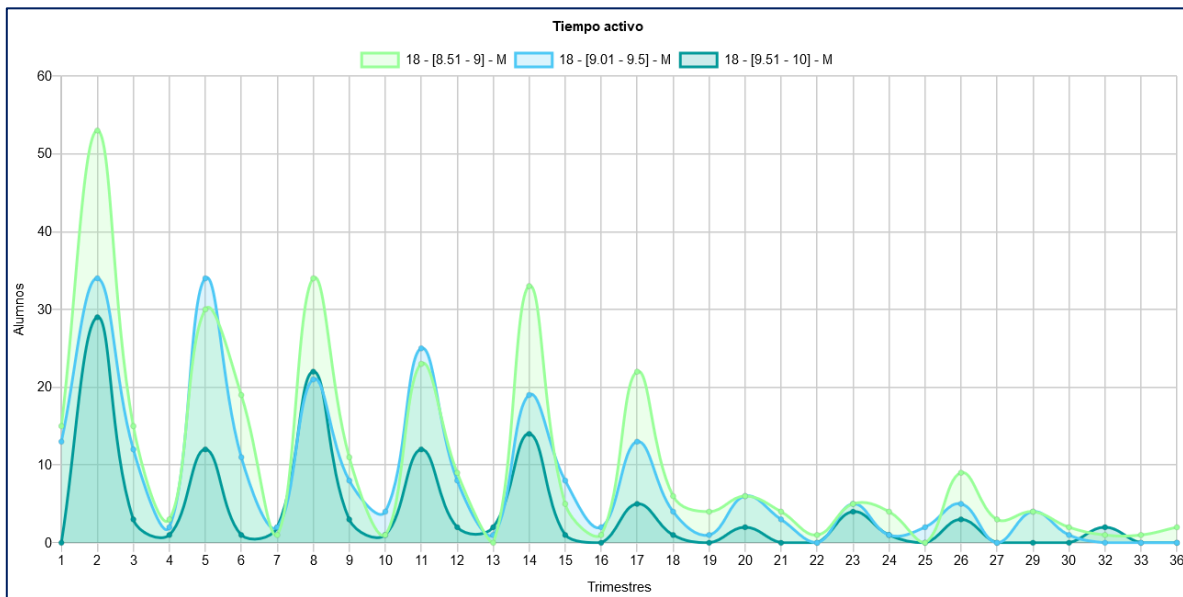


Figura 7 Gráfica de área - Tiempo activo.

Si, por el contrario, seleccionamos como criterio de salida el alumno finalizó o no, el resultado es una gráfica de radar que utiliza una escala logarítmica base 10 (Figura 8).

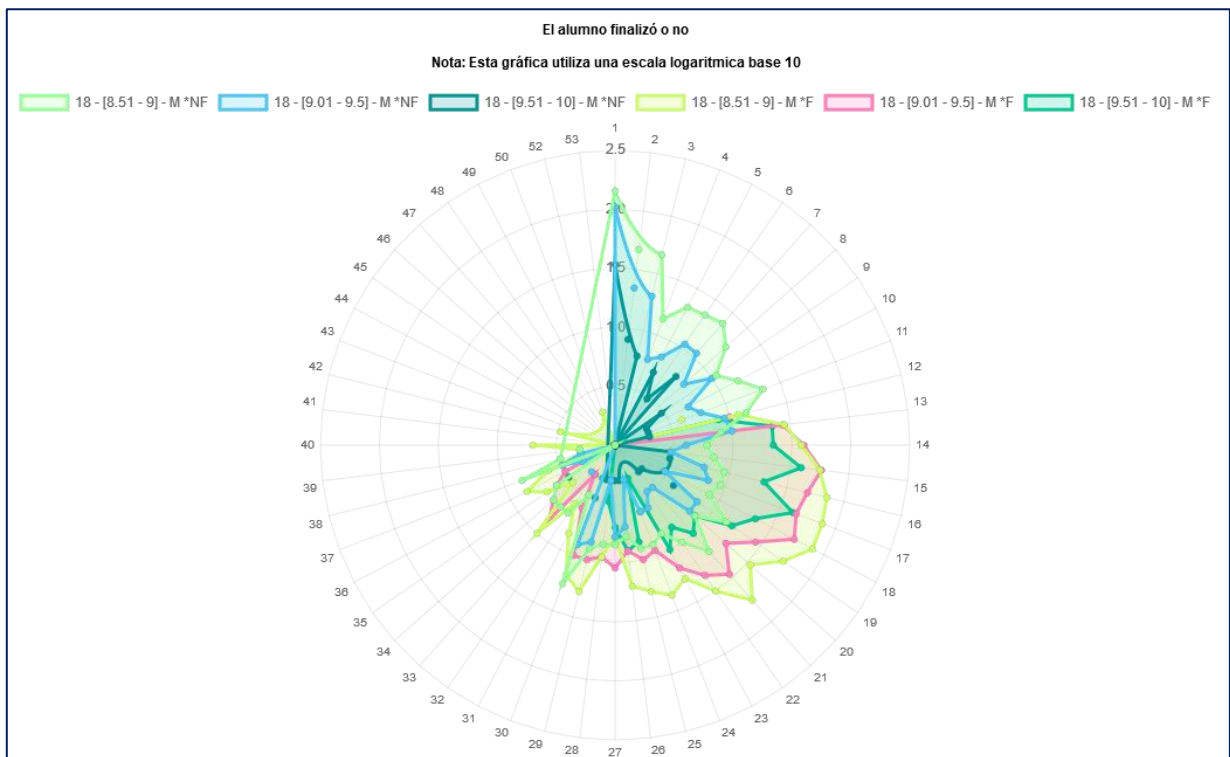


Figura 8 Gráfica radar - El alumno finalizó o no.

Conclusiones

El objetivo general del proyecto se cumplió ya que el sistema desarrollado permite visualizar estadísticas académicas con base en criterios de ingreso, esto gracias a que se logró diseñar e implementar los módulos que lo conforman, los cuales permiten la interacción del usuario con el sistema y la comunicación del sistema con la base de datos, así como la exportación de las gráficas.

La utilización de la librería Chart.js fue de gran ayuda en la elaboración del proyecto ya que representa una manera fácil y sencilla de crear y visualizar gráficos en HTML, fue de crucial importancia su uso ya que gracias a esta herramienta se pudo lograr el objetivo general del proyecto.

De igual manera, la librería jsPDF permitió que la exportación de las gráficas a un documento PDF fuera mucho más fácil y rápido, lo que nos permitió cumplir con el objetivo de la creación de un módulo que permitiera al usuario disponer de las gráficas que el sistema genera.

Referencias bibliográficas

- [1] Forbes, “How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read” [En línea]
Disponible: <https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#1333991160ba>
- [2] O. A. Rosa Remedios “Sistema para la visualización de factores de abandono escolar a partir de estadísticas educativas,” Proyecto terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, México, 2018.
- [3] A. de la Rosa Esteban “Sistema para la visualización de estadísticas relacionadas con la seriación de Unidades de Enseñanza y aprendizaje,” Proyecto terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, México, 2018.
- [4] B. Cruz Hernández y Y. G. Osio Soto “Aplicación para la consulta de información escolar para asesores y educandos del Instituto Nacional de Educación para los Adultos (AMINEA),” Proyecto terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, México, 2018.
- [5] N. López Hernández y R. M. Rosas Sánchez, “Factores de Riesgo y Rendimiento Académico en Estudiantes del Nivel Medio Superior (Licenciatura),” Tesis de Licenciatura, Universidad Autónoma del Estado de México, Facultad de Ciencias de la Conducta. México, 2014.
- [6] RAWGraphs: A Visualisation Platform to Create Open Outputs (2017) [En línea].
Disponible: <https://rawgraphs.io/>
- [7] DataWrapper Versión 4 (2019) [En línea] Disponible: <https://www.datawrapper.de>
- [8] Rininsland, Ae., Heydt, M., & Navarro Castillo, P. (2016). D3.js: Cutting-edge Data Visualization. Packt Publishing.
- [9] Chart.js Versión 2.0 (2020) [En línea] Disponible: <https://www.chartjs.org/>
- [10] Navathe, S. B. y B. Navathe, S. (2007). Fundamentos de sistemas de bases de datos (5a. ed.). Madrid, Mexico: Pearson Educación.
- [11] Cobo, A., Gómez, P., Pérez, D., & Rocha, R. (2005). PHP y MySQL Tecnologías para el desarrollo de aplicaciones web. Díaz Santos.
- [12] Lopez, J. (2011). Domine JavaScript (3ª ed.). RA – MA Editorial.

Apéndice A. Código fuente módulo de selección de criterios de entrada y salida

index.html

```
<!DOCTYPE html>
<html lang="es">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Criterios de Entrada</title>
  <link rel="stylesheet" href="css/w3.css">
  <script src="js/Chart.min.js"></script>
  <script src="js/Utils.js"></script>
  <script type="text/javascript"
src="js/obtenerYprocesarDatos.js"></script>
  <script type="text/javascript" src="js/jspdf.min.js"></script>
  <script type="text/javascript" src="js/estadoInicial.js"></script>
</head>

<body>
  <div id="CriteriosEntrada" class="criterioDeEntrada w3-container">
    <h3 class="w3-mipink w3-padding-small">Selecciona los criterios
de entrada a utilizar:</h3>
    <label id="ErrorCE" class="w3-padding-small w3-bottombar w3-
border w3-border-mired w3-round-large">Selecciona al
menos un criterio de entrada</label>
    <form id="formulario">
      <div class="w3-half">
        <div class="w3-panel w3-pale-red w3-round-large"
id="SeleccionEdad">
          <p>
            <input type="checkbox" id="Edad" name="Edad"
onchange="criteriosCh('Edad','checksEdad') ">
            <label for="Edad"> Edad de ingreso </label>
            <label id="ErrorEdad"
class="w3-padding-small w3-bottombar w3-
border w3-border-mired w3-round-large">Selecciona al
menos un rango de edad</label>
            <fieldset>
              <input type="checkbox" id="REdad1" name="REdad1"
class="checksEdad" disabled
onchange="ocultarMensajeDeError('ErrorEdad') ">
              <label for="REdad1"> Menores de 18 años</label>
<br>
              <input type="checkbox" id="REdad2" name="REdad2"
class="checksEdad" disabled
onchange="ocultarMensajeDeError('ErrorEdad') ">
              <label for="REdad2"> 18 años</label> <br>
              <input type="checkbox" id="REdad3" name="REdad3"
class="checksEdad" disabled
```



```

onchange="ocultarMensajeDeError('ErrorEdad')">
    <label for="REdad3"> de más de 18 a 20
años</label> <br>

    <input type="checkbox" id="REdad4" name="REdad4"
class="checksEdad" disabled

onchange="ocultarMensajeDeError('ErrorEdad')">
    <label for="REdad4">de más de 20 a 23
años</label> <br>

    <input type="checkbox" id="REdad5" name="REdad5"
class="checksEdad" disabled

onchange="ocultarMensajeDeError('ErrorEdad')">
    <label for="REdad5">de más de 23 a 26
años</label> <br>

    <input type="checkbox" id="REdad6" name="REdad6"
class="checksEdad" disabled

onchange="ocultarMensajeDeError('ErrorEdad')">
    <label for="REdad6">Más de 26 años </label> <br>

    <input type="checkbox" id="EdadT" name="EdadT"
class="checksEdad" disabled

onchange="seleccionarTodo('EdadT','checksEdad','ErrorEdad')">
    <label for="EdadT">Todas las edades</label> <br>
</fieldset>
</p>
</div>
<div class="EscuelaDeProcedencia w3-panel w3-pale-red
w3-round-large">
    <p>
        <input type="checkbox" id="Escuela"
name="Escuela"

onchange="criteriosCh('Escuela','checksEscuela')">
        <label for="Escuela"> Escuela de
procedencia</label>

        <label id="ErrorEscuela"
class="w3-padding-small w3-bottombar w3-
border w3-border-mired w3-round-large">Selecciona al
menos una escuela</label>
        <fieldset>
            <table>
                <tr>
                    <td>
                        <input type="checkbox"
class="checksEscuela" id="UNAM" name="UNAM" disabled

onchange="ocultarMensajeDeError('ErrorEscuela')">
                        <label for="UNAM">UNAM</label> <br>
                    </td>
                    <td>

```

```

                                <input type="checkbox"
class="checksEscuela" id="IPN" name="IPN" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="IPN">IPN</label> <br>
                                </td>
                                </tr>
                                <tr>
                                <td>
                                <input type="checkbox"
class="checksEscuela" id="DGETI" name="DGETI" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="DGETI">DGETI</label> <br>
                                </td>
                                <td>
                                <input type="checkbox"
class="checksEscuela" id="CBACH" name="CBACH" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="CBACH">Colegio de
Bachilleres</label> <br>
                                </td>
                                </tr>
                                <tr>
                                <td>
                                <input type="checkbox"
class="checksEscuela" id="CONALEP" name="CONALEP" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="CONALEP">CONALEP</label>
<br>
                                </td>
                                <td>
                                <input type="checkbox"
class="checksEscuela" id="UAEM" name="UAEM" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="UAEM">UAEM</label> <br>
                                </td>
                                </tr>
                                <tr>
                                <td>
                                <input type="checkbox"
class="checksEscuela" id="UNIVEST" name="UNIVEST" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="UNIVEST">Universidad
Estatad </label> <br>
                                </td>
                                <td>
                                <input type="checkbox"
class="checksEscuela" id="NORMAL" name="NORMAL" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                                <label for="NORMAL">Normal
Primaria</label> <br>
                                </td>

```

```

        </tr>
        <tr>
            <td>
                <input type="checkbox"
class="checksEscuela" id="SEP" name="SEP" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                <label for="SEP">Incorporada a la
SEP</label> <br>
            </td>
            <td>
                <input type="checkbox"
class="checksEscuela" id="OTRA" name="OTRA" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                <label for="OTRA">Otra</label> <br>
            </td>
        </tr>
        <tr>
            <td>
                <input type="checkbox"
class="checksEscuela" id="ND" name="ND" disabled
onchange="ocultarMensajeDeError('ErrorEscuela') ">
                <label for="ND">No definida</label>
<br>
            </td>
            <td>
                <input type="checkbox"
class="checksEscuela" id="EscT" name="EscT" disabled
onchange="seleccionarTodo('EscT','checksEscuela','ErrorEscuela') ">
                <label for="EscT">Todas las
escuelas</label> <br>
            </td>
        </tr>
    </table>
</fieldset>
</p>
</div>
</div>
<div class="w3-half w3-container">
    <!-- <div class="w3-half"> -->
    <div class="Genero w3-panel w3-pale-red w3-round-large">
        <p>
            <input type="checkbox" id="Sexo" name="Sexo"
onchange="sexoCh() ">
            <label for="Sexo"> Genero</label>
            <label id="ErrorGenero"
class="w3-padding-small w3-bottombar w3-
border w3-border-mired w3-round-large">Selecciona el
genero</label>
            <fieldset>
                <select name="Genero" value="Seleccionar"
id="Genero" disabled
onchange="ocultarMensajeDeError('ErrorGenero') ">
                    <option value="">Seleccionar</option>

```

```

        <option value="M">Masculino</option>
        <option value="F">Femenino</option>
        <option value="A">Ambos</option>
    </select>
</fieldset>
</p>
</div>
<div class="Puntaje w3-panel w3-pale-red w3-round-
large">
    <p>
        <input type="checkbox" id="Puntaje"
name="Puntaje"
onchange="criteriosCh('Puntaje','checksPuntaje')">
        <label for="Puntaje"> Puntaje en el examen de
ingreso a la UAM</label>
        <label id="ErrorPuntaje"
class="w3-padding-small w3-bottombar w3-
border w3-border-mired w3-round-large">Selecciona al
menos un rango</label>
        <fieldset>
            <input type="checkbox" class="checksPuntaje"
id="PuntajeMin" name="PuntajeMin" disabled
onchange="ocultarMensajeDeError('ErrorPuntaje')">
            <label for="PuntajeMin">De 500 a 750</label> <br>
            <input type="checkbox" class="checksPuntaje"
id="PuntajeMed" name="PuntajeMed" disabled
onchange="ocultarMensajeDeError('ErrorPuntaje')">
            <label for="PuntajeMed">De más de 750 a
850</label> <br>
            <input type="checkbox" class="checksPuntaje"
id="PuntajeMax" name="PuntajeMax" disabled
onchange="ocultarMensajeDeError('ErrorPuntaje')">
            <label for="PuntajeMax">Más de 850</label> <br>
            <input type="checkbox" class="checksPuntaje"
id="PuntajeT" name="PuntajeT" disabled
onchange="seleccionarTodo('PuntajeT','checksPuntaje','ErrorPuntaje')">
            <label for="PuntajeT">Todos los puntajes</label>
<br>
        </fieldset>
    </p>
</div>
<div class="Promedio w3-panel w3-border w3-pale-red w3-
round-large">
    <p>
        <input type="checkbox" id="Promedio"
name="Promedio"
onchange="criteriosCh('Promedio','checksPromedio')">
        <label for="Promedio"> Promedio de
bachillerato</label>

```

```

        <label id="ErrorPromedio"
            class="w3-padding-small w3-bottombar w3-
border w3-border-mired w3-round-large">Selecciona al
            menos un rango</label>
        <fieldset>
            <input type="checkbox" class="checksPromedio"
id="PromMin" name="PromMin" disabled
onchange="ocultarMensajeDeError('ErrorPromedio') ">
            <label for="PromMin">De menos de 7 a 7.5</label>
        <br>
            <input type="checkbox" class="checksPromedio"
id="PromMed" name="PromMed" disabled
onchange="ocultarMensajeDeError('ErrorPromedio') ">
            <label for="PromMed">De más de 7.5 a 8.5</label>
        <br>
            <input type="checkbox" class="checksPromedio"
id="PromMax" name="PromMax" disabled
onchange="ocultarMensajeDeError('ErrorPromedio') ">
            <label for="PromMax">Más de 8.5</label> <br>
            <input type="checkbox" class="checksPromedio"
id="PromT" name="PromT" disabled
onchange="seleccionarTodo('PromT','checksPromedio','ErrorPromedio') ">
            <label for="PromT">Todos los promedios</label>
        <br>
    </fieldset>
</p>
</div>
</div>
<div class="w3-container">
    <h3 class="w3-mipink w3-padding-small">Selecciona el criterio de
salida:</h3>
    <div class="CriteriosDeSalida w3-row-padding w3-pale-red ">
        <label id="ErrorCS"
            class="w3-padding-small w3-bottombar w3-border w3-border-
mired w3-round-large">Selecciona el criterio de
            salida</label>
        <p class="w3-col s4">
            <input type="radio" id="Estatus" name="CriterioSalida"
value="Estatus"
onchange="ocultarMensajeDeError('ErrorCS') ">
            <label for="Estatus"> El alumno finalizó o nó</label><br>
        </p>
        <p class="w3-col s4">
            <input type="radio" id="tiempoF" name="CriterioSalida"
value="tiempoF"
onchange="ocultarMensajeDeError('ErrorCS') ">
            <label for="tiempoF"> Tiempo de finalización del
alumno</label><br>
        </p>
        <p class="w3-col s4">

```

```

        <input type="radio" id="tiempoA" name="CriterioSalida"
value="tiempoA"
        onchange="ocultarMensajeDeError('ErrorCS') ">
        <label for="tiempoA">Tiempo activo del alumno</label><br>
    </p>
</div>
</div>
<div>
    <p class="w3-container w3-center">
        <input class="w3-btn w3-border w3-xlarge w3-grey w3-hover-
green" type="submit" value="Aceptar"
        name="Aceptar" id="Aceptar">
    </p>
</form>
</div>
<div class="w3-panel w3-pale-red w3-round-large" id="Graficas">
    <p></p>
    <div class="w3-container w3-center " id="graficaBarras">
        <canvas id="graficaB" class="w3-center w3-white"> </canvas>
        <p class="w3-center">
            <input type="button" value="Descargar PDF" name="PDFB"
id="PDFB"></input>
            <input type="button" value="Descargar PNG" name="PNGB"
id="PNGB"></input>
            <input type="button" value="Nueva gráfica" name="nuevaG"
id="nuevaG" onclick="recargarPagina()" "></input>
        </p>
    </div>
    <div class="w3-container w3-center " id="graficaBarrasP">
        <canvas id="graficaBP" class="w3-center w3-white"></canvas>
        <p class="w3-center">
            <input type="button" value="Descargar PDF" name="PDFBP"
id="PDFBP"></input>
            <input type="button" value="Descargar PNG" name="PNGBP"
id="PNGBP"></input>
            <input type="button" value="Nueva gráfica" name="nuevaG"
id="nuevaG" onclick="recargarPagina()" "></input>
        </p>
    </div>

    <div class="w3-container" id="graficaArea">
        <canvas id="graficaA" class="w3-center w3-white"></canvas>
        <p class="w3-center">
            <input type="button" value="Descargar PDF" name="PDFA"
id="PDFA"></input>
            <input type="button" value="Descargar PNG" name="PNGA"
id="PNGA"></input>
            <input type="button" value="Nueva gráfica" name="nuevaG"
id="nuevaG" onclick="recargarPagina()" "></input>
        </p>
    </div>
    <div class="w3-container" id="graficaAreaP">
        <canvas id="graficaAP" class="w3-center w3-white"></canvas>
        <p class="w3-center">
            <input type="button" value="Descargar PDF" name="PDFAP"
id="PDFAP"></input>
            <input type="button" value="Descargar PNG" name="PNGAP"
id="PNGAP"></input>

```

```

        <input type="button" value="Nueva gráfica" name="nuevaG"
id="nuevaG" onclick="recargarPagina()" "></input>
    </p>
</div>
<div class="w3-container" id="graficaRadar">
    <canvas id="graficaR" class="w3-center w3-white"></canvas>
    <p class="w3-center">
        <input type="button" value="Descargar PDF" name="PDFR"
id="PDFR"></input>
        <input type="button" value="Descargar PNG" name="PNGR"
id="PNGR"></input>
        <input type="button" value="Nueva gráfica" name="nuevaG"
id="nuevaG" onclick="recargarPagina()" "></input>
    </p>
</div>
</div>
<script type="text/javascript" src="js/validarDatos.js"></script>
<script type="text/javascript" src="js/crearGrafica.js"></script>
<script type="text/javascript" src="js/exportarGrafica.js"></script>
</body>

</html>

```

validarDatos.js

```
var form;
var data;
var datosF;
var seleccionEdades = [];
var sexo = "";
var seleccionPromedios = [];
var seleccionPuntajes = [];
var seleccionEscuelas = [];
var cSalida;
var indice = 0;
ocultarGraficas();
ocultarMensajeDeError('ErrorEdad');
ocultarMensajeDeError('ErrorEscuela');
ocultarMensajeDeError('ErrorGenero');
ocultarMensajeDeError('ErrorPromedio');
ocultarMensajeDeError('ErrorPuntaje');
ocultarMensajeDeError('ErrorCS');
ocultarMensajeDeError('ErrorCE');
/** Definimos un evento y lo asociamos al botón Aceptar */
document.getElementById('Aceptar').addEventListener('click', function (e)
{
    e.preventDefault();
    /** Mandamos a llamar a la función validarFormulario, si esta regresa true continua con lo que sigue en el código */
    if (!validarFormulario())
        return;
    /**Ocultamos los errores que se mostraran en caso de que alguna opción este mal seleccionada */
    ocultarMensajeDeError('ErrorEdad');
    ocultarMensajeDeError('ErrorEscuela');
    ocultarMensajeDeError('ErrorGenero');
    ocultarMensajeDeError('ErrorPromedio');
    ocultarMensajeDeError('ErrorPuntaje');
    ocultarMensajeDeError('ErrorCS');
    ocultarMensajeDeError('ErrorCE');
    /**Recuperamos los datos del formulario y los enviamos a consultas.php, las consultas que nos regresa * las guardamos en datosF */
    form = document.getElementById('formulario');
    data = new FormData(form);
    fetch('consultas.php', {
        method: 'POST',
        body: data
    }).then(datos => datos.json()).then(datosformulario => {
        datosF = datosformulario;
        deshabilitarFormulario();
        procesarDatos(datosF);
        location.href = '#Graficas';
    })
});

function validarFormulario() {
    let chido = false;
    let criteriosEntrada = 0;
    let criterioSalida = false;
```



```

    /**Verificamos si el criterio Edad esta seleccionado, si es así
    revisamos que al
    * menos una de sus respectivas opciones también este seleccionada */
    if (criteriosCh("Edad", "checksEdad")) {
        chido = false
        if (alMenosUno("checksEdad") > 0) {
            chido = true;
            recuperarDatos(seleccionEdades, 'checksEdad');
        } else {
            document.getElementById("Edad").focus();
            document.getElementById("ErrorEdad").style.display = 'block';
        }
        criteriosEntrada++;
    }
    /**Verificamos si el criterio Sexo esta seleccionado, si es así
    revisamos
    * una de sus respectivas opciones también este seleccionada */
    if (document.getElementById('Sexo').checked) {
        chido = false
        if (document.getElementById('Genero').selectedIndex != 0) {
            chido = true;
            sexo = document.getElementById('Genero').value;
        }
        else {
            document.getElementById("Sexo").focus();
            document.getElementById("ErrorGenero").style.display =
'block';
        }
        criteriosEntrada++;
    }
    /**Verificamos si el criterio Escuela esta seleccionado, si es así
    revisamos que al
    * menos una de sus respectivas opciones también este seleccionada */
    if (criteriosCh("Escuela", "checksEscuela")) {
        chido = false
        if (alMenosUno("checksEscuela") > 0) {
            chido = true;
            recuperarDatos(seleccionEscuelas, 'checksEscuela');
        } else {
            document.getElementById("Escuela").focus();
            document.getElementById("ErrorEscuela").style.display =
'block';
        }
        criteriosEntrada++;
    }
    /**Verificamos si el criterio Puntaje esta seleccionado, si es así
    revisamos que al
    * menos una de sus respectivas opciones también este seleccionada */
    if (criteriosCh("Puntaje", "checksPuntaje")) {
        chido = false
        if (alMenosUno("checksPuntaje") > 0) {
            chido = true;
            recuperarDatos(seleccionPuntajes, 'checksPuntaje');
        } else {
            document.getElementById("Puntaje").focus();
            document.getElementById("ErrorPuntaje").style.display =
'block';
        }
    }

```

```

        criteriosEntrada++;
    }
    /**Verificamos si el criterio Promedio esta seleccionado, si es así
    revisamos que al
    * menos una de sus respectivas opciones también este seleccionada */
    if (criteriosCh("Promedio", "checksPromedio")) {
        chido = false
        if (alMenosUno("checksPromedio") > 0) {
            chido = true;
            recuperarDatos(seleccionPromedios, 'checksPromedio');
        } else {
            document.getElementById("Promedio").focus();
            document.getElementById("ErrorPromedio").style.display =
'block';
        }
        criteriosEntrada++;
    }
    /**Si se seleccionó al menos un criterio de entrada validamos que se
    haya
    * seleccionado un criterio de Salida si no es así nos mostrara el
    error correspondiente
    *
    */
    if (criteriosEntrada >= 1) {
        if (document.getElementById('tiempoF').checked ||
            document.getElementById('tiempoA').checked ||
            document.getElementById('Estatus').checked) {
            criterioSalida = true;
            let radio = document.getElementsByName('CriterioSalida');
            for (i = 0; i < radio.length; i++) {
                if (radio[i].checked)
                    cSalida = radio[i].value;
            }
        }
        else {
            document.getElementById("Estatus").focus();
            document.getElementById("ErrorCS").style.display = 'block';
            criterioSalida = false;
        }
    } else {
        location.href = '#CriteriosEntrada';
        document.getElementById("ErrorCE").style.display = 'block';
        chido = false;
    }
    /**Validamos que se haya seleccionado al menos un criterio de entrada
    con al menos
    * una de sus respectivas opciones y el criterio de salida para así
    regresar true,
    * lo cual indica que el formulario es correcto
    */
    if (criteriosEntrada > 0 && chido == true && criterioSalida == true)
        chido = true;
    else
        chido = false;
    return chido;
}
/**Nos ayuda a verificar si al menos una de las diferentes opciones del
criterio en cuestión este seleccionada */

```

```

function alMenosUno(grupo) {
    let checkeds = 0;
    let cajitas = document.getElementsByClassName(grupo);
    for (i = 0; i < cajitas.length; i++) {
        if (cajitas[i].checked === true)
            checkeds++;
    }
    return checkeds;
}
/**Si un criterio es seleccionado se habilitan las casillas que
corresponden a sus diferentes opciones
* si la deseleccionamos sus respectivas opciones se vuelven a
deshabilitar
*/
function criteriosCh(criterio, opcion) {
    let chCriterio = document.getElementById(criterio);
    let opcionesCriterio = document.getElementsByClassName(opcion);
    let error = "Error" + criterio;
    if (chCriterio.checked) {
        document.getElementById("ErrorCE").style.display = 'none';
        for (i = 0; i < opcionesCriterio.length; i++)
            opcionesCriterio[i].disabled = false;
        return true;
    } else {
        for (i = 0; i < opcionesCriterio.length; i++) {
            opcionesCriterio[i].disabled = true;
            opcionesCriterio[i].checked = false;
        }
        document.getElementById(error).style.display = 'none';
        return false;
    }
}
/**Si se selecciona Genero como un criterio de entrada se habilita el
elemento
* <selection> para que pueda elegir una opción
*/
function sexoCh() {
    let chSexo = document.getElementById("Sexo");
    if (chSexo.checked) {
        document.getElementById("ErrorCE").style.display = 'none';
        document.getElementById("Genero").disabled = false;
    }
    else {
        document.getElementById("Genero").disabled = true;
        document.getElementById("Genero").value = "";
    }
}
/**Si seleccionamos la opción de Todos en algun criterio, esta función se
* encarga de marcar el resto de las casillas
*/
function seleccionarTodo(clase, nombreClase, error) {
    let todo = document.getElementById(clase);
    let opciones = document.getElementsByClassName(nombreClase);
    if (todo.checked) {
        document.getElementById(error).style.display = 'none';
        for (i = 0; i < opciones.length - 1; i++) {
            opciones[i].checked = true;
            opciones[i].disabled = true;
        }
    }
}

```

```
    }  
  }  
  else {  
    for (i = 0; i < opciones.length; i++) {  
      opciones[i].checked = false;  
      opciones[i].disabled = false;  
    }  
  }  
}
```

estadoInicial.js

```
/**Esta función se encarga de ocultar las <div> donde se mostraran las  
* diferentes gráficas con sus respectivos botones  
*/  
function ocultarGraficas()  
{  
    let divGraficaBarras = document.getElementById("graficaBarras");  
    divGraficaBarras.style.display = 'none';  
    let divGraficaBarrasP = document.getElementById("graficaBarrasP");  
    divGraficaBarrasP.style.display = 'none';  
    let divGraficaArea = document.getElementById('graficaArea');  
    divGraficaArea.style.display = 'none';  
    let divGraficaAreaP = document.getElementById('graficaAreaP');  
    divGraficaAreaP.style.display = 'none';  
    let divGraficaRadar = document.getElementById('graficaRadar');  
    divGraficaRadar.style.display = 'none';  
}  
  
/**Esta función se encarga de ocultar los errores que se mostraran cuando  
se haga  
* una selección incorrecta  
*/  
function ocultarMensajeDeError(error){  
    document.getElementById(error).style.display = 'none';  
}  
  
/**Esta función se encarga de recargar la página para que el usuario  
* pueda seleccionar nuevos criterios y crear nuevas gráficas  
*/  
function recargarPagina(){  
    var url = window.location.hostname;  
    var r = confirm("Se perderán las gráficas mostradas");  
    if (r == true) {  
        location.replace('http://' + url + '/PT/');  
    }  
}
```

Apéndice B. Código fuente módulo de obtención y procesamiento de datos

conexión.php

```
<?php
    $db_host = "localhost";
    $db_user = "user";
    $db_pwd = "password";
    $db_name = "datos";
    $conexion = mysqli_connect($db_host,$db_user,$db_pwd);
    if(!$conexion)
    {
        die("No se ha podido realizar la conexion al
servidor".mysqli_error());
    }
    else
    {
        mysqli_select_db($conexion,$db_name) or die("Error al conectar a
la base de datos");
        //echo "Conexion exitosa";
    }
?>
```

consultas.php

```
<?php
require("conexion.php"); //Agregamos el archivo conexion.php
/**En este arreglo se guardan los criterios de entrada y
 * salida que el usuario haya seleccionado
 */
$criteriosSeleccionados=array();
/**contamos cuantos criterios se seleccionaron su valor
 * maximo es 5 ya que solo se cuentan los de entrada
 */
$contadorCriterios = 0;
$conjuntoEdad;
$conjuntoSexo;
$conjuntoEscuela;
$conjuntoPromedio;
$conjuntoPuntaje;
seleccionarCriteriosEntrada();
crearConsultas();

/**Esta función se encarga de comunicarse con la base de datos y
 * realizar las consultas necesarias de acuerdo a los criterios
 * seleccionados
 */
function realizarConsultas($consultas)
{
    global $conexion;
    $datos = array(3);
    $datosGrafica = array(count($consultas));
    $bandera;
    $titulo;
    $contador=0;
    /**El inicio de las consultas que se realizaran a la base de datos
     * siempre sera el mismo, sin embargo, la parte final depende del
     * criterio de salida que se haya seleccionado, mientras que la
     * parte central sera de acuerdo al arreglo final de las consultas
     * construidas
     */
    $inicio="SELECT NTRI, count(*) as CONTEO FROM datosaga where";
    if($_POST['CriterioSalida'] == 'tiempoF')
    {
        $contador=0;
        $fin = "AND EDO2 = 22 AND NTRI > 8 group by NTRI";
        for($i=0;$i<count($consultas);$i++){
            $consultaFinal = $inicio." ".$consultas[$i]." ".$fin;
            $consulta = mysqli_query($conexion, ($consultaFinal.";"));
            $data = array();
            foreach($consulta as $row)
                $data[] = $row;
            $datosGrafica[$contador] = $data;
            $contador++;
        }
        /**Esta bandera indicara que las gráficas que deben generarse son
         * de barras y de área
         */
        $bandera=0;
        $titulo="Tiempo de finalización";
    }
}
```

```

elseif($_POST['CriterioSalida'] == 'tiempoA')
{
    $contador=0;
    $fin = " AND EDO IN (1,2,10) group by NTRI";
    for($i=0;$i<count($consultas);$i++){
        $consultaFinal = $inicio." ".$consultas[$i]." ".$fin;
        $consulta = mysqli_query($conexion, ($consultaFinal.";"));
        $data = array();
        foreach($consulta as $row)
            $data[] = $row;
        $datosGrafica[$contador] = $data;
        $contador++;
    }
    /**Esta bandera indicara que las gráficas que deben generarse son
    * de barras y de área
    */
    $bandera=0;
    $titulo = "Tiempo activo";
}
elseif($_POST['CriterioSalida'] == 'Estatus')
{
    $contador=0;
    $fin = " AND (EDO in (4,7) and NTRI>0) group by NTRI";
    for($i=0;$i<count($consultas);$i++){
        $consultaFinal = $inicio." ".$consultas[$i]." ".$fin;
        $consulta = mysqli_query($conexion, ($consultaFinal.";"));
        $data = array();
        foreach($consulta as $row)
            $data[] = $row;
        $datosGrafica[$contador] = $data;
        $contador++;
    }
    $fin = " AND (EDO2=22 and NTRI>8) group by NTRI";
    for($i=0;$i<count($consultas);$i++){
        $consultaFinal = $inicio." ".$consultas[$i]." ".$fin;
        $consulta = mysqli_query($conexion, ($consultaFinal.";"));
        $data = array();
        foreach($consulta as $row)
            $data[] = $row;
        $datosGrafica[$contador] = $data;
        $contador++;
    }
    /**Esta bandera indicara que debe generarse una gráfica radar
    */
    $bandera=1;
    $titulo = "El alumno finalizó o no";
}
$datos[0]=$datosGrafica;
$datos[1]=$bandera;
$datos[2]=$titulo;
return $datos;
}

/**Indica el conjunto según corresponda */
function seleccionarConjunto($criterio){
    global $conjuntoEdad;
    global $conjuntoSexo;
    global $conjuntoEscuela;

```



```

global $conjuntoPromedio;
global $conjuntoPuntaje;
$conjuntoSeleccionado;
switch ($criterio) {
    case 'Edad':
        $conjuntoSeleccionado = $conjuntoEdad;
        break;
    case 'Promedio':
        $conjuntoSeleccionado = $conjuntoPromedio;
        break;
    case 'Puntaje':
        $conjuntoSeleccionado = $conjuntoPuntaje;
        break;
    case 'Sexo':
        $conjuntoSeleccionado = $conjuntoSexo;
        break;
    case 'Escuela':
        $conjuntoSeleccionado = $conjuntoEscuela;
        break;
    default:
        break;
}
return $conjuntoSeleccionado;
}

/**Esta función se encarga de crear los arreglos con diferentes consultas
 * y letreros segun la cantidad de criterios seleccionados y realizar las
 * consultas (llamando a la función realizarConsultas,
 */

function crearConsultas()
{
    global $criteriosSeleccionados;
    global $contadorCriterios;
    $datos = array();
    $bandera;
    $titulo;
    $consultas = array();
    $labelsdata = array();
    $datos = array();
    if($contadorCriterios == 1)
    {
        $conjunto = seleccionarConjunto($criteriosSeleccionados[0]);
        $consultas = $conjunto[1];
        $datos = realizarConsultas($consultas);
        $datos= $datos[0];
        $bandera = $datos[1];
        $titulo = $datos[2];
        if($bandera == 0)
        {
            $labelsdata = $conjunto[0];
        }
        elseif($bandera == 1)
        {
            $contadorL=0;
            for($i=0;$i<count($conjunto[0]);$i++)
            {
                $labelsdata[$contadorL]=$conjunto[0][$i]." - *NF";
                $contadorL++;
            }
        }
    }
}

```

```

    }
    for($i=0;$i<count($conjunto[0]);$i++)
    {
        $labelsdata[$contadorL]=$conjunto[0][$i]." - *F";
        $contadorL++;
    }
}
elseif($contadorCriterios == 2)
{
    $conjunto1 = seleccionarConjunto($criteriosSeleccionados[0]);
    $conjunto2 = seleccionarConjunto($criteriosSeleccionados[1]);
    $consultasAux1 = array();
    $consultasAux2 = array();
    $letrerosAux1 = array();
    $letrerosAux2 = array();
    $consultasAux1 = $conjunto1[1];
    $consultasAux2 = $conjunto2[1];
    $letrerosAux1 = $conjunto1[0];
    $letrerosAux2 = $conjunto2[0];
    $contador=0;
    for($i=0;$i<count($consultasAux1);$i++)
        for($j=0;$j<count($consultasAux2);$j++){
            $consultas[$contador]=$consultasAux1[$i].' AND
'. $consultasAux2[$j];
            $contador++;
        }
    $datos = realizarConsultas($consultas);
    $datos= $datos[0];
    $bandera = $datos[1];
    $titulo = $datos[2];
    $contador=0;
    if($bandera == 0)
    {
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++){
                $labelsdata[$contador]=$letrerosAux1[$i].' -
'. $letrerosAux2[$j];
                $contador++;
            }
    }
    elseif($bandera == 1)
    {
        $contadorL=0;
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++){
                $labelsdata[$contadorL]=$letrerosAux1[$i].' -
'. $letrerosAux2[$j]." *NF";
                $contadorL++;
            }
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++){
                $labelsdata[$contadorL]=$letrerosAux1[$i].' -
'. $letrerosAux2[$j]." *F";
                $contadorL++;
            }
    }
}

```

```

}
elseif($contadorCriterios == 3)
{
    $conjunto1 = seleccionarConjunto($criteriosSeleccionados[0]);
    $conjunto2 = seleccionarConjunto($criteriosSeleccionados[1]);
    $conjunto3 = seleccionarConjunto($criteriosSeleccionados[2]);
    $consultasAux1 = array();
    $consultasAux2 = array();
    $consultasAux3 = array();
    $letrerosAux1 = array();
    $letrerosAux2 = array();
    $letrerosAux3 = array();
    $consultasAux1 = $conjunto1[1];
    $consultasAux2 = $conjunto2[1];
    $consultasAux3 = $conjunto3[1];
    $letrerosAux1 = $conjunto1[0];
    $letrerosAux2 = $conjunto2[0];
    $letrerosAux3 = $conjunto3[0];
    $contador=0;
    for($i=0;$i<count($consultasAux1);$i++)
        for($j=0;$j<count($consultasAux2);$j++)
            for($k=0;$k<count($consultasAux3);$k++){
                $consultas[$contador]=$consultasAux1[$i].' AND
'.$consultasAux2[$j].' AND '.$consultasAux3[$k];
                $contador++;
            }
    $datos = realizarConsultas($consultas);
    $datos= $datos[0];
    $bandera = $datos[1];
    $titulo = $datos[2];
    $contador=0;
    if($bandera == 0)
    {
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++)
                for($k=0;$k<count($letrerosAux3);$k++){
                    $labelsdata[$contador]=$letrerosAux1[$i].' -
'.$letrerosAux2[$j].' - '.$letrerosAux3[$k];
                    $contador++;
                }
    }
    elseif ($bandera == 1 )
    {
        $contadorL=0;
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++)
                for($k=0;$k<count($letrerosAux3);$k++){
                    $labelsdata[$contador]=$letrerosAux1[$i].' -
'.$letrerosAux2[$j].' - '.$letrerosAux3[$k]." *NF";
                    $contador++;
                }
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++)
                for($k=0;$k<count($letrerosAux3);$k++){
                    $labelsdata[$contador]=$letrerosAux1[$i].' -
'.$letrerosAux2[$j].' - '.$letrerosAux3[$k]." *F";
                    $contador++;
                }
    }
}

```

```

    }

}

elseif($contadorCriterios == 4)
{
    $conjunto1 = seleccionarConjunto($criteriosSeleccionados[0]);
    $conjunto2 = seleccionarConjunto($criteriosSeleccionados[1]);
    $conjunto3 = seleccionarConjunto($criteriosSeleccionados[2]);
    $conjunto4 = seleccionarConjunto($criteriosSeleccionados[3]);
    $consultasAux1 = array();
    $consultasAux2 = array();
    $consultasAux3 = array();
    $consultasAux4 = array();
    $letrerosAux1 = array();
    $letrerosAux2 = array();
    $letrerosAux3 = array();
    $letrerosAux4 = array();
    $consultasAux1 = $conjunto1[1];
    $consultasAux2 = $conjunto2[1];
    $consultasAux3 = $conjunto3[1];
    $consultasAux4 = $conjunto4[1];
    $letrerosAux1 = $conjunto1[0];
    $letrerosAux2 = $conjunto2[0];
    $letrerosAux3 = $conjunto3[0];
    $letrerosAux4 = $conjunto4[0];
    $contador=0;
    for($i=0;$i<count($consultasAux1);$i++)
        for($j=0;$j<count($consultasAux2);$j++)
            for($k=0;$k<count($consultasAux3);$k++)
                for($l=0;$l<count($consultasAux4);$l++)
                {
                    $consultas[$contador]=$consultasAux1[$i].
                        ' AND ' . $consultasAux2[$j].
                        ' AND ' . $consultasAux3[$k].
                        ' AND ' . $consultasAux4[$l];
                    $contador++;
                }
    $datos = realizarConsultas($consultas);
    $datos= $datos[0];
    $bandera = $datos[1];
    $titulo = $datos[2];
    $contador=0;
    if($bandera == 0)
    {
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++)
                for($k=0;$k<count($letrerosAux3);$k++)
                    for($l=0;$l<count($letrerosAux4);$l++){
                        $labelsdata[$contador]=$letrerosAux1[$i]. ' -
'. $letrerosAux2[$j]. ' - ' . $letrerosAux3[$k]. ' - ' . $letrerosAux4[$l];
                        $contador++;
                    }
    }
    elseif($bandera == 1)
    {
        $contadorL=0;
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++)

```

```

        for($k=0;$k<count($letrerosAux3);$k++)
            for($l=0;$l<count($letrerosAux4);$l++){
                $labelsdata[$contador]=$letrerosAux1[$i].' -
'. $letrerosAux2[$j].' - '. $letrerosAux3[$k].' - '. $letrerosAux4[$l].'
*NF';
                $contador++;
            }
        for($i=0;$i<count($letrerosAux1);$i++)
            for($j=0;$j<count($letrerosAux2);$j++)
                for($k=0;$k<count($letrerosAux3);$k++)
                    for($l=0;$l<count($letrerosAux4);$l++){
                        $labelsdata[$contador]=$letrerosAux1[$i].' -
'. $letrerosAux2[$j].' - '. $letrerosAux3[$k].' - '. $letrerosAux4[$l].'
*F';
                        $contador++;
                    }
    }
}
elseif($contadorCriterios == 5)
{
    $conjunto1 = seleccionarConjunto($criteriosSeleccionados[0]);
    $conjunto2 = seleccionarConjunto($criteriosSeleccionados[1]);
    $conjunto3 = seleccionarConjunto($criteriosSeleccionados[2]);
    $conjunto4 = seleccionarConjunto($criteriosSeleccionados[3]);
    $conjunto5 = seleccionarConjunto($criteriosSeleccionados[4]);
    $consultasAux1 = array();
    $consultasAux2 = array();
    $consultasAux3 = array();
    $consultasAux4 = array();
    $consultasAux5 = array();
    $letrerosAux1 = array();
    $letrerosAux2 = array();
    $letrerosAux3 = array();
    $letrerosAux4 = array();
    $letrerosAux5 = array();
    $consultasAux1 = $conjunto1[1];
    $consultasAux2 = $conjunto2[1];
    $consultasAux3 = $conjunto3[1];
    $consultasAux4 = $conjunto4[1];
    $consultasAux5 = $conjunto5[1];
    $letrerosAux1 = $conjunto1[0];
    $letrerosAux2 = $conjunto2[0];
    $letrerosAux3 = $conjunto3[0];
    $letrerosAux4 = $conjunto4[0];
    $letrerosAux5 = $conjunto5[0];
    $contador=0;
    for($i=0;$i<count($consultasAux1);$i++)
        for($j=0;$j<count($consultasAux2);$j++)
            for($k=0;$k<count($consultasAux3);$k++)
                for($l=0;$l<count($consultasAux4);$l++)
                    for($m=0;$m<count($consultasAux5);$m++)
                    {
                        $consultas[$contador]=$consultasAux1[$i].
                        ' AND '. $consultasAux2[$j].
                        ' AND '. $consultasAux3[$k].
                        ' AND '. $consultasAux4[$l].
                        ' AND '. $consultasAux5[$m];
                        $contador++;
                    }
}

```

```

    }
    $datos = realizarConsultas($consultas);
    $datos= $datos[0];
    $bandera = $datos[1];
    $titulo = $datos[2];
    $contador = 0;
    if($bandera == 0)
    {
        for($i=0;$i<count($letrerosAux1);$i++)
        for($j=0;$j<count($letrerosAux2);$j++)
        for($k=0;$k<count($letrerosAux3);$k++)
        for($l=0;$l<count($letrerosAux4);$l++)
        for($m=0;$m<count($letrerosAux5);$m++){
            $labelsdata[$contador]=$letrerosAux1[$i].
                ' - '.$letrerosAux2[$j].
                ' - '.$letrerosAux3[$k].
                ' - '.$letrerosAux4[$l].
                ' - '.$letrerosAux5[$m].' *NF';
            $contador++;
        }
    }
    elseif($bandera == 1)
    {
        for($i=0;$i<count($letrerosAux1);$i++)
        for($j=0;$j<count($letrerosAux2);$j++)
        for($k=0;$k<count($letrerosAux3);$k++)
        for($l=0;$l<count($letrerosAux4);$l++)
        for($m=0;$m<count($letrerosAux5);$m++){
            $labelsdata[$contador]=$letrerosAux1[$i].
                ' - '.$letrerosAux2[$j].
                ' - '.$letrerosAux3[$k].
                ' - '.$letrerosAux4[$l].
                ' - '.$letrerosAux5[$m].' *F';
            $contador++;
        }
    }
}

$arregloFinal = array(4);
$arregloFinal[0] = $labelsdata;
$arregloFinal[1]= $datos;
$arregloFinal[2]=$bandera;
$arregloFinal[3]=$titulo;
print json_encode($arregloFinal);
}
/**Esta función se encarga de validar que criterios fueron seleccionados
*/
function seleccionarCriteriosEntrada()
{
    global $criteriosSeleccionados;
    global $contadorCriterios;
    global $conjuntoEdad;
    global $conjuntoSexo;
    global $conjuntoEscuela;
    global $conjuntoPromedio;
    global $conjuntoPuntaje;
    if(isset($_POST["Edad"]))
    {

```

```

        $criteriosSeleccionados[$contadorCriterios] = 'Edad';
        $contadorCriterios++;
        $conjuntoEdad = procesarEdad();
    }
    if(isset($_POST["Promedio"]))
    {
        $criteriosSeleccionados[$contadorCriterios] = 'Promedio';
        $contadorCriterios++;
        $conjuntoPromedio = procesarPromedio();
    }
    if(isset($_POST["Puntaje"]))
    {
        $criteriosSeleccionados[$contadorCriterios] = 'Puntaje';
        $contadorCriterios++;
        $conjuntoPuntaje = procesarPuntaje();
    }
    if(isset($_POST["Sexo"]))
    {
        $criteriosSeleccionados[$contadorCriterios] = 'Sexo';
        $contadorCriterios++;
        $conjuntoSexo = procesarSexo();
    }
    if(isset($_POST["Escuela"]))
    {
        $criteriosSeleccionados[$contadorCriterios] = 'Escuela';
        $contadorCriterios++;
        $conjuntoEscuela = procesarEscuela();
    }
}
/**Esta función se encarga de validar cuantos rangos correspondientes a
 * Edad fueron seleccionados, de acuerdo al numero de rangos
 seleccionados
 * crea las diferentes "consultas" y "letreros" para este campo
 */
function procesarEdad()
{
    $rangosEdadSeleccionados=array();
    $contadorRangos=0;
    $contadorConsultas=0;
    $consultasEdad=array();
    $letrerosEdad=array();
    if(isset($_POST['REdad1']))
    {
        $rangosEdadSeleccionados[$contadorRangos]='REdad1';
        $contadorRangos++;
    }
    if(isset($_POST['REdad2']))
    {
        $rangosEdadSeleccionados[$contadorRangos]='REdad2';
        $contadorRangos++;
    }
    if(isset($_POST['REdad3']))
    {
        $rangosEdadSeleccionados[$contadorRangos]='REdad3';
        $contadorRangos++;
    }
    if(isset($_POST['REdad4']))

```

```

{
    $rangosEdadSeleccionados[$contadorRangos]='REdad4';
    $contadorRangos++;
}
if(isset ($_POST['REdad5']))
{
    $rangosEdadSeleccionados[$contadorRangos]='REdad5';
    $contadorRangos++;
}
if(isset ($_POST['REdad6']))
{
    $rangosEdadSeleccionados[$contadorRangos]='REdad6';
    $contadorRangos++;
}
if($contadorRangos<3)
{
    for($i=0;$i<$contadorRangos;$i++)
    {
        switch ($rangosEdadSeleccionados[$i]) {
            case 'REdad1':
                $consultasEdad[$contadorConsultas]=' EDAD = 16 ';
                $letrerosEdad[$contadorConsultas] = '16';
                $contadorConsultas++;
                $consultasEdad[$contadorConsultas]=' EDAD = 17 ';
                $letrerosEdad[$contadorConsultas] = '17';
                $contadorConsultas++;
                break;
            case 'REdad2':
                $consultasEdad[$contadorConsultas]=' EDAD = 18 ';
                $letrerosEdad[$contadorConsultas] = '18';
                $contadorConsultas++;
                break;
            case 'REdad3':
                $consultasEdad[$contadorConsultas]=' EDAD = 19 ';
                $letrerosEdad[$contadorConsultas] = '19';
                $contadorConsultas++;
                $consultasEdad[$contadorConsultas]=' EDAD = 20 ';
                $letrerosEdad[$contadorConsultas] = '20';
                $contadorConsultas++;
                break;
            case 'REdad4':
                $consultasEdad[$contadorConsultas]=' EDAD = 21 ';
                $letrerosEdad[$contadorConsultas] = '21';
                $contadorConsultas++;
                $consultasEdad[$contadorConsultas]=' EDAD = 22 ';
                $letrerosEdad[$contadorConsultas] = '22';
                $contadorConsultas++;
                $consultasEdad[$contadorConsultas]=' EDAD = 23 ';
                $letrerosEdad[$contadorConsultas] = '23';
                $contadorConsultas++;
                break;
            case 'REdad5':
                $consultasEdad[$contadorConsultas]=' EDAD = 24 ';
                $letrerosEdad[$contadorConsultas] = '24';
                $contadorConsultas++;
                $consultasEdad[$contadorConsultas]=' EDAD = 25 ';
                $letrerosEdad[$contadorConsultas] = '25';
                $contadorConsultas++;
        }
    }
}

```



```

        $consultasEdad[$contadorConsultas]=' EDAD = 26 ';
        $letrerosEdad[$contadorConsultas] = '26';
        $contadorConsultas++;
        break;
    case 'REdad6':
        $consultasEdad[$contadorConsultas]=' EDAD between 27
and 37 ';
        $letrerosEdad[$contadorConsultas] = '27 a 37';
        $contadorConsultas++;
        $consultasEdad[$contadorConsultas]=' EDAD between 38
and 47 ';
        $letrerosEdad[$contadorConsultas] = '38 a 47';
        $contadorConsultas++;
        $consultasEdad[$contadorConsultas]=' EDAD between 48
and 57 ';
        $letrerosEdad[$contadorConsultas] = '48 a 57';
        $contadorConsultas++;
        $consultasEdad[$contadorConsultas]=' EDAD > 57 ';
        $letrerosEdad[$contadorConsultas] = '> 57';
        $contadorConsultas++;
        break;
    default:
        break;
    }
}
}
else{
    for($i=0;$i<$contadorRangos;$i++)
    {
        switch ($rangosEdadSeleccionados[$i]) {
            case 'REdad1':
                $consultasEdad[$contadorConsultas]='EDAD BETWEEN 16
AND 17 ';
                $letrerosEdad[$contadorConsultas] = '<18';
                $contadorConsultas++;
                break;
            case 'REdad2':
                $consultasEdad[$contadorConsultas]='EDAD = 18 ';
                $letrerosEdad[$contadorConsultas] = '18';
                $contadorConsultas++;
                break;
            case 'REdad3':
                $consultasEdad[$contadorConsultas]='EDAD BETWEEN 19
AND 20 ';
                $letrerosEdad[$contadorConsultas] = '19 - 20';
                $contadorConsultas++;
                break;
            case 'REdad4':
                $consultasEdad[$contadorConsultas]='EDAD BETWEEN 21
AND 23 ';
                $letrerosEdad[$contadorConsultas] = '21 - 23';
                $contadorConsultas++;
                break;
            case 'REdad5':
                $consultasEdad[$contadorConsultas]='EDAD BETWEEN 24
AND 26 ';
                $letrerosEdad[$contadorConsultas] = '24 - 26';

```

```

        $contadorConsultas++;
        break;
    case 'REdad6':
        $consultasEdad[$contadorConsultas]='EDAD > 26 ';
        $letrerosEdad[$contadorConsultas] = '> 26';
        $contadorConsultas++;
        break;
    default:
        break;
    }
}
}
$procesoEdad = array(2);
$procesoEdad[0] = $letrerosEdad;
$procesoEdad[1]= $consultasEdad;
return $procesoEdad;
}

/**Esta función se encarga de validar cuantos rangos correspondientes a
 * Promedio fueron seleccionados, de acuerdo al numero de rangos
 seleccionados
 * crea las diferentes "consultas" y "letreros" para este campo
 */
function procesarPromedio()
{
    $rangosPromSeleccionados=array();
    $contadorRangos=0;
    $contadorConsultas=0;
    $consultasPromedio=array();
    $letrerosPromedio=array();
    if(isset ($_POST['PromMin']))
    {
        $rangosPromSeleccionados[$contadorRangos]='PromMin';
        $contadorRangos++;
    }
    if(isset ($_POST['PromMed']))
    {
        $rangosPromSeleccionados[$contadorRangos]='PromMed';
        $contadorRangos++;
    }
    if(isset ($_POST['PromMax']))
    {
        $rangosPromSeleccionados[$contadorRangos]='PromMax';
        $contadorRangos++;
    }
    if($contadorRangos<3)
    {
        for($i=0;$i<$contadorRangos;$i++)
        {
            switch ($rangosPromSeleccionados[$i]) {
                case 'PromMin':
                    $consultasPromedio[$contadorConsultas]=' FROM BETWEEN
6 AND 7 ';
                    $letrerosPromedio[$contadorConsultas] = '<7';
                    $contadorConsultas++;
                    $consultasPromedio[$contadorConsultas]=' FROM BETWEEN
7.01 AND 7.5 ';

```

```

$letrerosPromedio[$contadorConsultas] = '[7.01 -
7.5]';
$contadorConsultas++;
break;
case 'PromMed':
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
7.51 AND 8 ';
$letrerosPromedio[$contadorConsultas] = '[7.51 - 8]';
$contadorConsultas++;
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
8.01 AND 8.5 ';
$letrerosPromedio[$contadorConsultas] = '[8.01 -
8.5]';
$contadorConsultas++;
break;
case 'PromMax':
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
8.51 AND 9 ';
$letrerosPromedio[$contadorConsultas] = '[8.51 - 9]';
$contadorConsultas++;
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
9.01 AND 9.5 ';
$letrerosPromedio[$contadorConsultas] = '[9.01 -
9.5]';
$contadorConsultas++;
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
9.51 AND 10 ';
$letrerosPromedio[$contadorConsultas] = '[9.51 -
10]';
$contadorConsultas++;
break;
default:
break;
}
}
}
else{
for($i=0;$i<$contadorRangos;$i++)
{
switch ($rangosPromSeleccionados[$i]) {
case 'PromMin':
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
6 AND 7.5 ';
$letrerosPromedio[$contadorConsultas] = '[<7 - 7.5]';
$contadorConsultas++;
break;
case 'PromMed':
$consultasPromedio[$contadorConsultas]=' FROM BETWEEN
7.51 AND 8.5 ';
$letrerosPromedio[$contadorConsultas] = '[7.51 -
8.5]';
$contadorConsultas++;
break;
case 'PromMax':
$consultasPromedio[$contadorConsultas]=' FROM > 8.5
';
$letrerosPromedio[$contadorConsultas] = '[8.5 - 10]';
$contadorConsultas++;

```

```

        break;
    default:
        break;
    }
}

$procesoPromedio = array(2);
$procesoPromedio[0] = $letrerosPromedio;
$procesoPromedio[1]= $consultasPromedio;
return $procesoPromedio;
}

/**Esta función se encarga de validar cuantos rangos correspondientes a
 * Puntaje fueron seleccionados, de acuerdo al numero de rangos
seleccionados
 * crea las diferentes "consultas" y "letreros" para este campo
 */
function procesarPuntaje()
{
    $rangosPuntajeSeleccionados=array();
    $contadorRangos=0;
    $contadorConsultas=0;
    $consultasPuntaje=array();
    $letrerosPuntaje=array();
    if(isset ($_POST['PuntajeMin']))
    {
        $rangosPuntajeSeleccionados[$contadorRangos]='PuntajeMin';
        $contadorRangos++;
    }
    if(isset ($_POST['PuntajeMed']))
    {
        $rangosPuntajeSeleccionados[$contadorRangos]='PuntajeMed';
        $contadorRangos++;
    }
    if(isset ($_POST['PuntajeMax']))
    {
        $rangosPuntajeSeleccionados[$contadorRangos]='PuntajeMax';
        $contadorRangos++;
    }
    if($contadorRangos<3)
    {
        for($i=0;$i<$contadorRangos;$i++)
        {
            switch ($rangosPuntajeSeleccionados[$i]) {
                case 'PuntajeMin':
                    $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 500 AND 600 ';
                    $letrerosPuntaje[$contadorConsultas] = ' [<600]';
                    $contadorConsultas++;
                    $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 601 AND 650 ';
                    $letrerosPuntaje[$contadorConsultas] = '[601 - 650]';
                    $contadorConsultas++;
                    $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 651 AND 700 ';
                    $letrerosPuntaje[$contadorConsultas] = '[651 - 700]';
                    $contadorConsultas++;

```

```

        $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 701 AND 750 ';
        $letrerosPuntaje[$contadorConsultas] = '[701 - 750]';
        $contadorConsultas++;
        break;
        case 'PuntajeMed':
        $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 751 AND 800 ';
        $letrerosPuntaje[$contadorConsultas] = '[751 - 800]';
        $contadorConsultas++;
        $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 801 AND 850 ';
        $letrerosPuntaje[$contadorConsultas] = '[801 - 850]';
        $contadorConsultas++;
        break;
        case 'PuntajeMax':
        $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 851 AND 900 ';
        $letrerosPuntaje[$contadorConsultas] = '[851 - 900]';
        $contadorConsultas++;
        $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 901 AND 950 ';
        $letrerosPuntaje[$contadorConsultas] = '[901 - 950]';
        $contadorConsultas++;
        $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 951 AND 1000 ';
        $letrerosPuntaje[$contadorConsultas] = '[951 -
1000]';

        $contadorConsultas++;
        break;
        default:
        break;
    }
}
}
else{
    for($i=0;$i<$contadorRangos;$i++)
    {
        switch ($rangosPuntajeSeleccionados[$i]) {
            case 'PuntajeMin':
                $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 500 AND 750 ';
                $letrerosPuntaje[$contadorConsultas] = '[<500 -
750]';

                $contadorConsultas++;
                break;
            case 'PuntajeMed':
                $consultasPuntaje[$contadorConsultas]=' PUNTAJE
BETWEEN 751 AND 850 ';
                $letrerosPuntaje[$contadorConsultas] = '[751 - 850]';
                $contadorConsultas++;
                break;
            case 'PuntajeMax':
                $consultasPuntaje[$contadorConsultas]=' PUNTAJE > 850
';

                $letrerosPuntaje[$contadorConsultas] = '[851 -
1000]';

                $contadorConsultas++;

```

```

                break;
            default:
                break;
        }
    }
}
$procesoPuntaje = array(2);
$procesoPuntaje[0] = $letrerosPuntaje;
$procesoPuntaje[1]= $consultasPuntaje;
return $procesoPuntaje;
}
/**Esta función se encarga de validar cuantos rangos correspondientes a
 * Escuela fueron seleccionados, de acuerdo al numero de rangos
 seleccionados
 * crea las diferentes "consultas" y "letreros" para este campo
 */
function procesarEscuela()
{
    $escuelasSeleccionadas=array();
    $contadorEscuelas=0;
    $contadorConsultas=0;
    $consultasEscuela=array();
    $letrerosEscuela=array();
    if(isset($_POST["UNAM"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='UNAM';
        $contadorEscuelas++;
    }
    if(isset($_POST["IPN"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='IPN';
        $contadorEscuelas++;
    }
    if(isset($_POST["DGETI"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='DGETI';
        $contadorEscuelas++;
    }
    if(isset($_POST["CBACH"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='CBACH';
        $contadorEscuelas++;
    }
    if(isset($_POST["CONALEP"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='CONALEP';
        $contadorEscuelas++;
    }
    if(isset($_POST["UAEM"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='UAEM';
        $contadorEscuelas++;
    }
    if(isset($_POST["UNIVEST"]))
    {
        $escuelasSeleccionadas[$contadorEscuelas]='UNIVEST';
        $contadorEscuelas++;
    }
}

```

```

if(isset($_POST["NORMAL"]))
{
    $escuelasSeleccionadas[$contadorEscuelas]='NORMAL';
    $contadorEscuelas++;
}
if(isset($_POST["SEP"]))
{
    $escuelasSeleccionadas[$contadorEscuelas]='SEP';
    $contadorEscuelas++;
}
if(isset($_POST["OTRA"]))
{
    $escuelasSeleccionadas[$contadorEscuelas]='OTRA';
    $contadorEscuelas++;
}
if(isset($_POST["ND"]))
{
    $escuelasSeleccionadas[$contadorEscuelas]='ND';
    $contadorEscuelas++;
}
for($i=0;$i<$contadorEscuelas;$i++)
{
    /**En la base de datos cada escuela esta representada por un
    número por lo
    * cual las consultas se realizan de la manera en la que se
    oobserva en el
    * siguiente switch
    */
    switch ($escuelasSeleccionadas[$i]){
        case 'UNAM':
            $consultasEscuela[$contadorConsultas]=' ESC IN ( 1,2,13 )
';
            $letrerosEscuela[$contadorConsultas] = ' UNAM ';
            $contadorConsultas++;
            break;
        case 'IPN':
            $consultasEscuela[$contadorConsultas]=' ESC IN ( 3,14 )
';
            $letrerosEscuela[$contadorConsultas] = ' IPN ';
            $contadorConsultas++;
            break;
        case 'DGETI':
            $consultasEscuela[$contadorConsultas]=' ESC IN (
5,6,7,8,9,10 ) ';
            $letrerosEscuela[$contadorConsultas] = ' DGETI ';
            $contadorConsultas++;
            break;
        case 'CBACH':
            $consultasEscuela[$contadorConsultas]=' ESC = 4 ';
            $letrerosEscuela[$contadorConsultas] = ' Colegio de
Bachilleres ';
            $contadorConsultas++;
            break;
        case 'CONALEP':
            $consultasEscuela[$contadorConsultas]=' ESC =23 ';
            $letrerosEscuela[$contadorConsultas] = ' CONALEP ';
            $contadorConsultas++;
            break;
    }
}

```

```

        case 'UAEM':
            $consultasEscuela[$contadorConsultas]=' ESC IN ( 18,19 )
';
            $letrerosEscuela[$contadorConsultas] = ' UAEM ';
            $contadorConsultas++;
        break;
        case 'UNIVEST':
            $consultasEscuela[$contadorConsultas]=' ESC IN ( 12,16 )
';
            $letrerosEscuela[$contadorConsultas] = ' Universidad
Estatul ';
            $contadorConsultas++;
        break;
        case 'NORMAL':
            $consultasEscuela[$contadorConsultas]=' ESC = 11  ';
            $letrerosEscuela[$contadorConsultas] = ' Normal Primaria
';
            $contadorConsultas++;
        break;
        case 'SEP':
            $consultasEscuela[$contadorConsultas]=' ESC = 15  ';
            $letrerosEscuela[$contadorConsultas] = ' Incorporada a la
SEP ';
            $contadorConsultas++;
        break;
        case 'OTRA':
            $consultasEscuela[$contadorConsultas]=' ESC = 17  ';
            $letrerosEscuela[$contadorConsultas] = ' Otra ';
            $contadorConsultas++;
        break;
        case 'ND':
            $consultasEscuela[$contadorConsultas]=' ESC = 0  ';
            $letrerosEscuela[$contadorConsultas] = ' No definida ';
            $contadorConsultas++;
        break;
    }
}
$procesoEscuela = array(2);
$procesoEscuela[0] = $letrerosEscuela;
$procesoEscuela[1]= $consultasEscuela;
return $procesoEscuela;
}
/**Esta función se encarga de validar que opción de
 * Sexo fue seleccionada, de acuerdo a la opción seleccionada
 * crea las diferentes "consultas" y "letreros" para este campo
 */
function procesarSexo()
{
    $consultasSexo = array();
    $letrerosSexo = array();
    $Genero = $_POST["Genero"];
    if($Genero == 'M'){
        $consultasSexo[0] = ' SEXO = "M" ';
        $letrerosSexo[0] = 'M';
    }
    elseif($Genero == 'F'){
        $consultasSexo[0] = ' SEXO = "F" ';
        $letrerosSexo[0] = 'F';
    }
}

```



```
}
elseif($Genero == 'A'){
    $consultasSexo[0] = ' SEXO = "M" ';
    $letrerosSexo[0] = 'M';
    $consultasSexo[1] = ' SEXO = "F" ';
    $letrerosSexo[1] = 'F';
}
$procesoSexo = array(2);
$procesoSexo[0] = $letrerosSexo;
$procesoSexo[1]= $consultasSexo;
return $procesoSexo;
}
?>
```

obtenerYprocesarDatos.js

```
/** Recuperamos los criterios que fueron seleccionados */
function recuperarDatos(seleccion, rangos) {
    indice = 0;
    let aux = document.getElementsByClassName(rangos);
    for (i = 0; i < aux.length; i++) {
        if (aux[i].checked) {
            seleccion[indice] = aux[i].id;
            indice++;
        }
    }
}

function deshabilitarFormulario() {
    let form = document.getElementById('formulario');
    for (i = 0; i < form.length; i++)
        document.getElementById('formulario')[i].disabled = true;
}

function obtenerPorcentaje(totales, datas, datas2) {
    for (i = 0; i < datas.length; i++)
        for (j = 0; j < datas[i].length; j++)
            datas2[i][j] = (datas[i][j] * 100) / totales[i];
    return datas2;
}

/** Procesamos los datos para después poder realizar la grafica */
function procesarDatos(datosF) {
    var bandera = datosF[2];
    var titulo = datosF[3];
    var labelsdata = new Array(datosF[0].length);
    var datas = new Array(datosF[1].length);
    var totales = new Array(datosF[1].length);
    var trimestres = [];
    var contador = 0;
    for (i = 0; i < datosF[0].length; i++)
        labelsdata[i] = datosF[0][i];
    for (i = 0; i < datosF[1].length; i++)
        datas[i] = datosF[1][i];
    for (i = 0; i < datosF[1].length; i++)
        totales[i] = 0;
    /** Crea un array donde se guardan los diferentes trimestres */
    for (i = 0; i < datas.length; i++)
        for (j = 0; j < datas[i].length; j++)
        {
            if (!trimestres.includes(datas[i][j].NTRI))
            {
                trimestres[contador] = datas[i][j].NTRI;
                contador++;
            }
        }

    trimestres.sort((a, b) => a - b);
    var datas = new Array(datas.length);
    for (i = 0; i < datas.length; i++)
        datas[i] = new Array(trimestres.length);
    var datas2 = new Array(datas.length);
    for (i = 0; i < datas.length; i++)
        datas2[i] = new Array(trimestres.length);
    for (i = 0; i < datas.length; i++)
```

```

        for(j=0;j<trimestres.length;j++)
            datas[i][j]=0;
var k = 0;
for(i=0;i<datos.length;i++)
{
    for(j=0;j<trimestres.length;j++)
    if(k<datos[i].length)
    {
        if(datos[i][k].NTRI == trimestres[j])
        {
            datas[i][j] = Number(datos[i][k].CONTEO);
            totales[i]= totales[i]+Number(datos[i][j]);
            k++;
        }
    }
    k=0;
}
datas2 = obtenerPorcentaje(totales,datas,datas2);
if(bandera == 0)
{
    crearGraficaBarras(datas,trimestres, labelsdata, titulo);
    crearGraficaArea(datas,trimestres, labelsdata, titulo);
    crearGraficaBarrasP(datas2,trimestres, labelsdata, titulo);
    crearGraficaAreaP(datas2,trimestres, labelsdata, titulo);
}

else if(bandera == 1)
{
    /**Aplicamos logaritmo base 10 a todos los valores para
    * poder realizar la comparación
    */
    for(i=0;i<datos.length;i++)
        for(j=0;j<datos[i].length;j++)
        {
            datas[i][j]= Math.log10(datos[i][j]);
        }

    crearGraficaRadar(datas,trimestres, labelsdata, titulo);
}
}

```

Apéndice C. Código fuente módulo de generación de gráficas

crearGrafica.js

```
var presets = window.chartColors;
var utils = Samples.utils;
function getLineColor(ctx) {
    return utils.color(ctx.datasetIndex);
}

function makeHalfAsOpaque(ctx) {
    return utils.transparentize(getLineColor(ctx));
}

function make20PercentOpaque(ctx) {
    return utils.transparentize(getLineColor(ctx), 0.8);
}

function crearGraficaBarras(datas, trimestres, labelsdata, titulo)
{
    var varDatasets = new Array();
    var elemento = document.getElementById("graficaBarras");
    elemento.style.display = 'block';
    for(i=0; i<labelsdata.length; i++){
        varDatasets[i] = {
            label: labelsdata[i],
            backgroundColor: getLineColor,
            data: datas[i],
        };
    }
    var barChartData = {
        labels: trimestres,
        datasets: varDatasets
    };

    var ctx = document.getElementById('graficaB').getContext('2d');
    Chart.defaults.global.defaultFontColor = 'black';
    Chart.defaults.global.defaultFontSize = 15;
    window.myBar = new Chart(ctx, {
        type: 'bar',
        data: barChartData,
        options: {
            title: {
                display: true,
                text: titulo,
                fontColor: 'black'
            },
            legend: {
                position: 'top',
            },
            tooltips: {
                mode: 'index',
                intersect: false
            },
            responsive: true,
            scales: {
                xAxes: [{
```

```

        stacked: true,
        scaleLabel: {
            display: true,
            labelString: 'Trimestres',
            fontColor: 'black'
        },
        gridLines: {
            display: true,
            color: '#cccccc',
            lineWidth: '1.5'
        }
    }],
    yAxes: [{
        stacked: true,
        scaleLabel: {
            display: true,
            labelString: 'Alumnos',
            fontColor: 'black'
        },
        gridLines: {
            display: true,
            color: '#cccccc',
            lineWidth: '1.5'
        }
    }
    ]
    },
    });
}

function crearGraficaBarrasP(datas, trimestres, labelsdata , titulo)
{
    var varDatasets = new Array();
    var elemento = document.getElementById("graficaBarrasP");
    elemento.style.display = 'block';
    for(i=0; i<labelsdata.length; i++){
        varDatasets[i] = {
            label: labelsdata[i],
            backgroundColor: getLineColor,
            data: datas[i],
        };
    }
    var barChartData = {
        labels: trimestres,
        datasets: varDatasets
    };

    var ctx = document.getElementById('graficaBP').getContext('2d');
    Chart.defaults.global.defaultFontColor = 'black';
    Chart.defaults.global.defaultFontSize = 15;
    window.myBar = new Chart(ctx, {
        type: 'bar',
        data: barChartData,
        options: {
            title: {
                display: true,
                text: titulo,
                fontColor: 'black'
            },
        },
    });
}

```

```

        legend: {
            position: 'top',
        },
        tooltips: {
            mode: 'index',
            intersect: false
        },
        responsive: true,
        scales: {
            xAxes: [{
                stacked: true,
                scaleLabel: {
                    display: true,
                    labelString: 'Trimestres',
                    fontColor: 'black'
                },
                gridLines: {
                    display: true,
                    color: '#cccccc',
                    lineWidth: '1.5'
                }
            }],
            yAxes: [{
                stacked: true,
                scaleLabel: {
                    display: true,
                    labelString: 'Alumnos (en porcentaje)',
                    fontColor: 'black'
                },
                gridLines: {
                    display: true,
                    color: '#cccccc',
                    lineWidth: '1.5'
                }
            }],
        },
    });
}

function crearGraficaArea(datas, trimestres, labelsdata, titulo)
{
    var varDatasets = new Array();
    let divGraficaArea = document.getElementById('graficaArea');
    divGraficaArea.style.display = 'block';
    for (i=0; i<labelsdata.length; i++) {
        varDatasets[i] = {
            label: labelsdata[i],
            data: datas[i],
        };
    }
    var barChartData = {
        labels: trimestres,
        datasets: varDatasets
    };

    var ctx = document.getElementById('graficaA').getContext('2d');
    Chart.defaults.global.defaultFontColor = 'black';
    Chart.defaults.global.defaultFontSize = 15;

```

```

window.myLine = new Chart(ctx, {
  type: 'line',
  data: barChartData,
  options: {
    title: {
      display: true,
      text: titulo,
      fontColor: 'black'
    },
    legend: {
      position: 'top',
    },
    tooltips: {
      mode: 'index',
      intersect: false
    },
    responsive: true,
    hover: {
      mode: 'index'
    },
    elements: {
      line: {
        backgroundColor: make20PercentOpaque,
        borderColor: getLineColor,
      },
      point: {
        backgroundColor: getLineColor,
        hoverBackgroundColor: makeHalfAsOpaque,
      }
    },
    scales: {
      xAxes: [{
        stacked: false,
        scaleLabel: {
          display: true,
          labelString: 'Trimestres',
          fontColor: 'black'
        },
        gridLines: {
          display: true,
          color: '#cccccc',
          lineWidth: '1.5'
        }
      }],
      yAxes: [{
        stacked: false,
        scaleLabel: {
          display: true,
          labelString: 'Alumnos',
          fontColor: 'black'
        },
        gridLines: {
          display: true,
          color: '#cccccc',
          lineWidth: '1.5'
        }
      }],
    },
  },
});

```

```

    });
}

function crearGraficaAreaP(datas, trimestres, labelsdata, titulo)
{
    var varDatasets = new Array();
    let divGraficaArea = document.getElementById('graficaAreaP');
    divGraficaArea.style.display = 'block';
    for(i=0; i<labelsdata.length; i++){
        varDatasets[i] = {
            label: labelsdata[i],
            data: datas[i],
        };
    }
    var barChartData = {
        labels: trimestres,
        datasets: varDatasets
    };

    var ctx = document.getElementById('graficaAP').getContext('2d');
    Chart.defaults.global.defaultFontColor = 'black';
    Chart.defaults.global.defaultFontSize = 15;
    window.myLine = new Chart(ctx, {
        type: 'line',
        data: barChartData,
        options: {
            title: {
                display: true,
                text: titulo,
                fontColor: 'black'
            },
            legend: {
                position: 'top',
            },
            tooltips: {
                mode: 'index',
                intersect: false
            },
            responsive: true,
            hover: {
                mode: 'index'
            },
            elements: {
                line: {
                    backgroundColor: make20PercentOpaque,
                    borderColor: getLineColor,
                },
                point: {
                    backgroundColor: getLineColor,
                    hoverBackgroundColor: makeHalfAsOpaque,
                }
            },
            scales: {
                xAxes: [{
                    stacked: false,
                    scaleLabel: {
                        display: true,
                        labelString: 'Trimestres',
                    }
                }
            ]
        }
    });
}

```



```

        point: {
            backgroundColor: getLineColor,
            hoverBackgroundColor: makeHalfAsOpaque,
        }
    },
    scales: {
        ticks: {
            beginAtZero: true
        },
        gridLines: {
            display: true,
            color: 'pink',
            lineWidth: '11.5'
        }
    },
    plugins: {
        filler: {
            propagate: false
        },
        'samples-filler-analyser': {
            target: 'chart-analyser'
        }
    }
};

var data = {
    labels: trimestres,
    datasets: varDatasets
};

var chart = new Chart('graficaR', {
    type: 'radar',
    data: data,
    options: options
});
}

```

Apéndice D. Código fuente módulo de exportación de gráficas

exportarGrafica.js

```
/**le asignamos un evento al boton Descargar PDF correspondiente a la grafica de barras*/
document.getElementById('PDFB').addEventListener("click", function(e){
    downloadPDF('graficaB');});
/**le asignamos un evento al boton Descargar PDF correspondiente a la grafica de area*/
document.getElementById('PDFA').addEventListener("click", function(e){
    downloadPDF('graficaA');});
/**le asignamos un evento al boton Descargar PDF correspondiente a la grafica radar*/
document.getElementById('PDFR').addEventListener("click", function(e){
    downloadPDF('graficaR');});
/**le asignamos un evento al boton Descargar PNG correspondiente a la grafica de barras*/
document.getElementById('PNGB').addEventListener("click", function(e){
    downloadPNG('graficaB');});
/**le asignamos un evento al boton Descargar PNG correspondiente a la grafica de area*/
document.getElementById('PNGA').addEventListener("click", function(e){
    downloadPNG('graficaA');});
/**le asignamos un evento al boton Descargar PDF correspondiente a la grafica de barras
    * en porcentaje
*/
document.getElementById('PDFBP').addEventListener("click", function(e){
    downloadPDF('graficaBP');});
/**le asignamos un evento al boton Descargar PDF correspondiente a la grafica de
    * area con porcentaje
*/
document.getElementById('PDFAP').addEventListener("click", function(e){
    downloadPDF('graficaAP');});
/**le asignamos un evento al boton Descargar PNG correspondiente a la grafica de barras
    * con porcentaje
*/
document.getElementById('PNGBP').addEventListener("click", function(e){
    downloadPNG('graficaBP');});
/**le asignamos un evento al boton Descargar PNG correspondiente a la grafica de area
    * con porcentaje*/
document.getElementById('PNGAP').addEventListener("click", function(e){
    downloadPNG('graficaAP');});
/**le asignamos un evento al boton Descargar PNG correspondiente a la grafica radar*/
document.getElementById('PNGR').addEventListener("click", function(e){
    downloadPNG('graficaR');});

//descargar pdf
function downloadPDF(nombreGrafica) {
    var canvas = document.getElementById(nombreGrafica);
    //creamos una imagen
```

```

var canvasImg = canvas.toDataURL();

//creamos el PDF desde la imagen
var doc = new jsPDF('landscape');
doc.setFontSize(20);
doc.addImage(canvasImg, 'JPEG', 10, 10, 280, 150 );
doc.save('grafica.pdf');
}

//descargar PNG
function downloadPNG(nombreGrafica)
{
    var canvas = document.getElementById(nombreGrafica);
    /**crea un objeto Blob que representa la imagen contenida en el
    canvas */
    canvas.toBlob((blob) =>{
        const a = document.createElement('a');
        document.body.append(a);
        a.download = 'grafica.png';
        a.href = URL.createObjectURL(blob);
        a.click();
        a.remove();
    });
}

```