

INFORMÁTICA 506-B

CONTROL ESCOLAR

INTEGRANTES:

- Erik Jarquín Sánchez (Administrador)
- Luis Lennin Villarreal Castro (Programador Junior)
- Guillermo Daniel Cruz Ortega (Tester)

Índice general

1. Introducción	8
1.1. Personal involucrado	9
2. Especificación de requisitos	10
2.1. Propósito	10
2.2. Alcance (Ámbito del proyecto)	10
2.3. Viabilidad y factibilidad	10
2.4. Viabilidad	11
2.5. Factibilidad	11
2.6. Requisitos	12
2.6.1. Requisitos funcionales	12
2.6.2. Requisitos no funcionales	14
2.6.3. Mapa de navegación	16
3. Análisis	17
3.1. Introducción	17
3.2. Diagrama de estructura de UML	17
3.2.1. Diagrama de Flujo de datos	17
3.2.2. Diagrama de clases	19
3.2.3. Diagrama de Objetos	19
3.2.4. Diagrama de Componentes	38
3.2.5. Diagrama de Despliegue	42
3.3. Diagramas de comportamiento	44
3.3.1. Diagramas de casos de uso	44
3.3.2. Diagrama de Estado	110
3.3.3. Diagrama de Secuencia	113
3.3.4. Diagrama de Colaboración	118
3.3.5. Diagrama de Actividades	121
4. Diseño	122
4.1. Introducción	122
4.2. Descripción del sistema	123
4.3. Arquitectura del sistema	123
4.3.1. Arquitectura física	123
4.3.2. Arquitectura lógica	138
4.4. Estándar de codificación	138
4.5. Bases de datos	139
4.5.1. Diseño de datos	139

4.5.2.	Diagrama conceptual	139
4.5.3.	Diagrama Entidad-Relación	140
4.5.4.	Diagrama de Modelo Relacional	140
4.5.5.	Diagrama físico	141
4.5.6.	Diccionario de datos	143
5.	Implementación	146
5.1.	Intruducción	146
5.2.	Código fuente	146
6.	Conclusión	218
6.1.	Referencias	219
6.2.	Anexo 1.PSP	220
6.3.	Anexo 2.Galería de imágenes	221

Índice de figuras

1.1. Colaboradores	9
2.1. Tabla de Autentificación (Requisitos funcinales)	12
2.2. Tabla de dar de baja alumno (Requisitos funcinales)	12
2.3. Tabla de modificar perfil (Requisitos funcinales)	13
2.4. Tabla de nuevo alumno (Requisitos funcinales)	13
2.5. Tabla de registros de usuario (Requisitos funcinales)	13
2.6. Tabla de confidencialidad (Requisitos no funcinales)	14
2.7. Tabla de look and feel (Requisitos no funcinales)	14
2.8. Tabla de restricción de contenido (Requisitos no funcinales)	14
2.9. Tabla de robustez (Requisitos no funcinales)	15
2.10. Tabla de seguridad (Requisitos no funcinales)	15
2.11. Mapa de navegación	16
3.1. Diagrama de flujo de datos nivel 0	17
3.2. Diagrama de flujo nivel 1	18
3.3. Diagrama de Flujo de datos Iniciar sesión.	18
3.4. Diagrama de Flujo de datos Inicio.	19
3.5. Diagramas Objetos parte 1	20
3.6. Diagramas Objetos código 1	21
3.7. Diagramas Objetos parte 2	22
3.8. Diagramas Objetos código 2	23
3.9. Diagramas Objetos parte 3	24
3.10. Diagramas Objetos código 3	25
3.11. Diagramas Objetos código 4	26
3.12. Diagramas Objetos parte 5	27
3.13. Diagramas Objetos código 5	28
3.14. Diagramas Objetos parte 6	29
3.15. Diagramas Objetos código 6	30
3.16. Diagramas Objetos código 7	31
3.17. Diagramas Objetos código 8	32
3.18. Diagramas Objetos código 9	33
3.19. Diagramas Objetos código 10	34
3.20. Diagramas Objetos código 11	35
3.21. Diagramas Objetos código 12	36
3.22. Diagramas Objetos código 13	37
3.23. Diagramas Objetos código 14	38
3.24. Diagrama de Componentes	39
3.25. Diagrama de Componentes código	40

3.26. Diagrama de Componentes código	41
3.27. Diagrama de despliegue.	42
3.28. Diagrama de Despliegue código	43
3.29. Diagrama de casos de uso de inicio	44
3.30. Tabla del caso de uso de Login.	45
3.31. Código de Login	46
3.32. Diagrama de casos de uso de consulta	47
3.33. Tabla del caso de uso de Consulta	48
3.34. Código de Consulta	49
3.35. En la figura se muestra el caso de uso de Escoger Materia Primero	50
3.36. Tabla del caso de uso de Escoger Materia Primero.	51
3.37. Código de Escoger Materia Primero	52
3.38. En la figura se muestra el caso de uso de Matematicas Uno.	53
3.39. Tabla del caso de uso de Matematicas Uno	54
3.40. Código de Matematicas Uno	55
3.41. En la figura se muestra el caso de uso de Español Uno	56
3.42. Tabla del caso de uso de Español Uno.	57
3.43. Código de Español Uno	58
3.44. En la figura se muestra el caso de uso de Ciencias Uno	59
3.45. Tabla del caso de uso de Ciencias Uno.	60
3.46. Código de Ciencias Uno	61
3.47. En la figura se muestra el caso de uso de Historia	62
3.48. Tabla del caso de uso de Historia	63
3.49. Código de Historia	64
3.50. En la figura se muestra el caso de uso de Geografía	65
3.51. Tabla del caso de uso de Geografía.	66
3.52. Código de Actualizacion Geografía	67
3.53. En la figura se muestra el caso de uso de Actualizacion Calificaciones.	68
3.54. Tabla del caso de uso de Actualizacion Calificaciones.	69
3.55. Código de Actualizacion Calificaciones	70
3.56. En la figura se muestra el caso de uso de Escoger Materia Segundo	71
3.57. Tabla del caso de uso de Escoger Materia Segundo	72
3.58. Código de Escoger Materia Segundo	73
3.59. En la figura se muestra el caso de uso de Matematicas Dos.	74
3.60. Tabla del caso de uso de Matematicas Dos.	75
3.61. Código de Matematicas Dos	76
3.62. En la figura se muestra el caso de uso de Español Dos	77
3.63. Tabla del caso de uso de Español Dos.	78
3.64. Código de Español Dos	79
3.65. En la figura se muestra el caso de uso de Ingles Uno.	80
3.66. Tabla del caso de uso de Ingles Uno.	81
3.67. Código de Ingles Uno	82
3.68. En la figura se muestra el caso de uso de Cívica y Ética.	83
3.69. Tabla del caso de uso de Cívica y Ética.	84
3.70. Código de Cívica y Ética	85
3.71. En la figura se muestra el caso de uso de Actualizacion Calificaciones.	86
3.72. Tabla del caso de uso de Actualizacion Calificaciones.	87
3.73. Código de Actualizacion Calificaciones	88
3.74. En la figura se muestra el caso de uso de Escoger Materia Tercero.	89

3.75. Tabla del caso de uso de Escoger Materia Tercero.	90
3.76. Código de Escoger Materia Tercero	91
3.77. En la figura se muestra el caso de uso de Matematicas Tres	92
3.78. Tabla del caso de uso de Matematicas Tres.	93
3.79. Código de Matematicas Tres	94
3.80. En la figura se muestra el caso de uso de Ingles Dos	95
3.81. Tabla del caso de uso de Ingles Dos	96
3.82. Código de Ingles Dos	97
3.83. En la figura se muestra el caso de uso de Tecnologías	98
3.84. Tabla del caso de uso de Tecnologías	99
3.85. Código de Tecnologías	100
3.86. En la figura se muestra el caso de uso de Vida Saludable.	101
3.87. Tabla del caso de uso de Vida Saludable..	102
3.88. Código de Vida Saludable	103
3.89. En la figura se muestra el caso de uso de Química.	104
3.90. Tabla de Química	105
3.91. Código de Química	106
3.92. En la figura se muestra el caso de uso de Secretaria 1.5.	107
3.93. Tabla del caso de Secretaria 1.5	108
3.94. Código de Secretaria 1.5	109
3.95. Diagrama de estado	110
3.96. Diagrama de estado código 1	111
3.97. Diagrama de estado código 2	112
3.98. Diagrama de Secuencia parte 1	113
3.99. Diagrama de Secuencia parte 2	114
3.100Diagrama de Secuencia parte 1 código	115
3.101Diagrama de Secuencia parte 2 código	116
3.102Diagrama de Secuencia parte 3 código	117
3.103Diagrama de Colaboración.	118
3.104Diagrama de Colaboración.	119
3.105Diagrama de Colaboración.	120
3.106Diagrama de Actividades	121
4.1. Imagen de inicio	123
4.2. Imagen pantalla de alumnos	124
4.3. Imagen pantalla de maestros 1	124
4.4. Imagen pantalla de subir calificaciones 1	125
4.5. Imagen pantalla de maestros 2	125
4.6. Imagen pantalla de subir calificaciones 2	126
4.7. Imagen pantalla de maestros 3	126
4.8. Imagen pantalla de subir calificaciones 3	127
4.9. Imagen ventana secretaria	127
4.10. Imagen modificar grupo	128
4.11. Imagen modificar tabla secretaria	128
4.12. Imagen registrar alumno	129
4.13. Imagen pantalla de materias 1	129
4.14. Imagen ciencias 1	130
4.15. Imagen girografía	130
4.16. Imagen matemáticas 1	131
4.17. Imagen español 1	131

4.18. Imagen pantalla de materias 2	132
4.19. Imagen ciencias 2	132
4.20. Imagen español 2	133
4.21. Imagen cívica y ética	133
4.22. Imagen inglés 1	134
4.23. Imagen matemáticas 2	134
4.24. Imagen materias 3	135
4.25. Imagen inglés 2	135
4.26. Imagen matemáticas 3	136
4.27. Imagen química	136
4.28. Imagen tecnologías	137
4.29. Imagen vida saludable	137
4.30. Estándar de codificación	138
4.31. Diagrama Entidad-Relación del proyecto del control escolar.	140
4.32. Diagrama de modelo Relacional del proyecto del control escolar.	140
4.33. Diagrama físico de la creación de base de datos.	141
4.34. Diagrama físico de la creación de base de datos.	142
4.35. Tabla Alumnos	143
4.36. Tabla Usuarios	143
4.37. Tabla Calificaciones	144
4.38. Tabla Grupo	144
4.39. Tabla Materias	144
4.40. Tabla Roles	145
 6.1. Gráfica del uso de PSP	220
6.2. Evidencia de la colaboración del equipo	221
6.3. Evidencia de la colaboración de Guillermo y Erik	222
6.4. Evidencia de la colaboración Lennin y Erik	223
6.5. Evidencia de la colaboración de Lennin y Erik	224

Capítulo 1

Introducción

El proyecto es un software de control de escuela, en este caso de escuela secundaria, con el propósito de agilizar el control del alumnado, un software de control viene muy bien para cualquier ámbito, ya sea el industrial, ventas o escolar en este caso, ya que permite tener una visión de cómo van marchando las cosas con la compañía o institución, y así saber el futuro que te espera. En el caso de una institución de ámbito escolar como es el caso, conviene saber el rendimiento de los alumnos, los grupos y ver cuáles son los puntos débiles, como por ejemplo en qué períodos de exámenes sufren más, qué materias necesitan reforzar, etc. Ese tipo de información ayuda a agilizar la enseñanza y a tomar medidas para mejorar la calidad de esta, igual facilita a los profesores y padres de familia. A los profesores les ayuda mucho, ya

que les permite agilizar el subir calificaciones o modificar datos de los alumnos (calificaciones, nombre, materias), así se tiene un control eficiente y concentrado. A los alumnos les ayuda como un incentivo, ya que pueden ver en cualquier momento su progreso y ver qué materias tiene que estudiar más para obtener los créditos necesarios. A los padres de familia les ayuda en seguir el progreso de sus hijos y ayudarlos a consolidar las calificaciones de sus materias. También ayudaría a sacar los promedios de los alumnos al igual que los promedios generales por grupo dando una visión más amplia de las condiciones de cada salón y de cada maestro.

1.1. Personal involucrado

	Nombre	Roles	Descripción
	Lirio Ruiz Guerra	DBA	<p>Grado Académico</p> <ul style="list-style-type: none">• Maestría en Ciencias de la Computación• Ingeniería en Sistemas Computacionales <p>Líneas de investigación Actuales</p> <ul style="list-style-type: none">• Tecnología educativa• Procesos de software
	Rolando Pedro Gabriel	Usuario final	<p>Grado Académico</p> <ul style="list-style-type: none">• Maestría en Tecnologías de Cómputo Aplicado• Licenciatura en informática <p>Línea de Investigación actual</p> <ul style="list-style-type: none">• Tecnologías de la información y comunicación
	Everardo de Jesus Pacheco Antonio	Súper Administrador	<p>Grado Académico</p> <ul style="list-style-type: none">• Maestría en Tecnología Educativa• Licenciatura en Informática <p>Líneas de Investigación Actuales</p> <ul style="list-style-type: none">• Tecnología Educativa• Tecnologías de Información y Comunicación

Figura 1.1: Colaboradores

Capítulo 2

Especificación de requisitos

2.1. Propósito

El propósito de este proyecto es fabricar y proveer un software eficaz en el manejo de una escuela pequeña (escuela secundaria en este caso), asegurar un buen funcionamiento y cubrir la mayoría de las necesidades de un sistema escolar.

2.2. Alcance (Ámbito del proyecto)

El alcance del proyecto es de un plazo de 3 meses para la entrega en general, se busca trabajar en cuatro bloques que comprenderá cada uno de ellos un tiempo en el que se entregará un avance, dicho avance estará monitoreado por un profesional el cual guie el rumbo del proyecto, detectando errores o implementando opiniones para el mejoramiento del programa, con los bloques de tiempos y las entregas que se realizan, estaremos con un proyecto terminado hasta el tercer bloque de tiempo, donde se entregará ya el trabajo realizado, para la cuarta y última entrega se mostrara el proyecto ya terminado y listo para funcionar

Este sistema tendrá el objetivo de ser una gran herramienta que pueda ayudar al usuario llevar una mejor organización de sus operaciones en la escuela y también está claro que esta también debe cumplir las expectativas de cliente en cuanto al diseño del software, ya que el diseño se efectuará de forma que pueda ser más agradable y entendible para el usuario.

2.3. Viabilidad y factibilidad

Para la realización de este sistema primero tenemos que identificar los pasos que nos llevará a realizar

un buen desarrollo del proyecto, debido a que se deben de ir identificando los problemas futuros que surgirán en el transcurso de la realización del proyecto. En este estudio tendremos que invertir mucho tiempo para hacer pruebas y determinar si puede llegar a realizar de una manera correcta y buscar diferentes soluciones, cosas que se ajusten a nuestro presupuesto sin tener dificultades en el futuro. Una cosa que hemos mencionado antes, es que nuestro sistema tendrá implementado el ingreso al usuario, en cuestión técnica se utilizará una función para comparar los datos del usuario con los que está ingresando.

Recabando todas las secciones mencionadas anteriormente se llegó a la conclusión de que se tendrá que invertir la mayor parte del tiempo en la codificación y el diseño de nuestra interfaz, en cuestión económica no se tendrá que invertir demasiado, debido a que únicamente se estará utilizando internet y equipo de cómputo, mencionamos en este último punto porque el equipo que estaremos utilizando sufrirá desgaste y disminuirá su costo, las secciones que incluiremos no generarán costo. De acuerdo con el análisis mencionado en los párrafos anteriores podemos ver que nuestro sistema si lo podemos llevar a cabo ya que contamos con el presupuesto y las herramientas necesarias para desarrollarlo, así como el conocimiento necesario para elaborarlo sin ningún contratiempo. Nos llevará mucho tiempo en realizarlo, pero organizaremos nuestros tiempos para avanzar de una manera segura y viable.

2.4. Viabilidad

Técnico:

- Como estudiantes de informática es posible que podamos llevar a cabo el proyecto, ya que contamos hasta el momento un poco de conocimiento en la materia de programación, y lo que planeamos llevar a cabo es una aplicación en el lenguaje de programación Java y hasta el momento estamos estudiando dicho lenguaje.
- Referente a las herramientas para llevar a cabo dicho proyecto, se puede decir que contamos con lo suficiente porque se necesitarán computadores e IDES de programación en Java con las cuales si cuenta cada integrante del equipo.
- Un punto a considerar también sería la energía personal de cada integrante con esto nos referimos al esfuerzo que cada uno puede poner de su parte y también el esfuerzo que requerirá el proyecto. Referente al esfuerzo que se necesitará para el proyecto podríamos decir que si es algo considerable porque el proyecto debe contar con muchas especificaciones en el ámbito de programación que son difíciles, y esto nos lleva al esfuerzo personal que si será muy desgastante porque contamos con poco tiempo para la entrega.

Económico:

- Al referirnos a la viabilidad económica hablamos del capital o recursos económicos que requiere el proyecto para ponerlo en marcha.
- Para nuestro proyecto podría decirse que no se necesita mucho capital hasta donde tenemos planeado realizarlo, tal vez un poco para hacer un estudio de que se requiere en una escuela, por lo que entonces si es posible llevar a cabo este proyecto.

2.5. Factibilidad

La disponibilidad de los recursos que contamos con el proyecto de “Control escolar” es más que un recurso lógico-mental, ya que se llevará a cabo la programación de la interfaz, tal vez con los recursos que necesitemos a parte del trabajo lógico, sería el costo de un espacio en la base de datos si se piensa conectar a un servidor exterior para guardar los datos, se tendría que ver cuánto cuesta un paquete para almacenar los datos escolares en la base de datos, dentro del marco de posibilidades, si se puede realizar el proyecto y lo que causaría algunos problemas sería programar la base de datos y hacer que

se identifiquen los datos en Java, costará porque se realiza un trabajo mental, pero con un horario de trabajo adecuado se podrá realizar el proyecto.

2.6. Requisitos

2.6.1. Requisitos funcionales

Código del Requerimiento	RF02
Nombre	Autentificación
Propósito	Iniciar sesión.
Descripción	Una vez ubicados en la página de inicio de sesión, el usuario debe diligenciar sus credenciales en los respectivos campos y finalizar pulsando el botón de inicio de sesión.
Entrada	Credenciales o datos respectivos del usuario.
Salida	Redirección a la página principal del usuario.
Prioridad	Alta

Figura 2.1: Tabla de Autentificación (Requisitos funcionales)

Código del Requerimiento	RF05
Nombre	Dar de baja alumno (eliminar fila)
Propósito	Eliminar una fila de la tabla y el contenido del alumno.
Descripción	Se selecciona una fila de la tabla y se eliminan todos los campos relacionados con ese alumno.
Entrada	Tabla de alumnos.
Salida	Mensaje de acción realizada con éxito.
Prioridad	Alta

Figura 2.2: Tabla de dar de baja alumno (Requisitos funcionales)

Código del Requerimiento	RF03
Nombre	Modificar Perfil
Propósito	Corregir o actualizar información de cualquier tipo de usuario.
Descripción	A través del componente de perfil, cada usuario del sistema tiene la facilidad de modificar su información, incluida su contraseña.
Entrada	Formulario de Perfil.
Salida	Mensaje de acción satisfactoria.
Prioridad	Alta

Figura 2.3: Tabla de modificar perfil (Requisitos funcionales)

Código del Requerimiento	RF04
Nombre	Nuevo Alumno
Propósito	Crear un nuevo espacio en la tabla de alumnos por medio de un formulario que pide los datos del alumno a agregar
Descripción	A través de un cuestionario por el cual se accede por un botón el personal administrativo podrá registrar un nuevo alumno el cual sera agregado a la tabla correspondiente.
Entrada	Formulario de registro.
Salida	Mensaje de acción realizada con éxito.
Prioridad	Alta

Figura 2.4: Tabla de nuevo alumno (Requisitos funcionales)

Código del Requerimiento	RF01
Nombre	Registro de Usuario
Propósito	Crear un nuevo usuario que controle el registro.
Descripción	Una vez ubicados en la página de inicio de sesión, el potencial usuario debe dirigirse al hipervínculo que indica la creación de una nueva cuenta, diligenciar el formulario con sus respectivos campos obligatorios y proceder a finalizar el proceso pulsando en el botón indicado.
Entrada	Formulario de Registro diligenciado con información del usuario.
Salida	- Mensaje de bienvenida al correo electrónico registrado. - Redirección a la primer página.
Prioridad	Alta

Figura 2.5: Tabla de registros de usuario (Requisitos funcionales)

2.6.2. Requisitos no funcionales

Código del Requerimiento	RNF04
Nombre	Confidencialidad
Descripción	El acceso a cada página del aplicativo está determinado por el rol del usuario.
Prioridad	Alta

Figura 2.6: Tabla de cofidencialidad (Requisitos no funcinales)

Código del Requerimiento	RNF01
Nombre	Look and Feel (Vista de la interfaz)
Descripción	El aspecto del aplicativo debe ser consistente en todas sus páginas, además de amigable e intuitivo hacia el usuario.
Prioridad	Alta

Figura 2.7: Tabla de look and feel (Requisitos no funcinales)

Código del Requerimiento	RNF03
Nombre	Restricción de Contenido
Descripción	Toda la información otorgada por los usuarios se manipulará únicamente con fines corporativos y de manera limpia.
Prioridad	Alta

Figura 2.8: Tabla de restricción de contenido (Requisitos no funcinales)

Código del Requerimiento	RNF05
Nombre	Robustez
Descripción	El software debe ser capaz de manejar toda la información recolectada a través del tiempo con fluidez.
Prioridad	Alta

Figura 2.9: Tabla de robustez (Requisitos no funcinales)

Código del Requerimiento	RNF02
Nombre	Seguridad
Descripción	El protocolo o librería usado para manejar la seguridad en la sesión del usuario debe ser lo suficientemente confiable.
Prioridad	Alta

Figura 2.10: Tabla de seguridad (Requisitos no funcinales)

2.6.3. Mapa de navegación

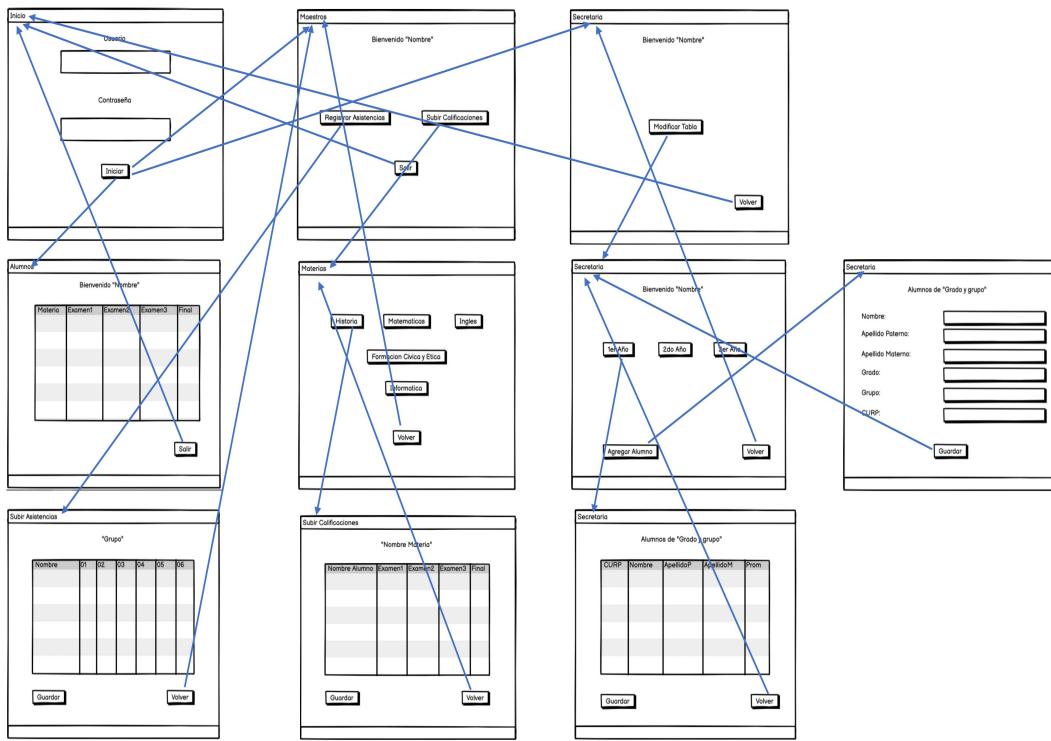


Figura 2.11: Mapa de navegación.

Capítulo 3

Análisis

3.1. Introducción

En el análisis de requisitos se detalla aquellos componentes del sistema que vamos a implementar mediante software. El objetivo es comprender cuales son los datos que se van a manejar, cual va a ser la función que tiene que cumplir el software, cuáles son las interfaces requeridas y cual es el rendimiento que se espera lograr. En este capítulo se presentan los distintos diagramas que fueron implementados mediante la utilización de la herramienta PlanUML. Se especifica el flujo de ejecución que tendrá el sistema mediante la representación gráfica que fueron implementado con la ayuda de la herramienta.

3.2. Diagrama de estructura de UML

3.2.1. Diagrama de Flujo de datos

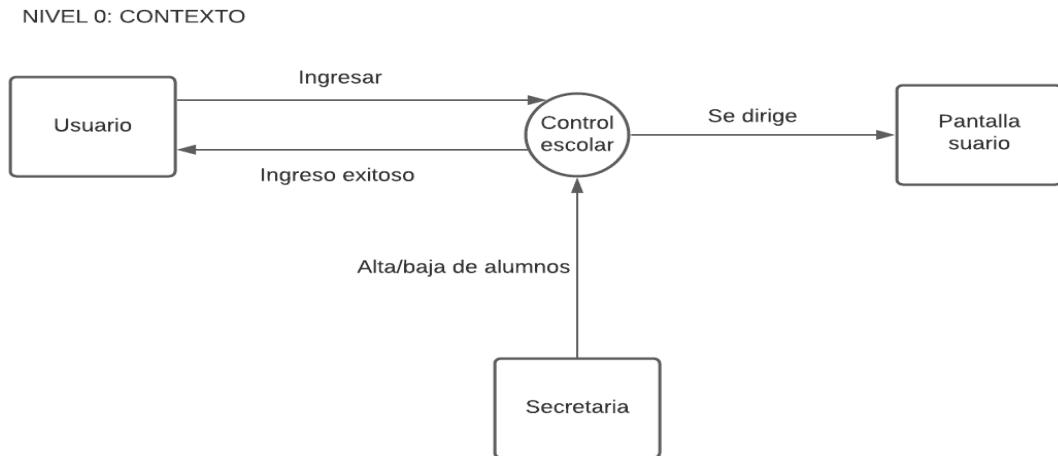


Figura 3.1: Diagrama de flujo de datos nivel 0

Se hace un diagrama de manera muy general de como está establecido el proyecto y de como será su funcionalidad.

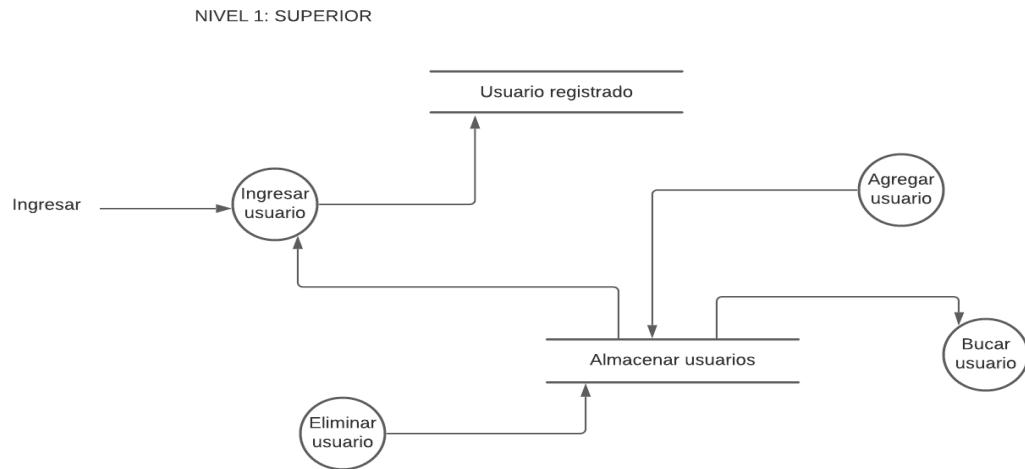


Figura 3.2: Diagrama de flujo nivel 1

Se detalla más a fondo las características del sistema, tomando en cuenta que se debe hacer en casos específicos y si hay alguna falla o no.

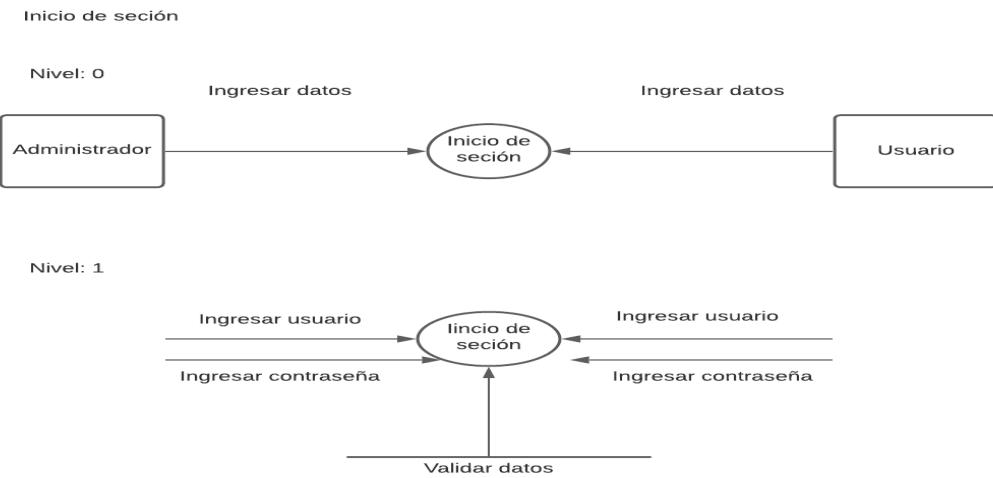


Figura 3.3: Diagrama de Flujo de datos Iniciar sesión.

En el diagrama de Inicio de sesión se puede observar el flujo que debe seguir el usuario como el administrador en este caso la secretaria, el alumno o maestro en sí para poder iniciar sesión.

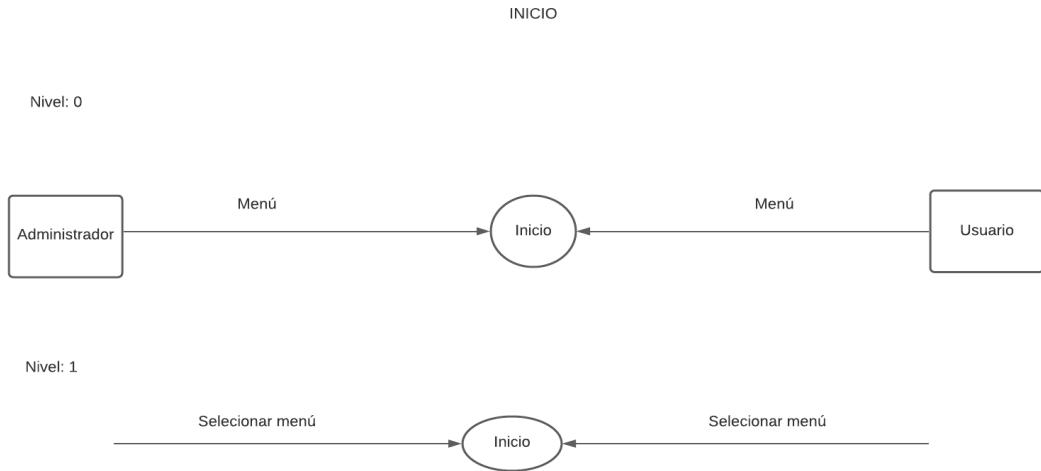


Figura 3.4: Diagrama de Flujo de datos Inicio.

Secretaria, alumno o maestro entra en su respectivo menú.

3.2.2. Diagrama de clases

El Diagrama de Clases nos permite trazar claramente la estructura de nuestro sistema de ventas al modelar sus clases, atributos, operaciones y relaciones entre objetos ya que es muy esencial para la creacion de nuestro sistema. En este caso tomamos como uno mismo el diagrama de objetos y de clases

3.2.3. Diagrama de Objetos

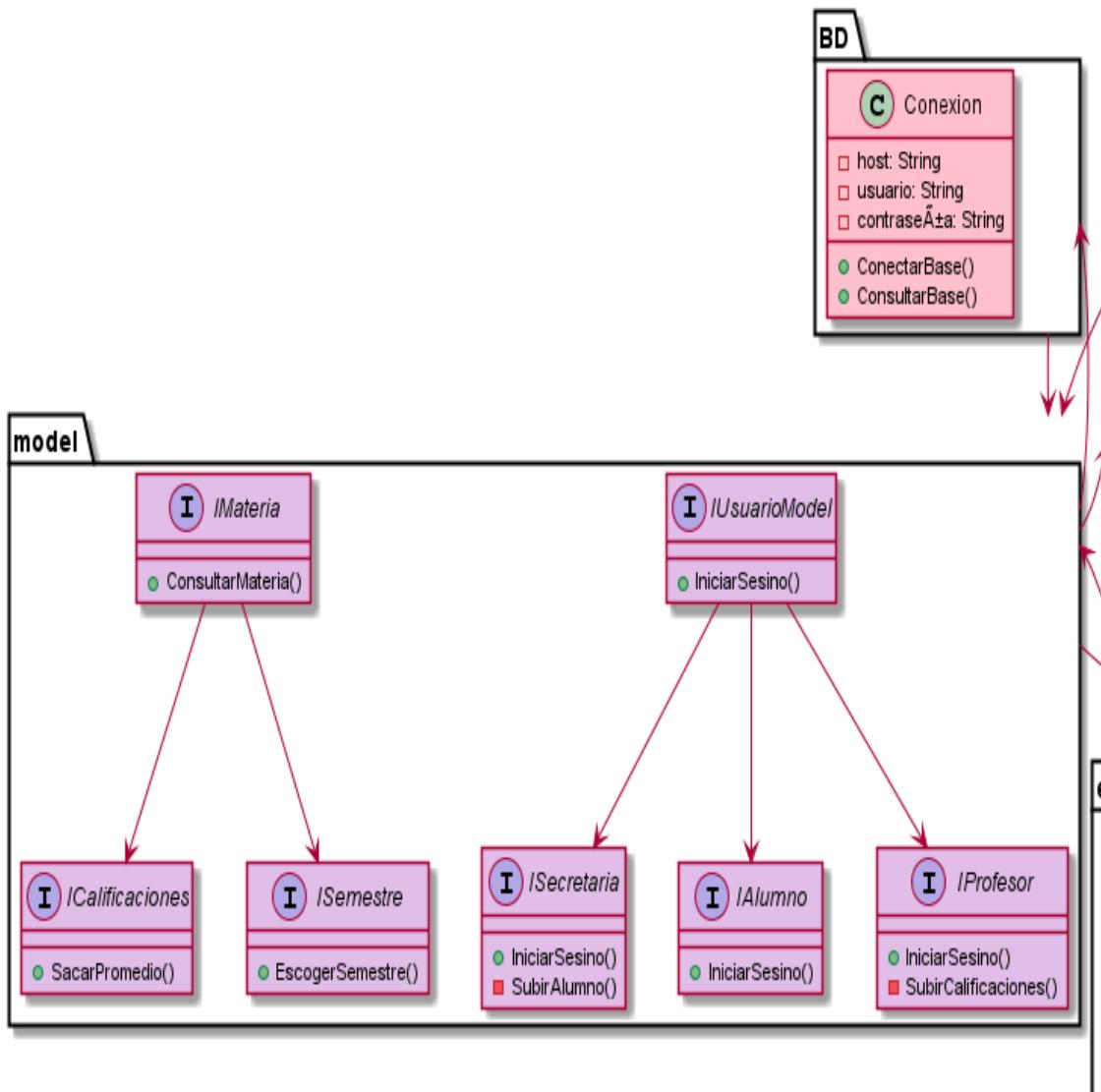


Figura 3.5: Diagramas Objetos parte 1

Este diagrama de objetos nos muestra una instancia de nuestros diagramas de clases y se enfoca en los atributos y sus relaciones que tienen los objetos en nuestro sistema y así como también nos describe la estructura estática de nuestro sistema.

```
@startuml
class BD.Conexion #pink{
    -host: String
    -usuario: String
    -contraseña: String
    +ConectarBase ()
    +ConsultarBase ()
}

class entity.Usuario #pink{
    -nombre: String
    -contraseña: String
}

class entity.Alumno #pink{
    -nombre: String
    -contraseña: String
}

class entity.Profesor #pink{
    -nombre: String
    -contraseña: String
}

class entity.Secretaria #pink{
    -nombre: String
    -contraseña: String
}

class entity.Materia #pink{
    -codigo: String
    -nombre: String
    -semestre: Semestre
    -calificacion: Calificaciones
}
```

Figura 3.6: Diagramas Objetos código 1

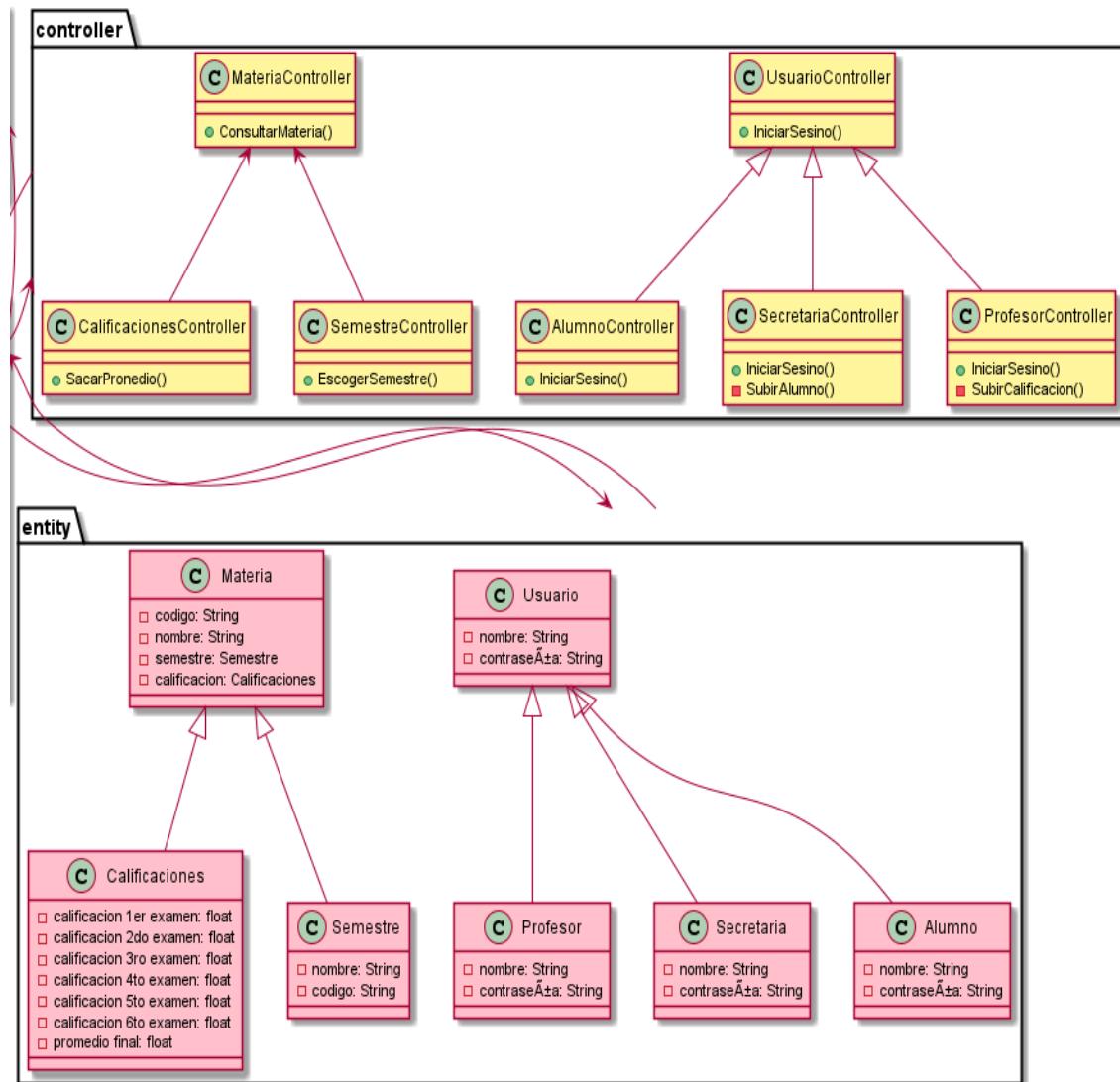


Figura 3.7: Diagramas Objetos parte 2

Este diagrama de objetos nos muestra una instancia de nuestros diagramas de clases y se enfoca en los atributos y sus relaciones que tienen los objetos en nuestro sistema y así como también nos describe la estructura estática de nuestro sistema.

```
class entity.Calificaciones #pink{
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}
class entity.Semestre #pink{
    -nombre: String
    -codigo: String
}
entity.Usuario <|-- entity.Alumno
entity.Usuario <|-- entity.Profesor
entity.Usuario <|-- entity.Secretaria
entity.Materia <|-- entity.Calificaciones
entity.Materia <|-- entity.Semestre

interface model.IUsuarioModel #elbee7 {
    +IniciarSesino()
}
interface model.IAlumno #elbee7 {
    +IniciarSesino()
}
interface model.IProfesor #elbee7 {
    +IniciarSesino()
    -SubirCalificaciones()
}
```

Figura 3.8: Diagramas Objetos código 2

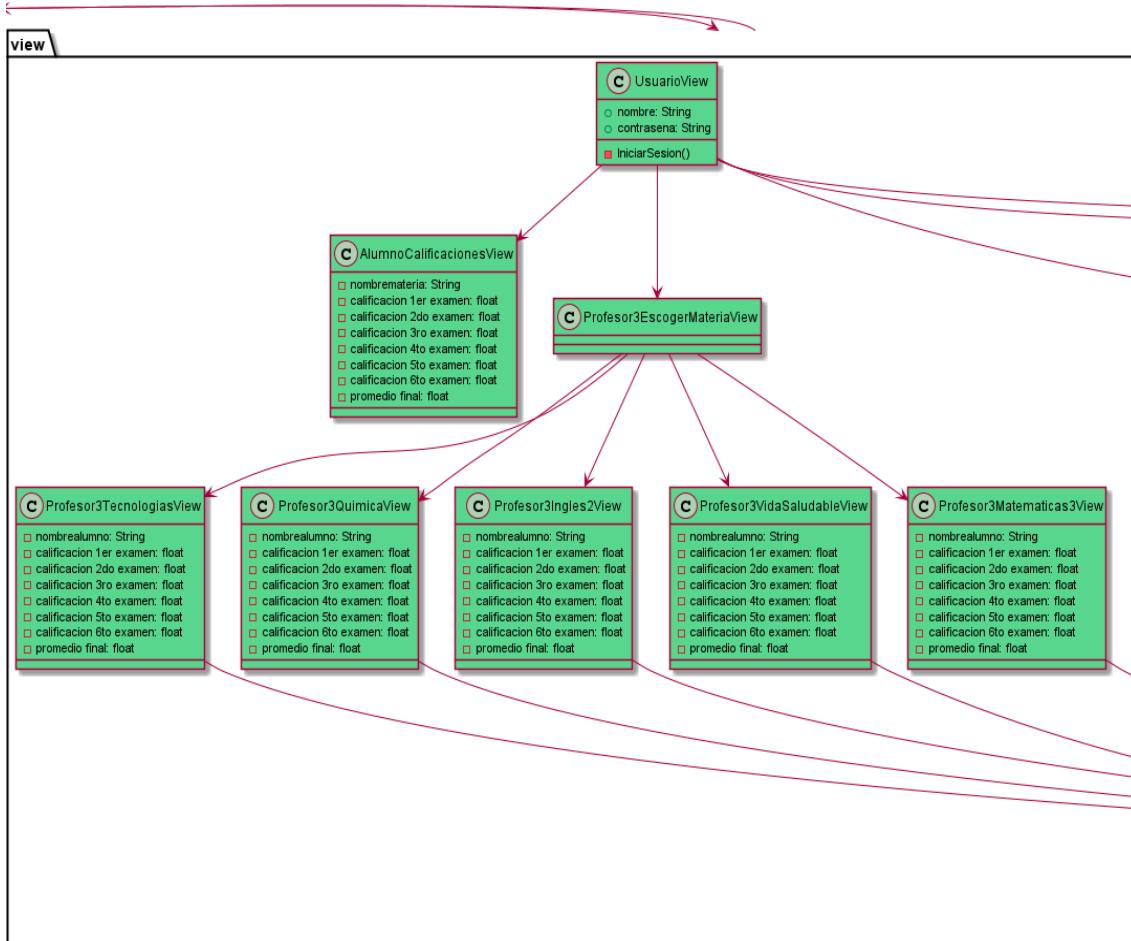


Figura 3.9: Diagramas Objetos parte 3

Este diagrama de objetos nos muestra una instancia de nuestros diagramas de clases y se enfoca en los atributos y sus relaciones que tienen los objetos en nuestro sistema y así como también nos describe la estructura estática de nuestro sistema.

```

interface model.ICalificaciones #elbee7 {
    +SacarPromedio ()
}

interface model.ISemestre #elbee7 {
    +EscogerSemestre ()
}

interface model.ISecretaria #elbee7 {
    +IniciarSesino ()
    -SubirAlumno ()
}

interface model.IMateria #elbee7{
    +ConsultarMateria ()
}

model.IMateria-->model.ISemestre
model.IMateria-->model.ICalificaciones
model.IUsuarioModel-->model.IAlumno
model.IUsuarioModel-->model.IProfesor
model.IUsuarioModel-->model.ISecretaria

class controller.UsuarioController    #ffff59d {
    +IniciarSesino ()
}

class controller.AlumnoController    #ffff59d {
    +IniciarSesino ()
}

class controller.SecretariaController  #ffff59d {
    +IniciarSesino ()
    -SubirAlumno ()
}

```

Figura 3.10: Diagramas Objetos código 3

```
class controller.ProfesorController #fff59d {
    +IniciarSesino()
    -SubirCalificacion()
}

class controller.MateriaController #ffff59d {
    +ConsultarMateria()
}

class controller.CalificacionesController #ffff59d {
    +SacarPronedio()
}

class controller.SemestreController #fff59d {
    +EscogerSemestre()
}

controller.UsuarioController <|-- controller.AlumnoController
controller.UsuarioController <|-- controller.ProfesorController
controller.UsuarioController <|-- controller.SecretariaController
controller.MateriaController <-- controller.CalificacionesController
controller.MateriaController <-- controller.SemestreController

class view.UsuarioView #58d68d{
    +nombre: String
    +contrasena: String
    -IniciarSesion()
}

class view.SecretariaEscogerGrupoView #58d68d{
}
```

Figura 3.11: Diagramas Objetos código 4

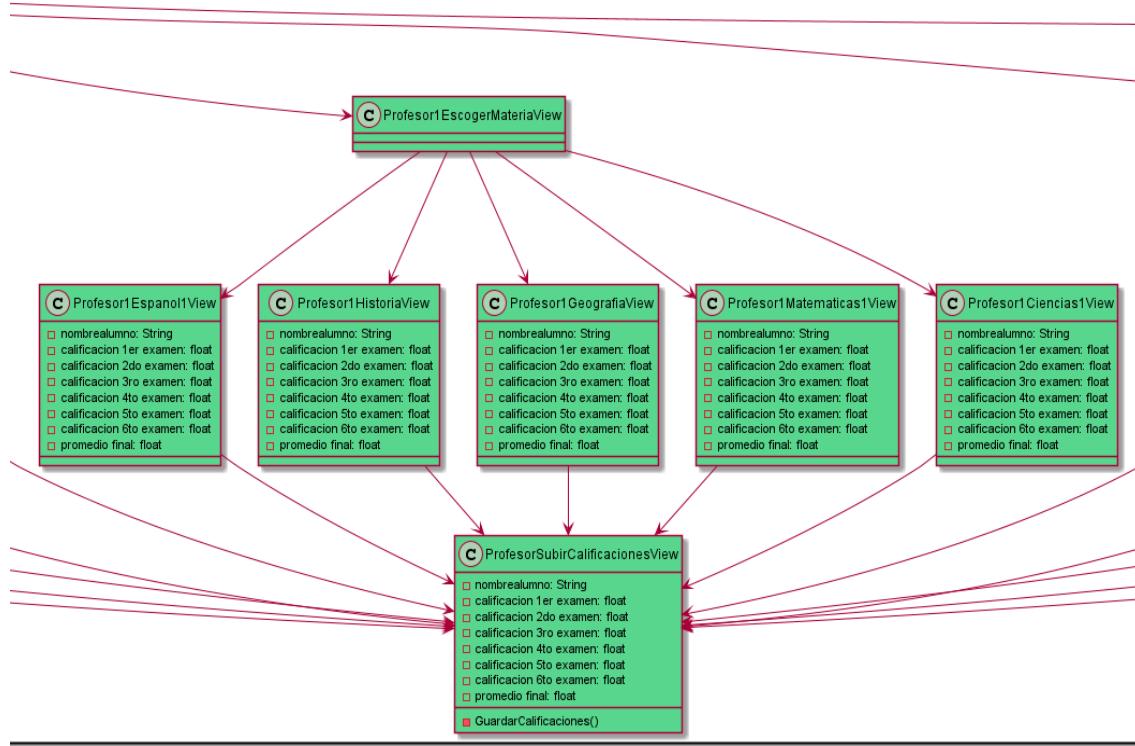


Figura 3.12: Diagramas Objetos parte 5

Este diagrama de objetos nos muestra una instancia de nuestros diagramas de clases y se enfoca en los atributos y sus relaciones que tienen los objetos en nuestro sistema y así como también nos describe la estructura estática de nuestro sistema.

```
class view.SecretariaModificarGrupalView #58d68d{
    -nombrealumno: String
    -grado: String
    -grupo: String
    -cicloescolar: String
}

class view.SecretariaModificarGrupo2View #58d68d{
    -nombrealumno: String
    -grado: String
    -grupo: String
    -cicloescolar: String
}

class view.SecretariaModificarGrupo3View #58d68d{
    -nombrealumno: String
    -grado: String
    -grupo: String
    -cicloescolar: String
}

class view.SecretariaAgregarAlumnoView #58d68d{
    -nombrealumno: String
    -grado: String
    -grupo: String
    -cicloescolar: String
    -GuardarAlumno ()
}
```

Figura 3.13: Diagramas Objetos código 5

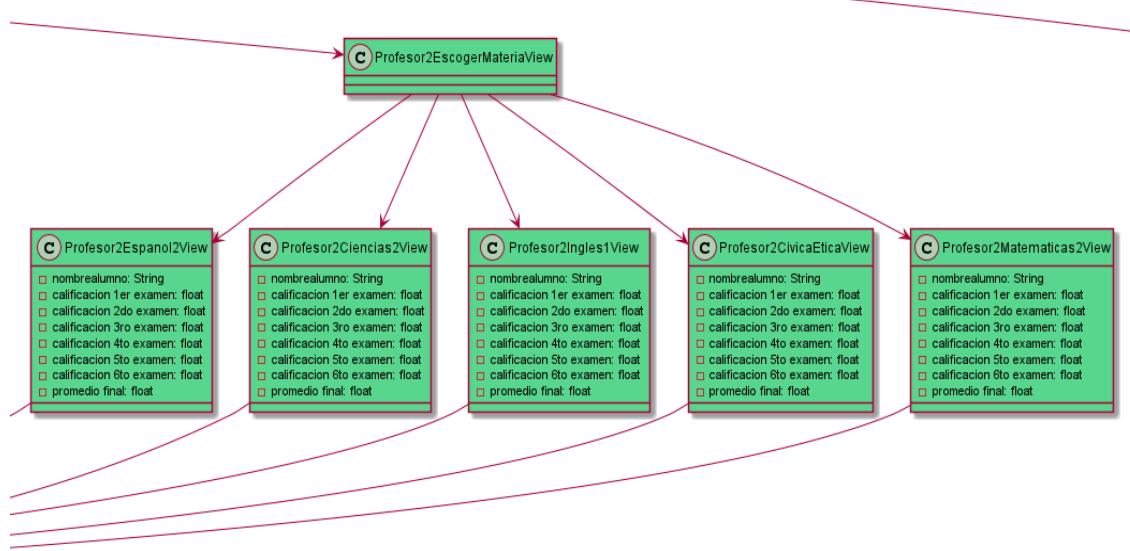


Figura 3.14: Diagramas Objetos parte 6

Este diagrama de objetos nos muestra una instancia de nuestros diagramas de clases y se enfoca en los atributos y sus relaciones que tienen los objetos en nuestro sistema y así como también nos describe la estructura estática de nuestro sistema.

```
class view.AlumnoCalificacionesView #58d68d{
    -nombremateria: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.ProfesorSubirCalificacionesView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
    -GuardarCalificaciones()
}

class view.Profesor1EscogerMateriaView #58d68d{
}
class view.Profesor2EscogerMateriaView #58d68d{
}
class view.Profesor3EscogerMateriaView #58d68d{
}
```

Figura 3.15: Diagramas Objetos código 6

```
class view.Profesor1Matematicas1View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor1Espanol1View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor1Ciencias1View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}
```

Figura 3.16: Diagramas Objetos código 7

```
class view.Profesor1HistoriaView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor1GeografiaView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor2Matematicas2View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}
```

Figura 3.17: Diagramas Objetos código 8

```
class view.Profesor2Espanol2View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor2Ciencias2View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor2Ingles1View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}
```

Figura 3.18: Diagramas Objetos código 9

```
class view.Profesor2CivicaEticaView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor3Matematicas3View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor3TecnologiasView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}
```

Figura 3.19: Diagramas Objetos código 10

```
class view.Profesor3QuimicaView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor3Ingles2View #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}

class view.Profesor3VidaSaludableView #58d68d{
    -nombrealumno: String
    -calificacion 1er examen: float
    -calificacion 2do examen: float
    -calificacion 3ro examen: float
    -calificacion 4to examen: float
    -calificacion 5to examen: float
    -calificacion 6to examen: float
    -promedio final: float
}
```

Figura 3.20: Diagramas Objetos código 11

```
view.UsuarioView-->view.Profesor1EscogerMateriaView  
view.UsuarioView-->view.Profesor2EscogerMateriaView  
view.UsuarioView-->view.Profesor3EscogerMateriaView  
view.UsuarioView-->view.AlumnoCalificacionesView  
view.UsuarioView-->view.SecretariaEscogerGrupoView  
view.SecretariaEscogerGrupoView-->view.SecretariaModificarGrupo1View  
view.SecretariaEscogerGrupoView-->view.SecretariaModificarGrupo2View  
view.SecretariaEscogerGrupoView-->view.SecretariaModificarGrupo3View  
view.SecretariaModificarGrupo1View-->view.SecretariaAgregarAlumnoView  
view.SecretariaModificarGrupo2View-->view.SecretariaAgregarAlumnoView  
view.SecretariaModificarGrupo3View-->view.SecretariaAgregarAlumnoView  
  
view.Profesor1EscogerMateriaView-->view.Profesor1Matematicas1View  
view.Profesor1EscogerMateriaView-->view.Profesor1Ciencias1View  
view.Profesor1EscogerMateriaView-->view.Profesor1HistoriaView  
view.Profesor1EscogerMateriaView-->view.Profesor1GeografiaView  
view.Profesor1EscogerMateriaView-->view.Profesor1Espanol1View  
view.Profesor1Espanol1View-->view.ProfesorSubirCalificacionesView  
view.Profesor1Matematicas1View-->view.ProfesorSubirCalificacionesView  
view.Profesor1Ciencias1View-->view.ProfesorSubirCalificacionesView  
view.Profesor1HistoriaView-->view.ProfesorSubirCalificacionesView  
view.Profesor1GeografiaView-->view.ProfesorSubirCalificacionesView
```

Figura 3.21: Diagramas Objetos código 12

```
view.Profesor2EscogerMateriaView-->view.Profesor2Matematicas2View
view.Profesor2EscogerMateriaView-->view.Profesor2Ciencias2View
view.Profesor2EscogerMateriaView-->view.Profesor2CivicaEticaView
view.Profesor2EscogerMateriaView-->view.Profesor2Ingles1View
view.Profesor2EscogerMateriaView-->view.Profesor2Espanol2View
view.Profesor2Espanol2View-->view.ProfesorSubirCalificacionesView
view.Profesor2Matematicas2View-->view.ProfesorSubirCalificacionesView
view.Profesor2Ciencias2View-->view.ProfesorSubirCalificacionesView
view.Profesor2Ingles1View-->view.ProfesorSubirCalificacionesView
view.Profesor2CivicaEticaView-->view.ProfesorSubirCalificacionesView

view.Profesor3EscogerMateriaView-->view.Profesor3Matematicas3View
view.Profesor3EscogerMateriaView-->view.Profesor3TecnologiasView
view.Profesor3EscogerMateriaView-->view.Profesor3VidaSaludableView
view.Profesor3EscogerMateriaView-->view.Profesor3Ingles2View
view.Profesor3EscogerMateriaView-->view.Profesor3QuimicaView
view.Profesor3TecnologiasView-->view.ProfesorSubirCalificacionesView
view.Profesor3Matematicas3View-->view.ProfesorSubirCalificacionesView
view.Profesor3QuimicaView-->view.ProfesorSubirCalificacionesView
view.Profesor3Ingles2View-->view.ProfesorSubirCalificacionesView
view.Profesor3VidaSaludableView-->view.ProfesorSubirCalificacionesView

controller --> model
controller <-- model
controller --> view
controller <-- view
```

Figura 3.22: Diagramas Objetos código 13

```
controller --> model
controller <-- model
controller --> view
controller <-- view
```

```
model --> entity
model <-- entity
```

```
BD --> model
BD <-- model
```

```
@enduml
```

Figura 3.23: Diagramas Objetos código 14

3.2.4. Diagrama de Componentes

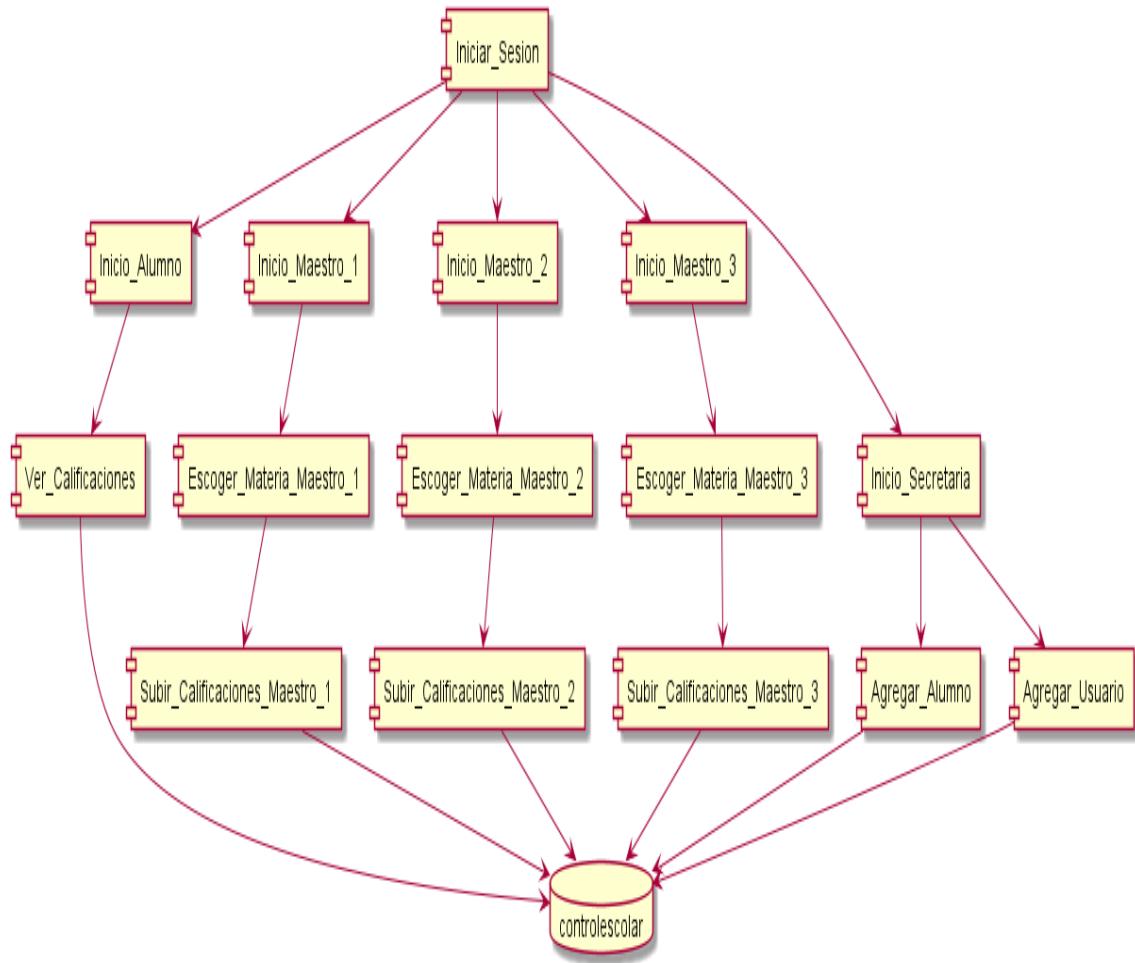


Figura 3.24: Diagrama de Componentes

Este diagrama nos proporciona una visión orientada a la ejecución de nuestro sistema ya que con ella se puede ver las organizaciones de nuestros componentes , también en este diagrama se puede observar las relaciones que tiene cada componente en nuestro sistema.

```
@startuml
component Iniciar_Sesion
component Inicio_Alumno
component Ver_Calificaciones
component Inicio_Maestro_1
component Escoger_Materia_Maestro_1
component Subir_Calificaciones_Maestro_1
component Inicio_Maestro_2
component Escoger_Materia_Maestro_2
component Subir_Calificaciones_Maestro_2
component Inicio_Maestro_3
component Escoger_Materia_Maestro_3
component Subir_Calificaciones_Maestro_3
component Inicio_Secretaria
component Agregar_Alumno
component Agregar_Usuario
database controlescolar
```

Figura 3.25: Diagrama de Componentes código

```
Iniciar_Sesion --> Inicio_Alumno
Inicio_Alumno --> Ver_Calificaciones
Ver_Calificaciones --> controlescolar
Iniciar_Sesion --> Inicio_Maestro_1
Inicio_Maestro_1 --> Escoger_Materia_Maestro_1
Escoger_Materia_Maestro_1 --> Subir_Calificaciones_Maestro_1
Subir_Calificaciones_Maestro_1 --> controlescolar
Iniciar_Sesion --> Inicio_Maestro_2
Inicio_Maestro_2 --> Escoger_Materia_Maestro_2
Escoger_Materia_Maestro_2 --> Subir_Calificaciones_Maestro_2
Subir_Calificaciones_Maestro_2 --> controlescolar
Iniciar_Sesion --> Inicio_Maestro_3
Inicio_Maestro_3 --> Escoger_Materia_Maestro_3
Escoger_Materia_Maestro_3 --> Subir_Calificaciones_Maestro_3
Subir_Calificaciones_Maestro_3 --> controlescolar
Iniciar_Sesion --> Inicio_Secretaria
Inicio_Secretaria --> Agregar_Alumno
Agregar_Alumno --> controlescolar
Inicio_Secretaria --> Agregar_Usuario
Agregar_Usuario --> controlescolar
@enduml
```

Figura 3.26: Diagrama de Componentes código

3.2.5. Diagrama de Despliegue

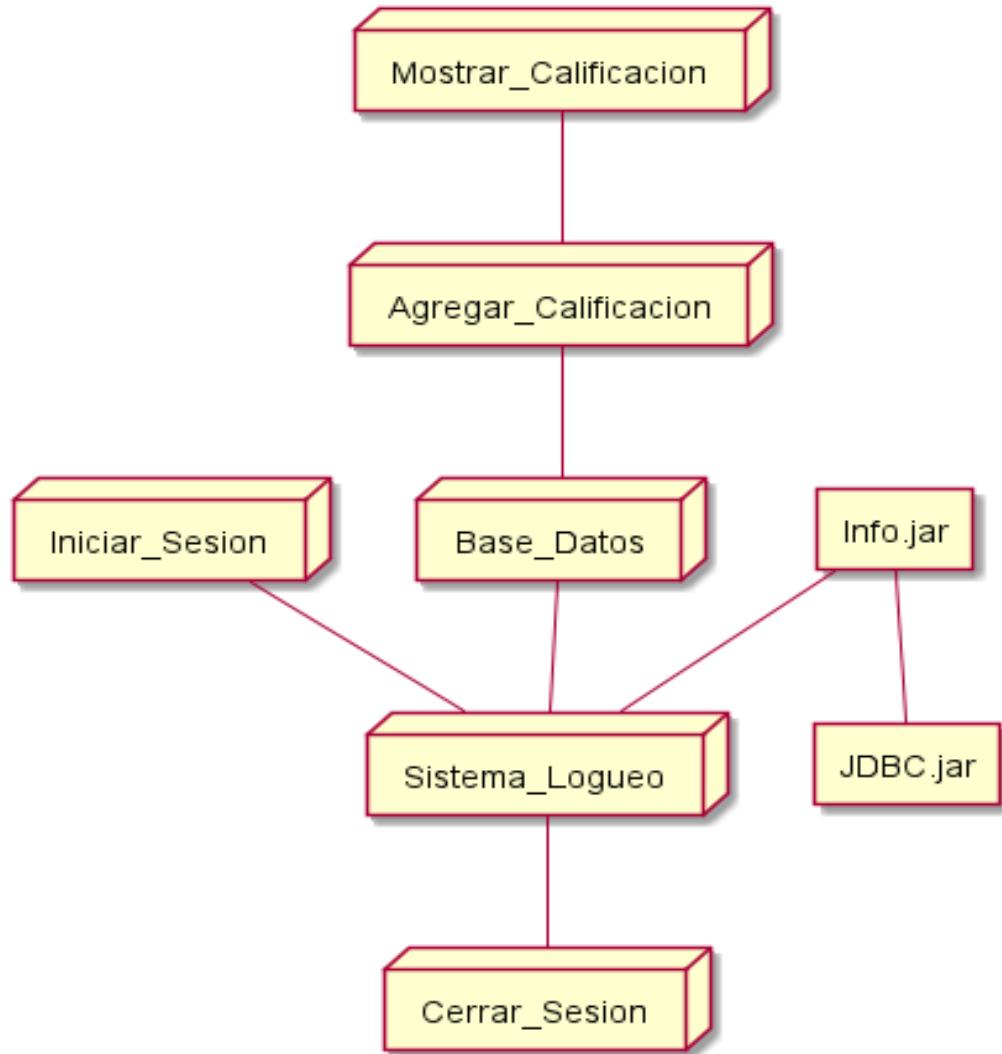


Figura 3.27: Diagrama de despliegue.

El Diagrama de despliegue que creamos nos muestran la arquitectura de ejecución de nuestro sistema y en este también se puede observar los enlaces de comunicación que tiene cada uno de los nodos de nuestro sistema ya que también nos ayudo a comprender como se desplegaría el sistema físicamente.

```
@startuml
node Iniciar_Sesion
node Base_Datos
node Agregar_Calificacion
node Mostrar_Calificacion
node Sistema_Logueo
node Cerrar_Sesion
rectangle JDBC.jar
rectangle Info.jar

Info.jar -- JDBC.jar
Info.jar -- Sistema_Logueo
Base_Datos -- Sistema_Logueo
Sistema_Logueo -- Cerrar_Sesion
Mostrar_Calificacion -- Agregar_Calificacion
Agregar_Calificacion -- Base_Datos
Iniciar_Sesion -- Sistema_Logueo
@enduml
```

Figura 3.28: Diagrama de Despliegue código

3.3. Diagramas de comportamiento

3.3.1. Diagramas de casos de uso

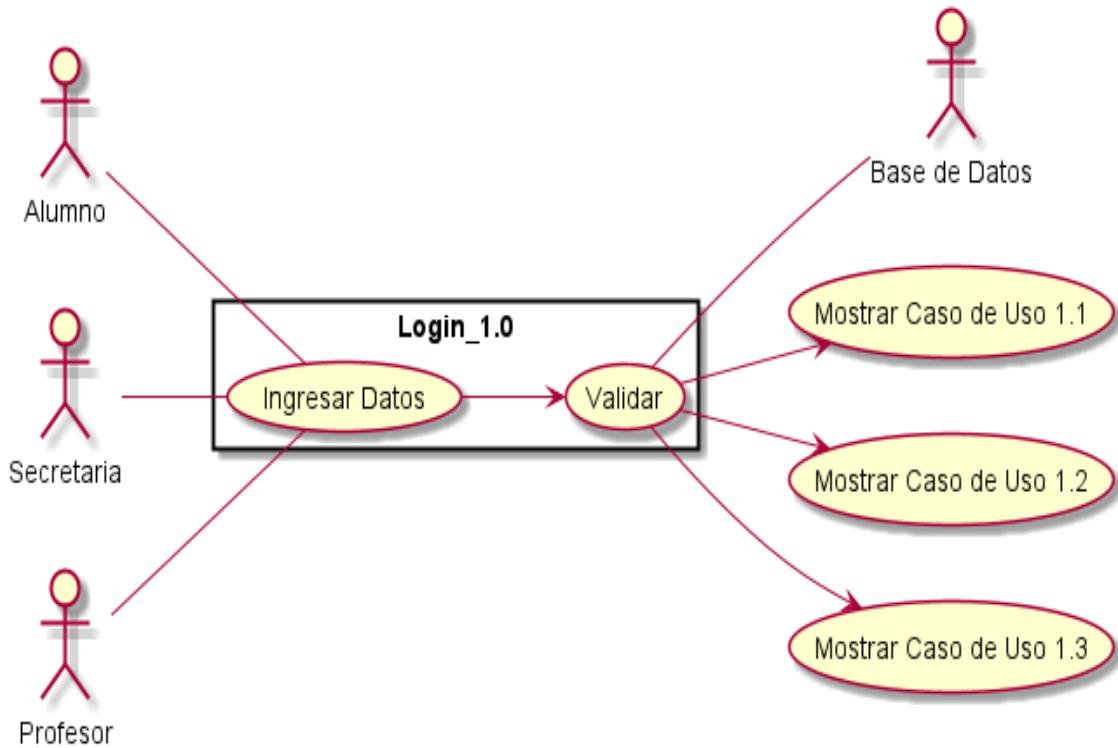


Figura 3.29: Diagrama de casos de uso de inicio

En el diagrama de caso de uso de iniciar sesión se muestra el comportamiento y la estructura de esta pantalla en la cual se puede observar que tenemos tres actores que es el maestro, el alumno y la secretaria.

ID	CU-1.0		
Nombre	Login/Inicio de sesión		
Creado	EJS	Actualizado:	GDCO
Fecha de creación	14/10/2021	Ultima versión:	01/11/2021
Actores	Usuario: Maestro, Alumno, secretaria, base de datos		
Descripción	Ingresar al programa por medio de un usuario y una contraseña		
Disparador	Ingresa los datos correspondientes y le da aceptar		
Precondiciones	N/A		
Postcondiciones	Si no son válidos los datos se vuelve a mostrar la misma pantalla		
Flujo normal	<p>1. Se ingresa el usuario y la contraseña</p> <p>1.1. Se da click al botón aceptar y si los datos son incorrectos se muestra un mensaje de ERROR.</p> <p>2. Si los datos son correctos se muestra la pantalla correspondiente del usuario.</p>		
Flujos alternos	Al terminar de ingresar se muestra la pantalla 1.1, 1.2, 1.3, 1.4 o 1.5		
Includes			
Frecuencia de uso	Frecuente		
Requerimientos especiales	N/A		
Supuestos			
Issues y notas			

Figura 3.30: Tabla del caso de uso de Login.

```

@startuml
left to right direction
actor "Alumno" as al
actor "Base de Datos" as bd
actor "Secretaria" as sc
actor "Profesor" as pf
rectangle Login_1.0 {
    usecase "Ingresar Datos" as UC1
    usecase "Validar" as UC2
}
usecase "Mostrar Caso de Uso 1.1" as UC3
usecase "Mostrar Caso de Uso 1.2" as UC4
usecase "Mostrar Caso de Uso 1.3" as UC5
al -- UC1
sc -- UC1
pf -- UC1
UC2 --> UC3
UC2 --> UC4
UC2 --> UC5
UC1 --> UC2
UC2 --> bd
@enduml

```

Figura 3.31: Código de Login

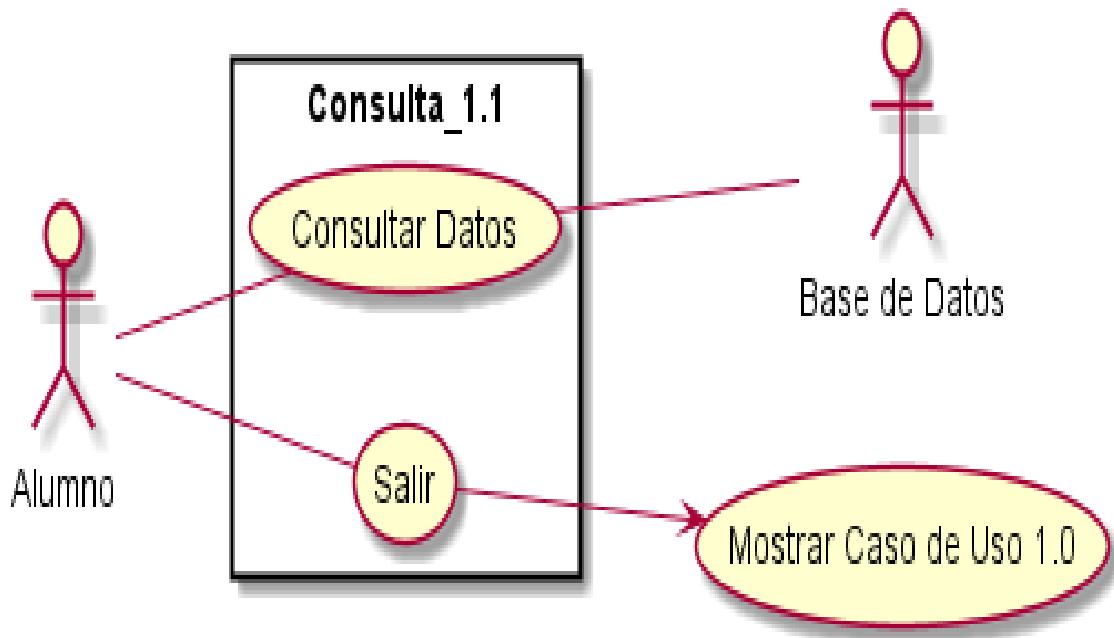


Figura 3.32: Diagrama de casos de uso de consulta

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.1		
Nombre	Consulta		
Creado	EJS	Actualizado:	GDCO
Fecha de creación	14/10/2021	Ultima versión:	01/11/2021
Actores	Alumno, base de datos		
Descripción	Se muestra la pantalla con las calificaciones de las materias del alumno		
Disparador	Se manda esta pantalla al terminar de ingresar en la pantalla 1.0 (Login)		
Precondiciones	N/A		
<u>Postcondiciones</u>	N/A		
Flujo normal	1. La función de esta pantalla solo será mostrar los datos en pantalla		
Flujos alternos	Al terminar se cierra el programa		
<u>Includes</u>			
Frecuencia de uso	Frecuente		
Requerimientos especiales	N/A		
Supuestos			
Issues y notas			

Figura 3.33: Tabla del caso de uso de Consulta

```
@startuml
left to right direction
actor "Alumno" as al
actor "Base de Datos" as bd
rectangle Consulta_1.1 {
    usecase "Consultar Datos" as UC1
    usecase "Salir" as UC2
}
usecase "Mostrar Caso de Uso 1.0" as UC3
al -- UC1
al -- UC2
UC2 --> UC3
UC1 -- bd
@enduml
```

Figura 3.34: Código de Consulta

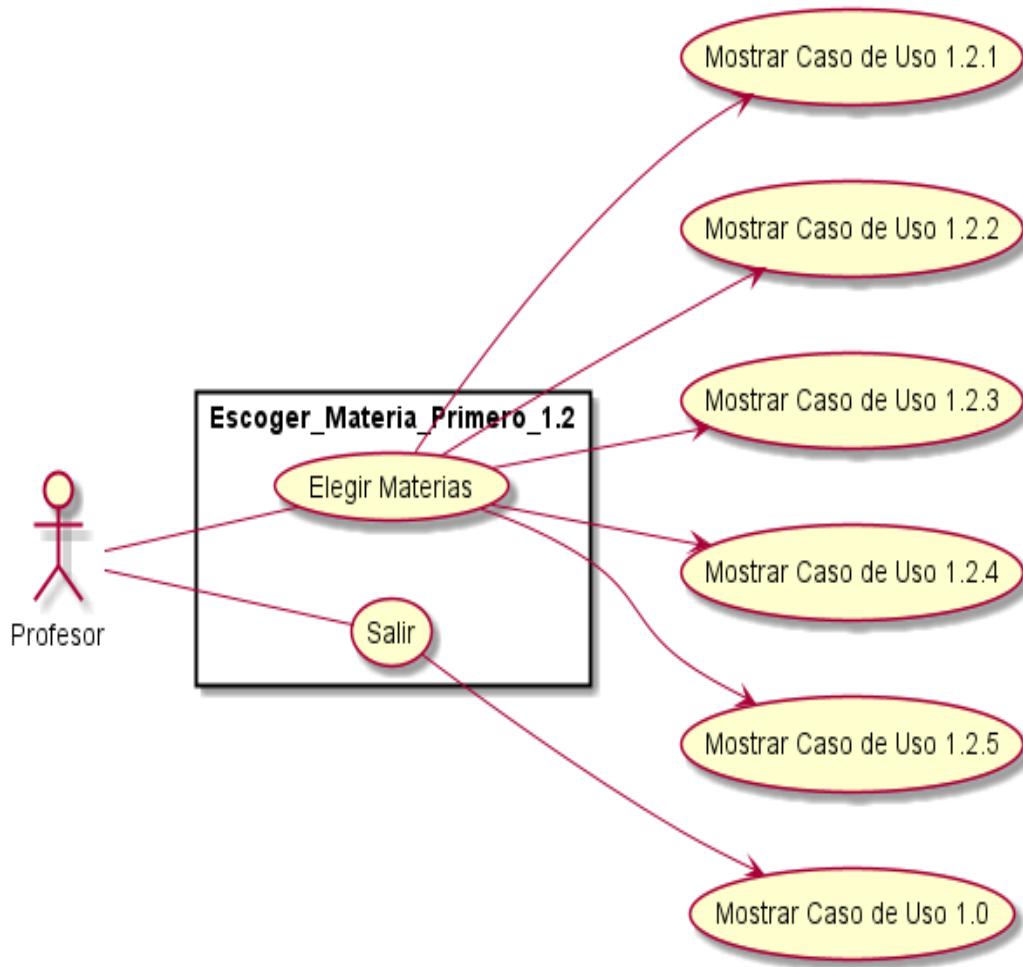


Figura 3.35: En la figura se muestra el caso de uso de Escoger Materia Primero
 Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2	
Nombre	Escoger Materia Primero	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Última versión: 01/11/2021
Actores	Maestro de primer año	
Descripción	Muestra la pantalla con todas las materias que tiene asignadas el maestro	
Disparador	Se manda esta pantalla al terminar de ingresar en la pantalla 1.0 (Login)	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se le da <u>click</u> al botón de una de las 5 materias asignadas y se envía a la pantalla 1.2.1 - 1.2.2 - 1.2.3 - 1.2.4 - 1.2.5 2. Se le da <u>click</u> al botón de salir y regresa a la pantalla de inicio 1.0	
Flujos alternos	Al terminar se regresa a la pantalla 1.0	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.36: Tabla del caso de uso de Escoger Materia Primero.

```

@startuml
left to right direction
actor "Profesor" as pf
rectangle Escoger_Materia_Primero_1.2 {
    usecase "Elegir Materias" as UC1
    usecase "Salir" as UC2
}
usecase "Mostrar Caso de Uso 1.0" as UC3
usecase "Mostrar Caso de Uso 1.2.1" as UC4
usecase "Mostrar Caso de Uso 1.2.2" as UC5
usecase "Mostrar Caso de Uso 1.2.3" as UC6
usecase "Mostrar Caso de Uso 1.2.4" as UC7
usecase "Mostrar Caso de Uso 1.2.5" as UC8
pf -- UC1
pf -- UC2
UC1 --> UC4
UC1 --> UC5
UC1 --> UC6
UC1 --> UC7
UC1 --> UC8
UC2 --> UC3
@enduml

```

Figura 3.37: Código de Escoger Materia Primero

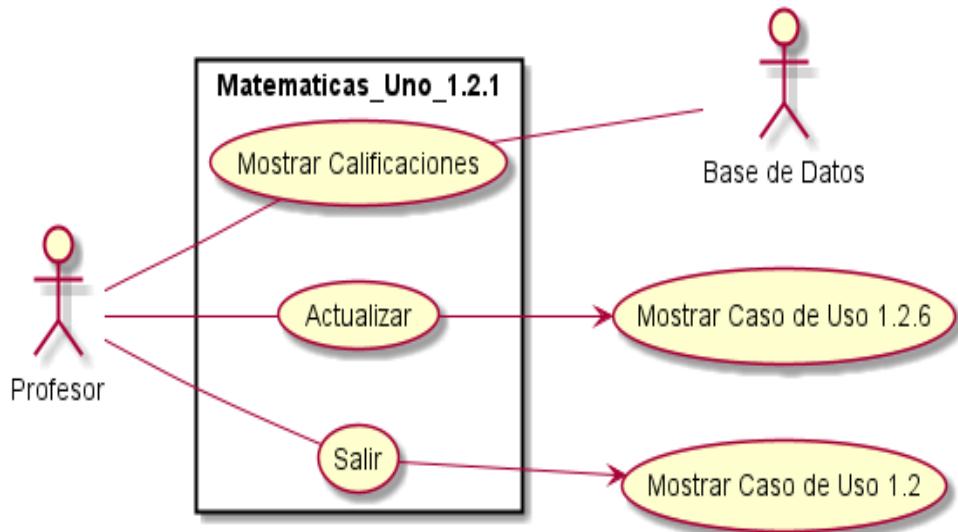


Figura 3.38: En la figura se muestra el caso de uso de Matematicas Uno.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2.1	
Nombre	Matemáticas uno	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Última versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de matemáticas 1	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón matemáticas 1 de la pantalla 1.2	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.2.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.2	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.39: Tabla del caso de uso de Matematicas Uno

```
@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Matematicas_Uno_1.2.1 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.2" as UC4
usecase "Mostrar Caso de Uso 1.2.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC2 --> UC5
UC3 --> UC4
UC1 --> bd
@enduml
```

Figura 3.40: Código de Matematicas Uno

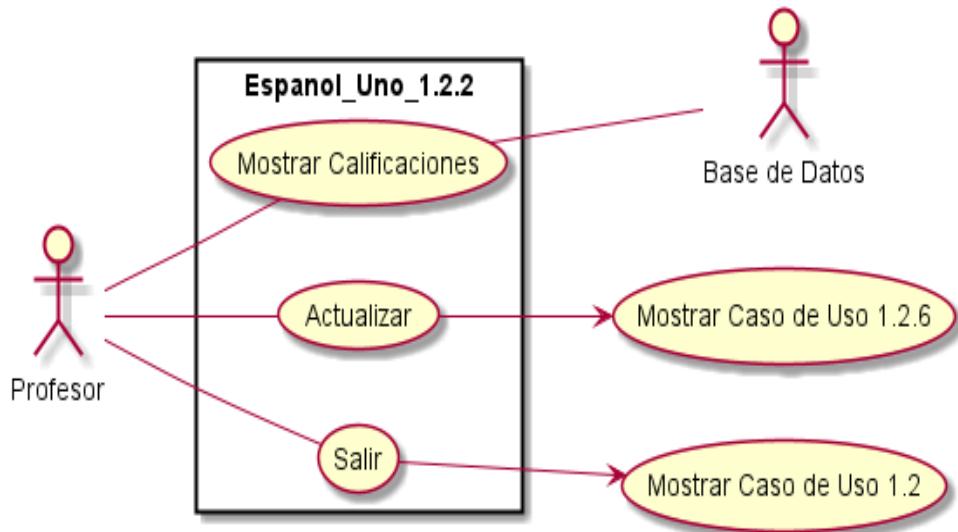


Figura 3.41: En la figura se muestra el caso de uso de Español Uno

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2.2	
Nombre	Español uno	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de español 1	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón español 1 de la pantalla 1.2	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.2.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.2	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.42: Tabla del caso de uso de Español Uno.

```
@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Espanol_Uno_1.2.2 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.2" as UC4
usecase "Mostrar Caso de Uso 1.2.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC2 --> UC5
UC3 --> UC4
UC1 --> bd
@enduml
```

Figura 3.43: Código de Español Uno

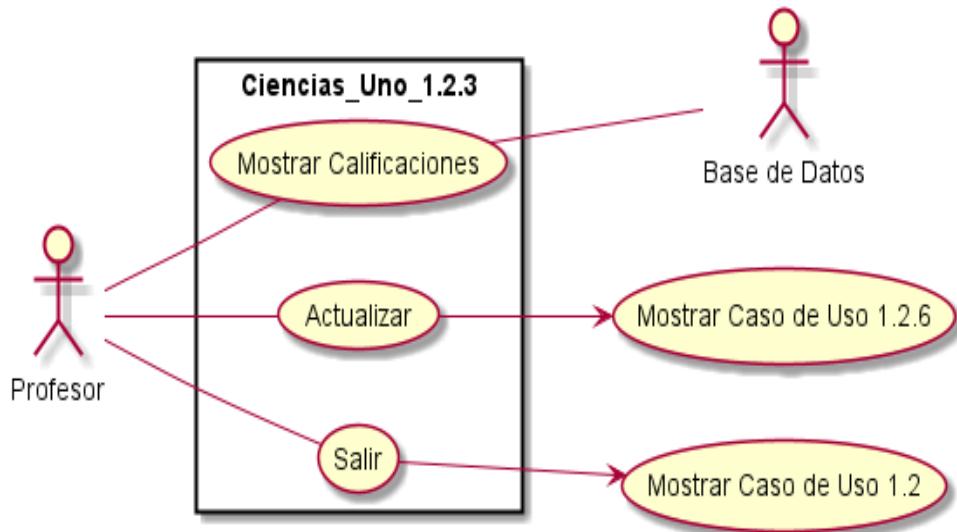


Figura 3.44: En la figura se muestra el caso de uso de Ciencias Uno

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2.3	
Nombre	Ciencias uno	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de ciencias 1	
Disparador	Se manda esta pantalla al dar click en el botón ciencias 1 de la pantalla 1.2	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da click al botón actualizar. 2. Al dar click en actualizar se le mandará a la pantalla actualizar 1.2.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.2	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.45: Tabla del caso de uso de Ciencias Uno.

```
@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Ciencias_Undo_1.2.3 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.2" as UC4
usecase "Mostrar Caso de Uso 1.2.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC2 --> UC5
UC3 --> UC4
UC1 --> bd
@enduml
```

Figura 3.46: Código de Ciencias Uno

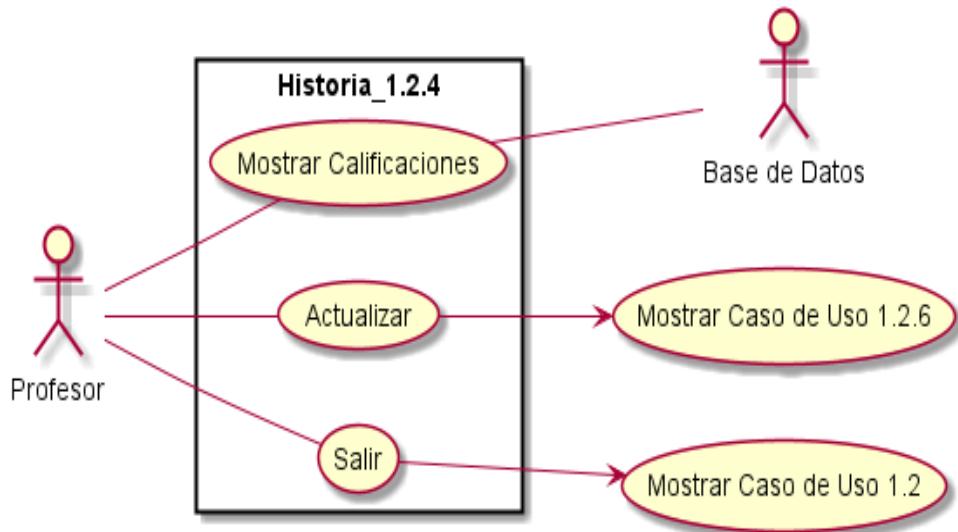


Figura 3.47: En la figura se muestra el caso de uso de Historia

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2.4	
Nombre	Historia	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de historia	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón historia de la pantalla 1.2	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.2.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.2	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.48: Tabla del caso de uso de Historia

```
@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Historia_1.2.4 {
    usecase "Modificar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.2" as UC4
usecase "Mostrar Caso de Uso 1.2.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC2 --> UC5
UC3 --> UC4
UC1 --> bd
@enduml
```

Figura 3.49: Código de Historia

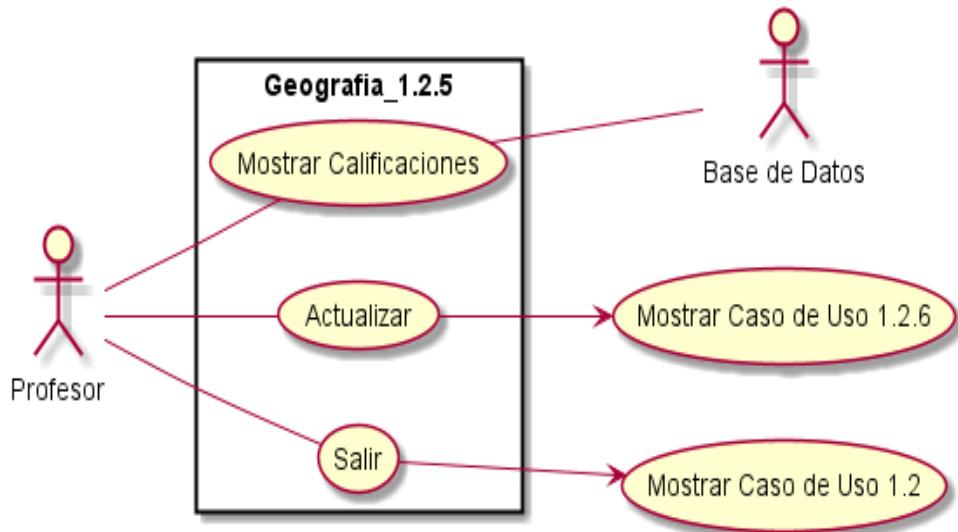


Figura 3.50: En la figura se muestra el caso de uso de Geografía

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2.5	
Nombre	Geografía	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de geografía	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón geografía de la pantalla 1.2	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.2.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.2	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.51: Tabla del caso de uso de Geografía.

```
@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Geografia_1.2.5 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.2" as UC4
usecase "Mostrar Caso de Uso 1.2.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC2 --> UC5
UC3 --> UC4
UC1 --> bd
@enduml
```

Figura 3.52: Código de Actualizacion Geografía

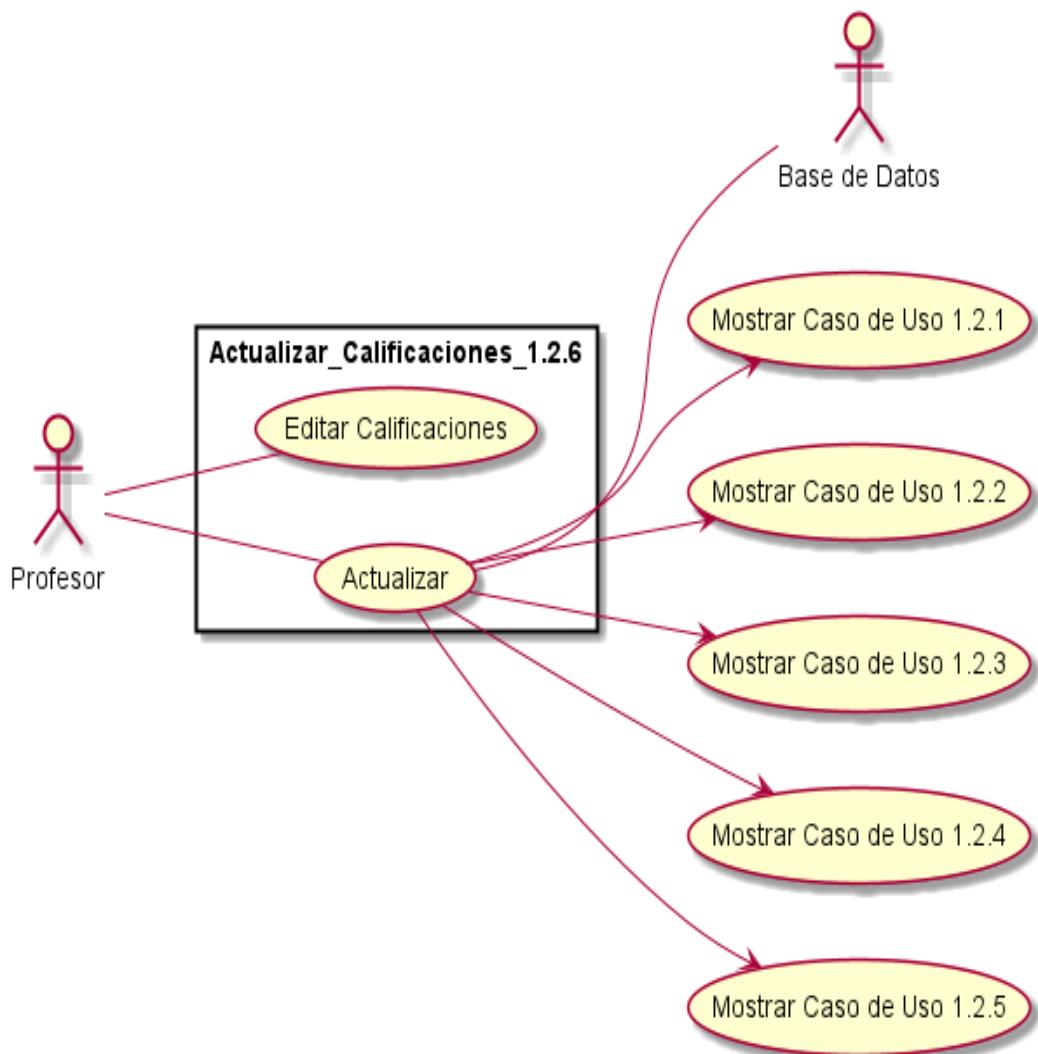


Figura 3.53: En la figura se muestra el caso de uso de Actualizacion Calificaciones.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.2.6	
Nombre	Actualización Calificación	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con los datos del alumno de la materia de geografía	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón actualizar de la pantalla 1.2.1 - 1.2.2 - 1.2.3 - 1.2.4 - 1.2.5	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se modifica el campo de calificaciones, solo ese campo 2. Al dar <u>click</u> en el botón guardar se hacen los cambios y se guardan en la base de datos.	
Flujos alternos	Al terminar se regresa a la pantalla 1.2.1 - 1.2.2 - 1.2.3 - 1.2.4 - 1.2.5	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.54: Tabla del caso de uso de Actualizacion Calificaciones.

```

@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Actualizar_Calificaciones_1.2.6 {
    usecase "Editar Calificaciones" as UC1
    usecase "Actualizar" as UC2
}
usecase "Mostrar Caso de Uso 1.2.1" as UC3
usecase "Mostrar Caso de Uso 1.2.2" as UC4
usecase "Mostrar Caso de Uso 1.2.3" as UC5
usecase "Mostrar Caso de Uso 1.2.4" as UC6
usecase "Mostrar Caso de Uso 1.2.5" as UC7
pf -- UC1
pf -- UC2
UC2 --> UC3
UC2 --> UC4
UC2 --> UC5
UC2 --> UC6
UC2 --> UC7
UC2 --> bd
@enduml

```

Figura 3.55: Código de Actualizacion Calificaciones

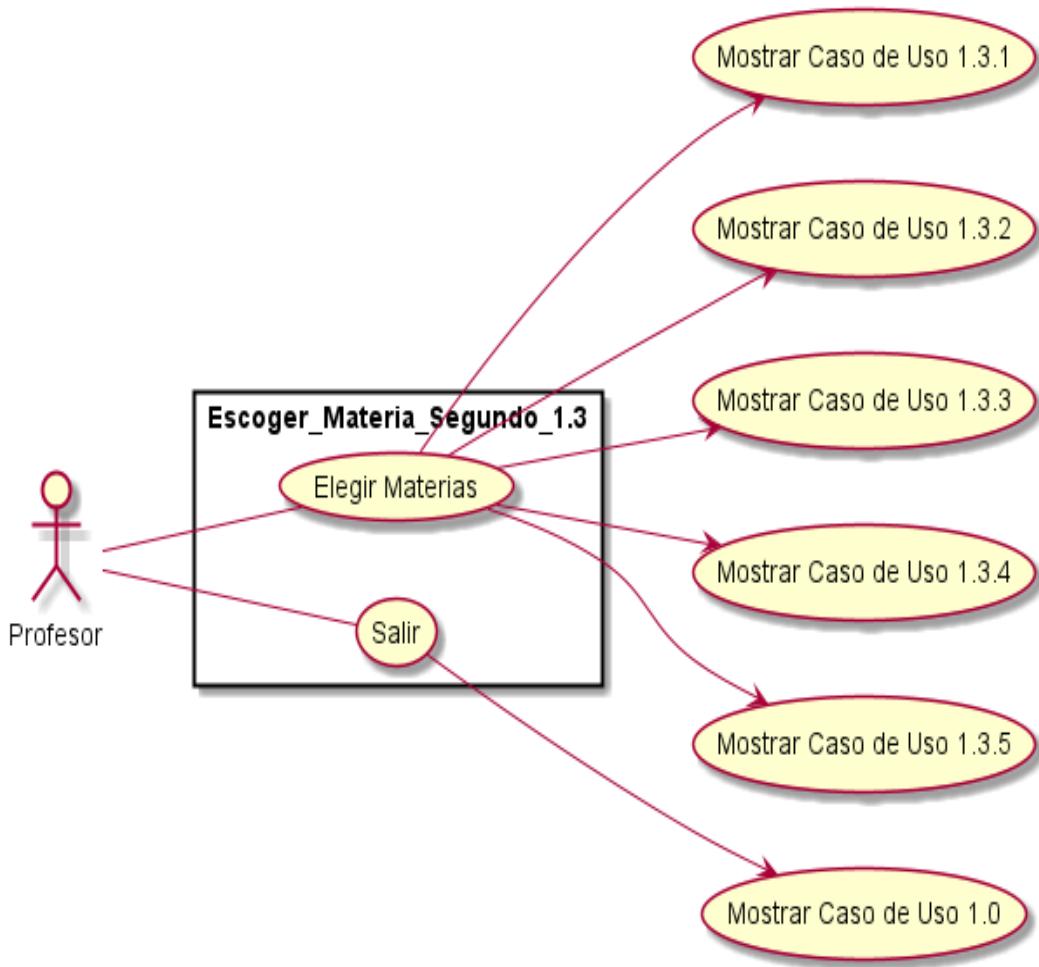


Figura 3.56: En la figura se muestra el caso de uso de Escoger Materia Segundo
 Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El
 diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar
 alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades
 que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.3	
Nombre	Escoger Materias Segundo	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de segundo año, base de datos	
Descripción	Muestra la pantalla con todas las materias que tiene asignadas el maestro	
Disparador	Se manda esta pantalla al terminar de ingresar en la pantalla 1.0 (Login)	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se le da click al botón de una de las 5 materias asignadas y se envía a la pantalla 1.3.1 - 1.3.2 - 1.3.3 - 1.3.4 - 1.3.5 2. Se le da click al botón de salir y regresa a la pantalla de inicio 1.0	
Flujos alternos	Al terminar se regresa a la pantalla 1.0	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.57: Tabla del caso de uso de Escoger Materia Segundo

```

@startuml
left to right direction
actor "Profesor" as pf
rectangle Escoger_Materia_Segundo_1.3 {
    usecase "Elegir Materias" as UC1
    usecase "Salir" as UC2
}
usecase "Mostrar Caso de Uso 1.0" as UC3
usecase "Mostrar Caso de Uso 1.3.1" as UC4
usecase "Mostrar Caso de Uso 1.3.2" as UC5
usecase "Mostrar Caso de Uso 1.3.3" as UC6
usecase "Mostrar Caso de Uso 1.3.4" as UC7
usecase "Mostrar Caso de Uso 1.3.5" as UC8
pf -- UC1
pf -- UC2
UC1 --> UC4
UC1 --> UC5
UC1 --> UC6
UC1 --> UC7
UC1 --> UC8
UC2 --> UC3
@enduml

```

Figura 3.58: Código de Escoger Materia Segundo

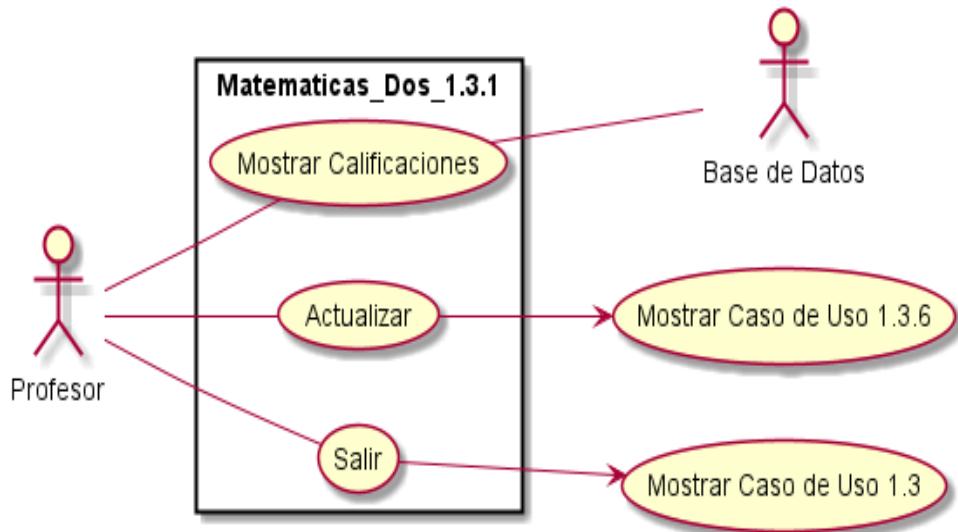


Figura 3.59: En la figura se muestra el caso de uso de Matematicas Dos.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.3.1	
Nombre	Matemáticas 2	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de segundo año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de matemáticas 2	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón matemáticas 2 de la pantalla 1.3	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.3.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.3	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.60: Tabla del caso de uso de Matematicas Dos.

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Matematicas_Dos_1.3.1 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.3" as UC4
usecase "Mostrar Caso de Uso 1.3.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.61: Código de Matematicas Dos

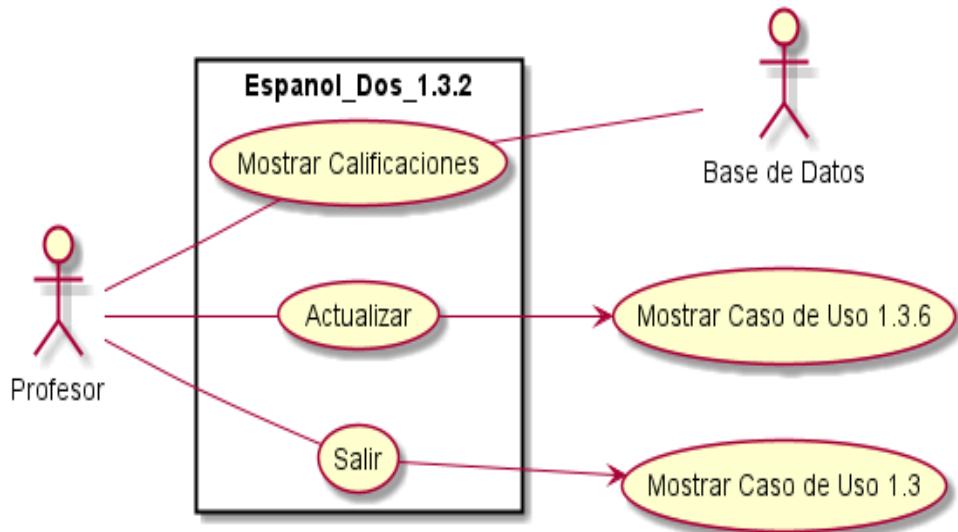


Figura 3.62: En la figura se muestra el caso de uso de Español Dos

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.3.2	
Nombre	Español 2	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de segundo año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de español 2	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón español 2 de la pantalla 1.3	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.3.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.3	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.63: Tabla del caso de uso de Español Dos.

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Espanol_Dos_1.3.2 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.3" as UC4
usecase "Mostrar Caso de Uso 1.3.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.64: Código de Español Dos

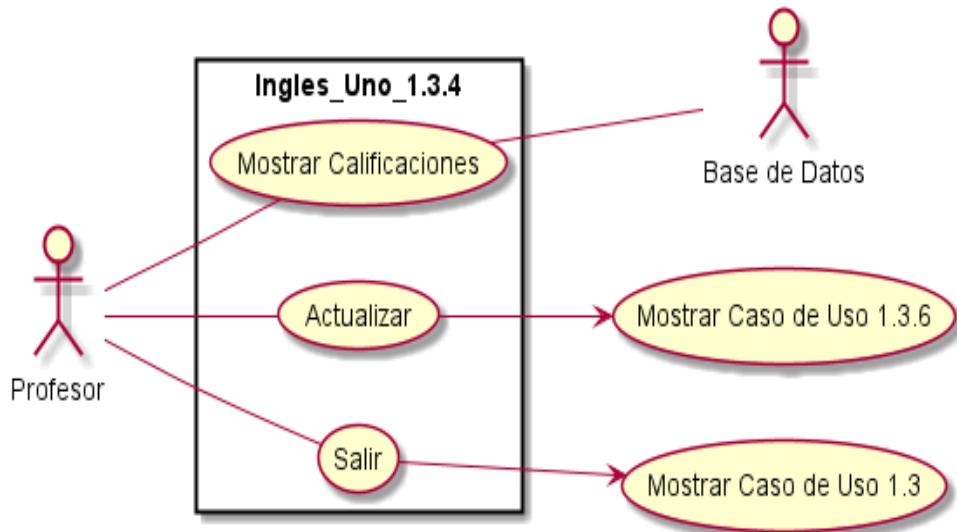


Figura 3.65: En la figura se muestra el caso de uso de Ingles Uno.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.3.4	
Nombre	Ingles uno	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de segundo año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de ingles 1	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón ingles 1 de la pantalla 1.3	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.3.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.3	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.66: Tabla del caso de uso de Ingles Uno.

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Ingles_Uno_1.3.4 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.3" as UC4
usecase "Mostrar Caso de Uso 1.3.6" as UC5
pf --- UC1
pf --- UC2
pf --- UC3
UC1 --- bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.67: Código de Ingles Uno

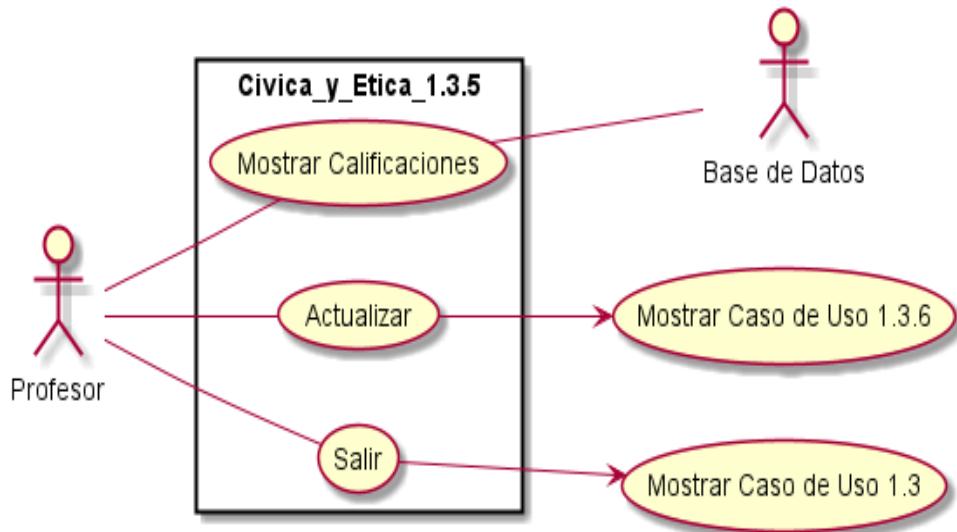


Figura 3.68: En la figura se muestra el caso de uso de Cívica y Ética.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.3.5		
Nombre	Cívica y ética		
Creado	EJS	Actualizado:	GDCO
Fecha de creación	14/10/2021	Ultima versión:	01/11/2021
Actores	Maestro de segundo año, base de datos		
Descripción	Muestra la pantalla con las calificaciones de la materia de cívica y ética		
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón geografía de la pantalla 1.3		
Precondiciones	N/A		
<u>Postcondiciones</u>	N/A		
Flujo normal	<p>1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar.</p> <p>2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.3.6 que contendrá los datos del alumno y los cuales se actualizarán.</p>		
Flujos alternos	Al terminar se regresa a la pantalla 1.3		
<u>Includes</u>			
Frecuencia de uso	Muy Frecuente		
Requerimientos especiales	N/A		
Supuestos			
Issues y notas			

Figura 3.69: Tabla del caso de uso de Cívica y Ética.

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Civica_y_Etica_1.3.5 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.3" as UC4
usecase "Mostrar Caso de Uso 1.3.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.70: Código de Cívica y Ética

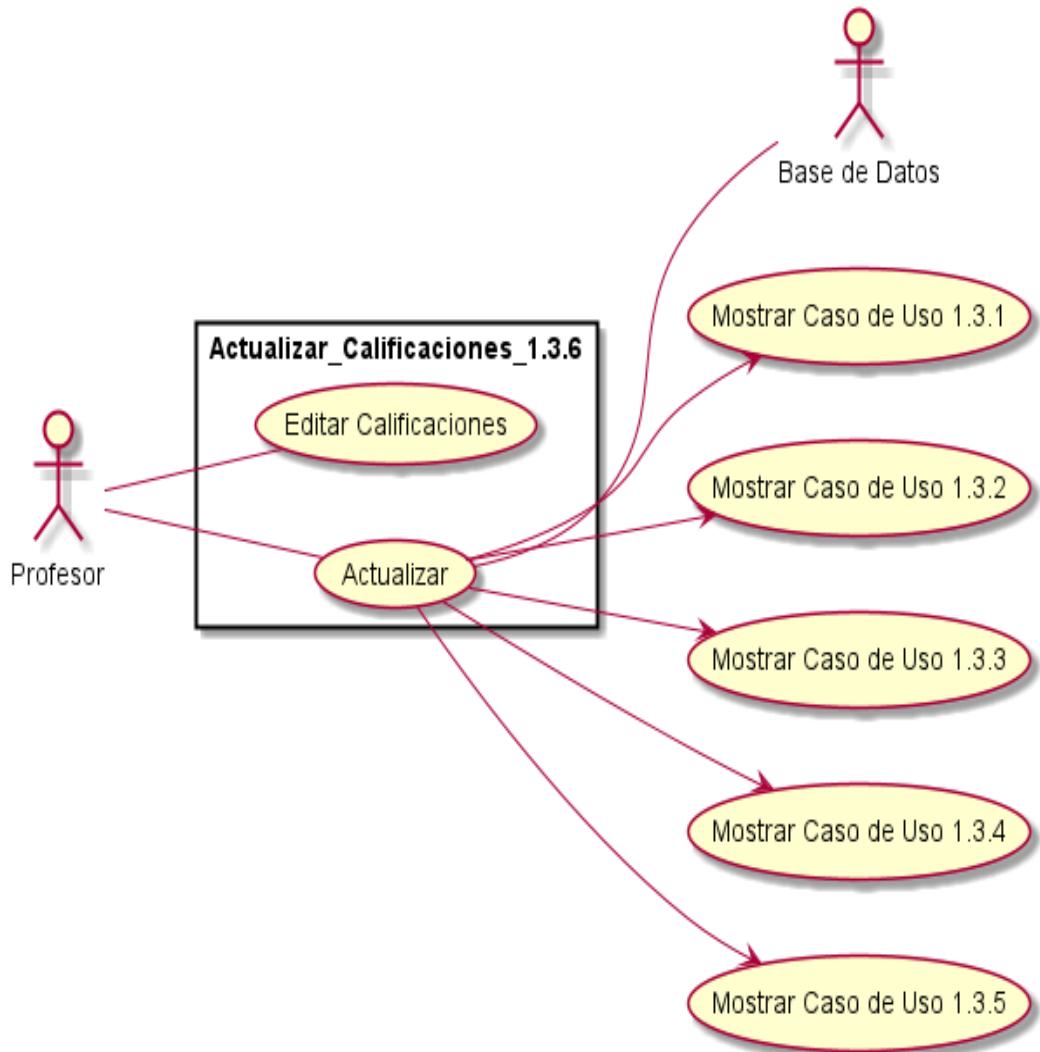


Figura 3.71: En la figura se muestra el caso de uso de Actualizacion Calificaciones.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.3.6	
Nombre	Actualización Calificación	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de primer año, base de datos	
Descripción	Muestra la pantalla con los datos del alumno de la materia de cívica y ética	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón actualizar de la pantalla 1.3.1 - 1.3.2 - 1.3.3 - 1.3.4 - 1.3.5	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se modifica el campo de calificaciones, solo ese campo 2. Al dar <u>click</u> en el botón guardar se hacen los cambios y se guardan en la base de datos.	
Flujos alternos	Al terminar se regresa a la pantalla 1.3.1 - 1.3.2 - 1.3.3 - 1.3.4 - 1.3.5	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.72: Tabla del caso de uso de Actualizacion Calificaciones.

```
@startuml
left to right direction
actor "Base de Datos" as bd
actor "Profesor" as pf
rectangle Actualizar_Calificaciones_1.3.6 {
    usecase "Editar Calificaciones" as UC1
    usecase "Actualizar" as UC2
}
usecase "Mostrar Caso de Uso 1.3.1" as UC3
usecase "Mostrar Caso de Uso 1.3.2" as UC4
usecase "Mostrar Caso de Uso 1.3.3" as UC5
usecase "Mostrar Caso de Uso 1.3.4" as UC6
usecase "Mostrar Caso de Uso 1.3.5" as UC7
pf -- UC1
pf -- UC2
UC2 --> UC3
UC2 --> UC4
UC2 --> UC5
UC2 --> UC6
UC2 --> UC7
UC2 --> bd
@enduml
```

Figura 3.73: Código de Actualizacion Calificaciones

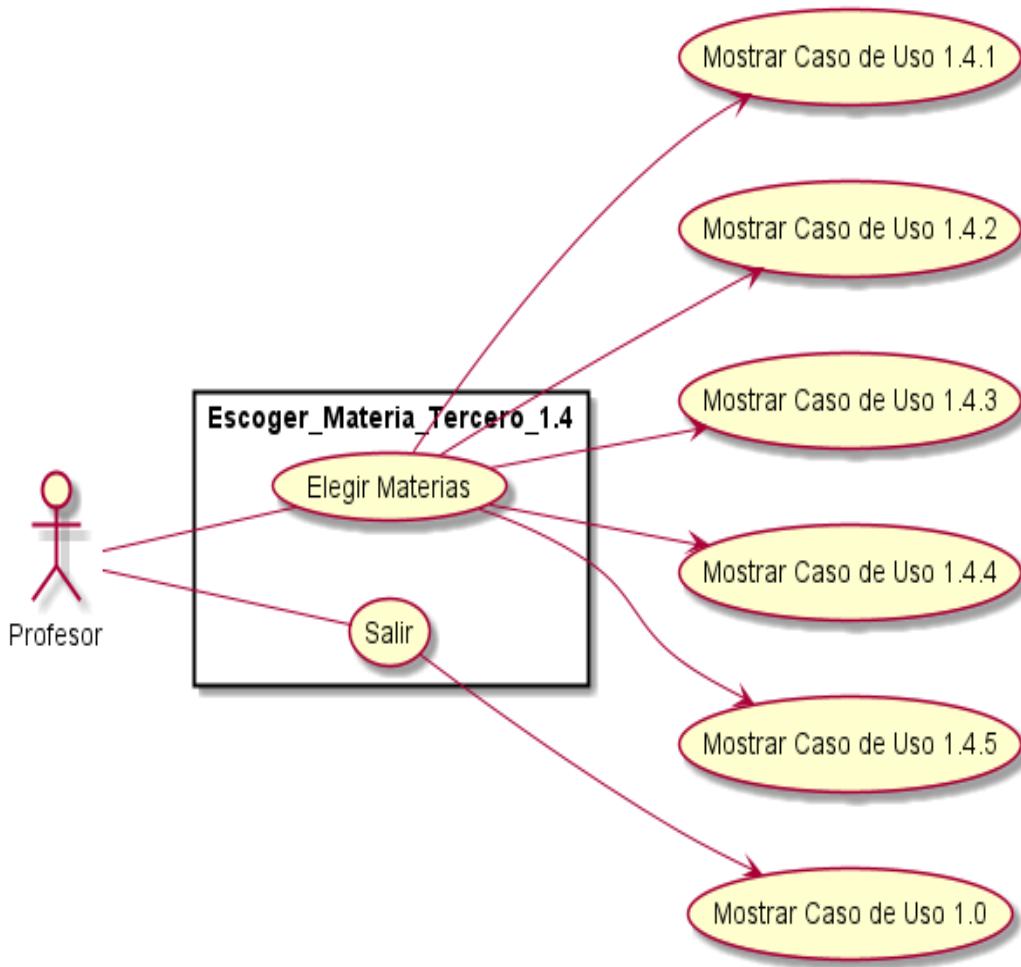


Figura 3.74: En la figura se muestra el caso de uso de Escoger Materia Tercero.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.4	
Nombre	Escoger Materias	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de tercer año	
Descripción	Muestra la pantalla con todas las materias que tiene asignadas el maestro	
Disparador	Se manda esta pantalla al terminar de ingresar en la pantalla 1.0 (Login)	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se le da click al botón de una de las 5 materias asignadas y se envía a la pantalla 1.4.1 - 1.4.2 - 1.4.3 - 1.4.4 - 1.4.5 2. Se le da click al botón de salir y regresa a la pantalla de inicio 1.0	
Flujos alternos	Al terminar se regresa a la pantalla se regresa a la pantalla 1.0	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.75: Tabla del caso de uso de Escoger Materia Tercero.

```

@startuml
left to right direction
actor "Profesor" as pf
rectangle Escoger_Materia_Tercero_1.4 {
    usecase "Elegir Materias" as UC1
    usecase "Salir" as UC2
}
usecase "Mostrar Caso de Uso 1.0" as UC3
usecase "Mostrar Caso de Uso 1.4.1" as UC4
usecase "Mostrar Caso de Uso 1.4.2" as UC5
usecase "Mostrar Caso de Uso 1.4.3" as UC6
usecase "Mostrar Caso de Uso 1.4.4" as UC7
usecase "Mostrar Caso de Uso 1.4.5" as UC8
pf -- UC1
pf -- UC2
UC1 --> UC4
UC1 --> UC5
UC1 --> UC6
UC1 --> UC7
UC1 --> UC8
UC2 --> UC3
@enduml

```

Figura 3.76: Código de Escoger Materia Tercero

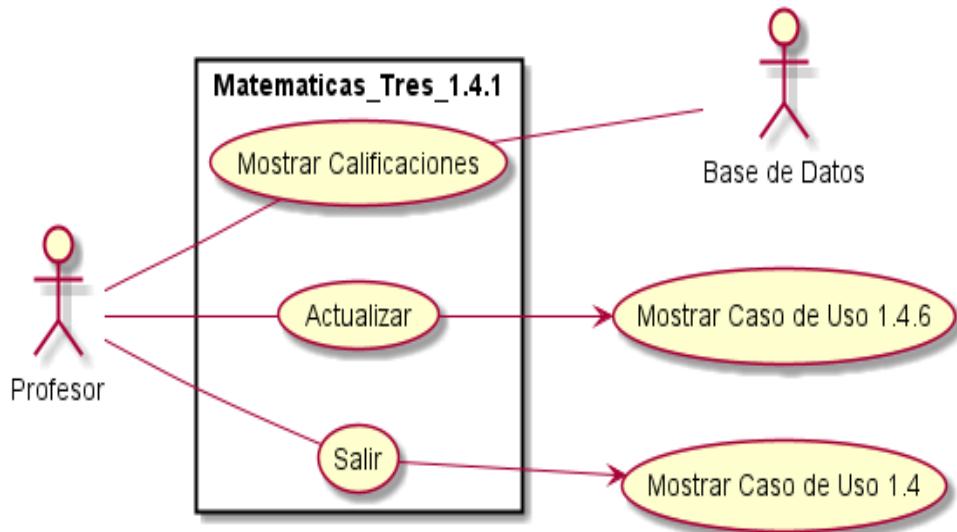


Figura 3.77: En la figura se muestra el caso de uso de Matematicas Tres

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.4.1	
Nombre	Matemáticas tres	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de tercer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de matemáticas 3	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón matemáticas 3 de la pantalla 1.4	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.4.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.4	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.78: Tabla del caso de uso de Matematicas Tres.

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Matematicas_Tres_1.4.1 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.4" as UC4
usecase "Mostrar Caso de Uso 1.4.6" as UC5
pf --- UC1
pf --- UC2
pf --- UC3
UC1 --- bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.79: Código de Matematicas Tres

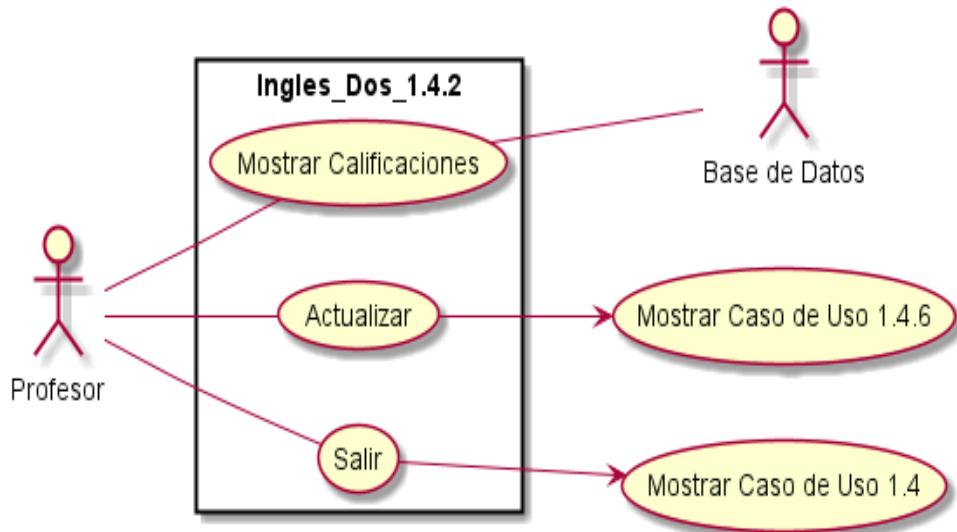


Figura 3.80: En la figura se muestra el caso de uso de Ingles Dos

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.4.2	
Nombre	Ingles dos	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de tercer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de ingles 2	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón ingles 2 de la pantalla 1.4	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.4.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.4	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.81: Tabla del caso de uso de Ingles Dos

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Ingles_Dos_1.4.2 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.4" as UC4
usecase "Mostrar Caso de Uso 1.4.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.82: Código de Ingles Dos

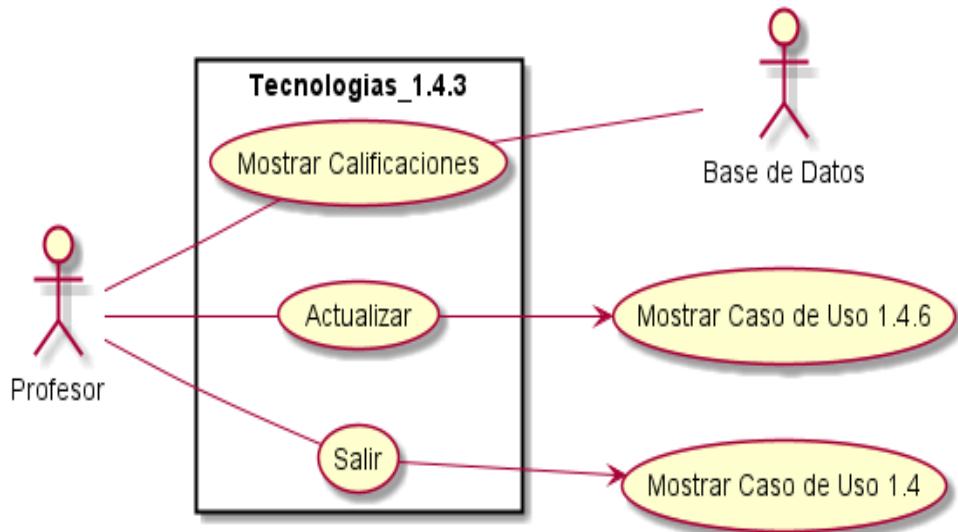


Figura 3.83: En la figura se muestra el caso de uso de Tecnologías

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.4.3		
Nombre	Tecnologías		
Creado	EJS	Actualizado:	GDCO
Fecha de creación	14/10/2021	Ultima versión:	01/11/2021
Actores	Maestro de tercer año, base de datos		
Descripción	Muestra la pantalla con las calificaciones de la materia de tecnologías		
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón tecnologías de la pantalla 1.4		
Precondiciones	N/A		
<u>Postcondiciones</u>	N/A		
Flujo normal	<p>1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar.</p> <p>2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.4.6 que contendrá los datos del alumno y los cuales se actualizarán.</p>		
Flujos alternos	Al terminar se regresa a la pantalla 1.4		
<u>Includes</u>			
Frecuencia de uso	Muy Frecuente		
Requerimientos especiales	N/A		
Supuestos			
Issues y notas			

Figura 3.84: Tabla del caso de uso de Tecnologías

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Tecnologias_1.4.3 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.4" as UC4
usecase "Mostrar Caso de Uso 1.4.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.85: Código de Tecnologías

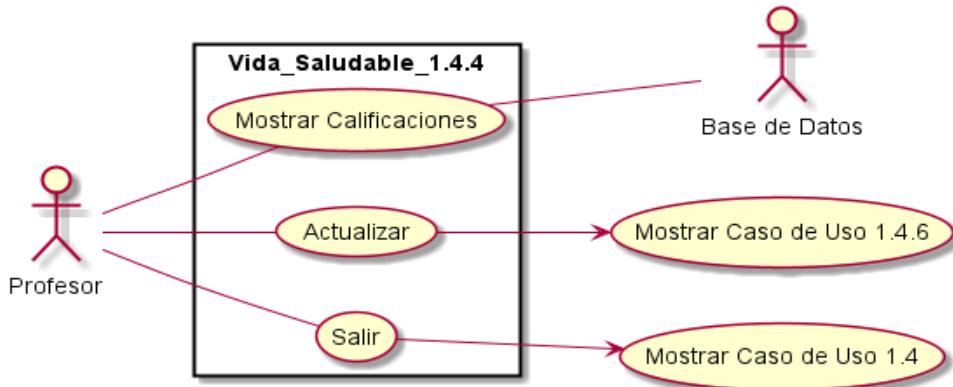


Figura 3.86: En la figura se muestra el caso de uso de Vida Saludable.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.4.4	
Nombre	Vida saludable	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Maestro de tercer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de vida saludable	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón vida saludable de la pantalla 1.4	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.4.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.4	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.87: Tabla del caso de uso de Vida Saludable..

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Vida_Saludable_1.4.4 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.4" as UC4
usecase "Mostrar Caso de Uso 1.4.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.88: Código de Vida Saludable

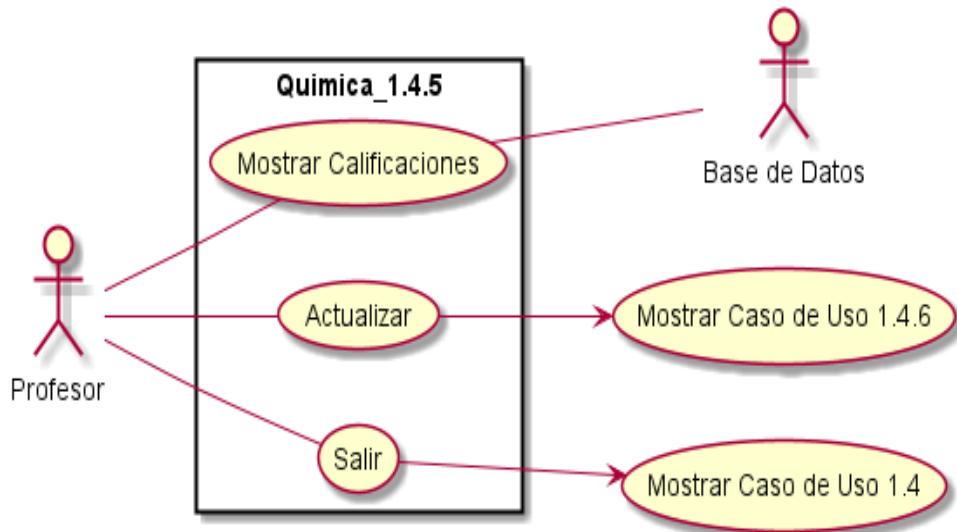


Figura 3.89: En la figura se muestra el caso de uso de Química.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.4.5	
Nombre	Química	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Última versión: 01/11/2021
Actores	Maestro de tercer año, base de datos	
Descripción	Muestra la pantalla con las calificaciones de la materia de química	
Disparador	Se manda esta pantalla al dar <u>click</u> en el botón química de la pantalla 1.4	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se elige la fila del alumno al cual se le va a actualizar y se le da <u>click</u> al botón actualizar. 2. Al dar <u>click</u> en actualizar se le mandará a la pantalla actualizar 1.4.6 que contendrá los datos del alumno y los cuales se actualizarán.	
Flujos alternos	Al terminar se regresa a la pantalla 1.4	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.90: Tabla de Química

```
@startuml
left to right direction
actor "Profesor" as pf
actor "Base de Datos" as bd
rectangle Quimica_1.4.5 {
    usecase "Mostrar Calificaciones" as UC1
    usecase "Actualizar" as UC2
    usecase "Salir" as UC3
}
usecase "Mostrar Caso de Uso 1.4" as UC4
usecase "Mostrar Caso de Uso 1.4.6" as UC5
pf -- UC1
pf -- UC2
pf -- UC3
UC1 --> bd
UC2 --> UC5
UC3 --> UC4
@enduml
```

Figura 3.91: Código de Química

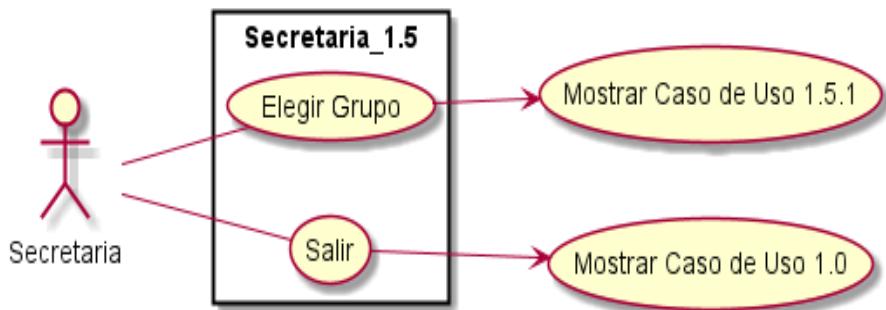


Figura 3.92: En la figura se muestra el caso de uso de Secretaria 1.5.

Un caso de uso del control escolar en la que se muestra la descripción de una acción o actividad. El diagrama de caso de uso del Control Escolar es una descripción de las actividades que deberá realizar alguien o algo para llevar a cabo algún proceso, en este caso los usuarios. Los personajes o entidades que participarán en este diagrama de caso de uso se denominan actores.

ID	CU-1.5	
Nombre	Pantalla de inicio de la secretaria	
Creado	EJS	Actualizado: GDCO
Fecha de creación	14/10/2021	Ultima versión: 01/11/2021
Actores	Secretaria	
Descripción	Muestra la pantalla de inicio de la secretaria con la opción de hacer modificaciones en los grupos de los alumnos	
Disparador	Se manda esta pantalla al terminar de ingresar en la pantalla 1.0 (Login)	
Precondiciones	N/A	
<u>Postcondiciones</u>	N/A	
Flujo normal	1. Se le da click al botón de actualizar y se envía a la pantalla 1.5.1 2. Se le da click al botón de salir y se cierra programa	
Flujos alternos	Al terminar se cierra el programa	
<u>Includes</u>		
Frecuencia de uso	Muy Frecuente	
Requerimientos especiales	N/A	
Supuestos		
Issues y notas		

Figura 3.93: Tabla del caso de Secretaria 1.5

```
@startuml
left to right direction
actor "Secretaria" as pf
rectangle Secretaria_1.5 {
    usecase "Elegir Grupo" as UC1
    usecase "Salir" as UC2
}
usecase "Mostrar Caso de Uso 1.0" as UC3
usecase "Mostrar Caso de Uso 1.5.1" as UC4
pf -- UC1
pf -- UC2
UC1 --> UC4
UC2 --> UC3
@enduml
```

Figura 3.94: Código de Secretaria 1.5

3.3.2. Diagrama de Estado

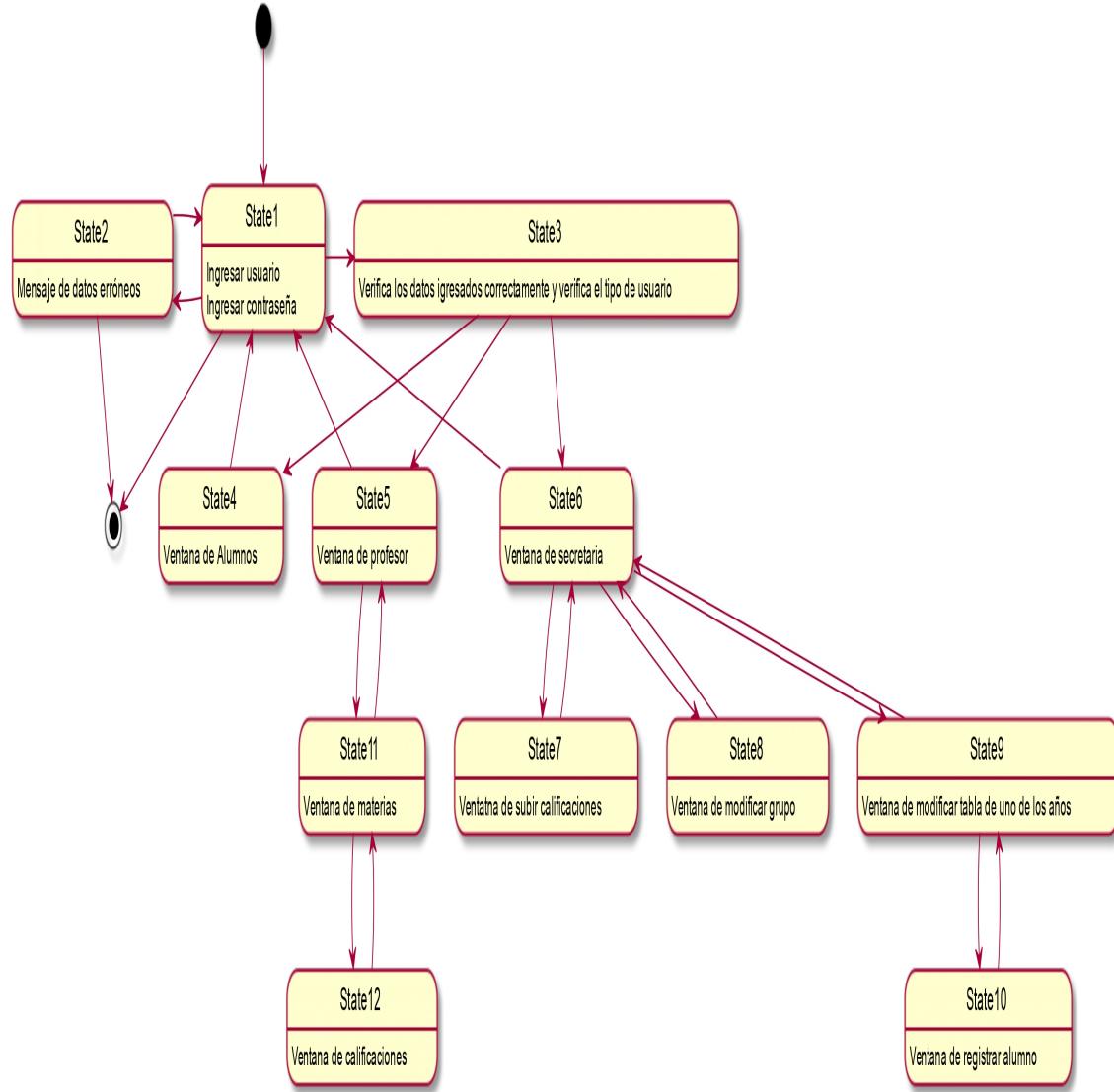


Figura 3.95: Diagrama de estado

Tiene un estado inicial y un estado final, este diagrama nos muestra una navegación por nuestro sistema de manera un poco más eficiente, sabiendo que pasa en determinados momentos.

```
1 @startuml
2
3
4
5 State1 : Ingresar usuario
6 State1 : Ingresar contraseña
7
8
9
10 State2 : Mensajes erróneos
11
12
13 State3 : Verifica los datos ingresados correctamente y verifica el tipo de usuario
14
15
16 State4 : Ventana alumnos
17
18
19 State5 : Ventana de profesores
20
21 State6 : Ventana de secretaria
22
23
24 |
25 State7 : Ventana subir calificaciones
26
27
28 State8 : Venatana modificar grupo
29
30
31 State9 : Venatana de modificar tabla de uno de los años
32
33
```

Output - DiagramaObjetos (run) x



Figura 3.96: Diagrama de estado código 1

```
43 [*] --> State1
44 State1--> State2
45 State2--> State1
46 State1--> State3
47 State1--> State1
48 State4--> State1
49 State5--> State1
50 State6--> State1
51 State3--> State4
52 State3--> State5
53 State3--> State6
54 State11--> State5
55 State5--> State11
56 State7--> State6
57 State6--> State7
58 State8--> State6
59 State6--> State8
60 State9--> State6
61 State6--> State9
62 State9--> State10
63 State10--> State9
64 State11--> State12
65 State12--> State11
66 State1--> [*]
67 State2--> [*]
68
69
70
```

Figura 3.97: Diagrama de estado código 2

3.3.3. Diagrama de Secuencia

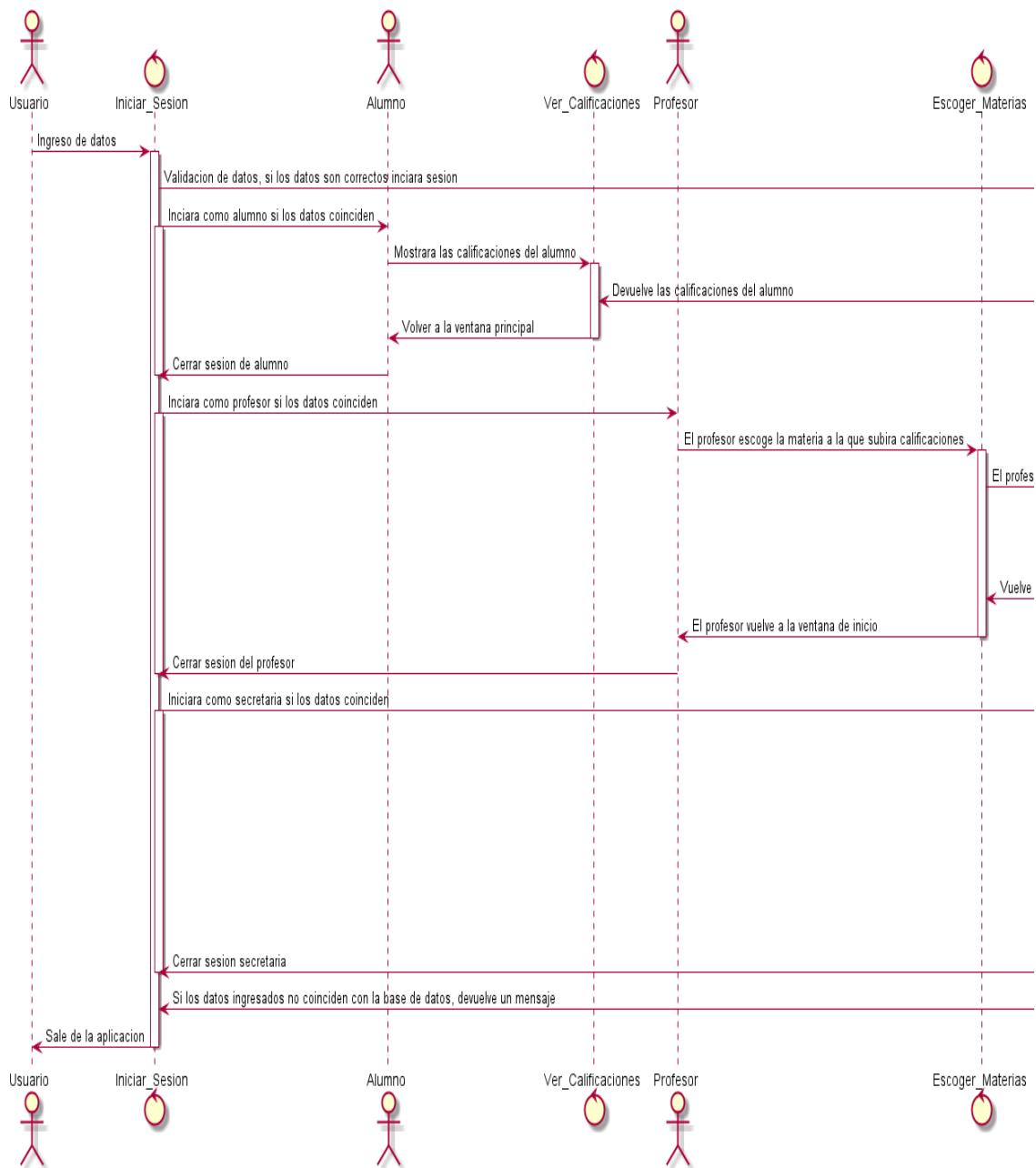


Figura 3.98: Diagrama de Secuencia parte 1

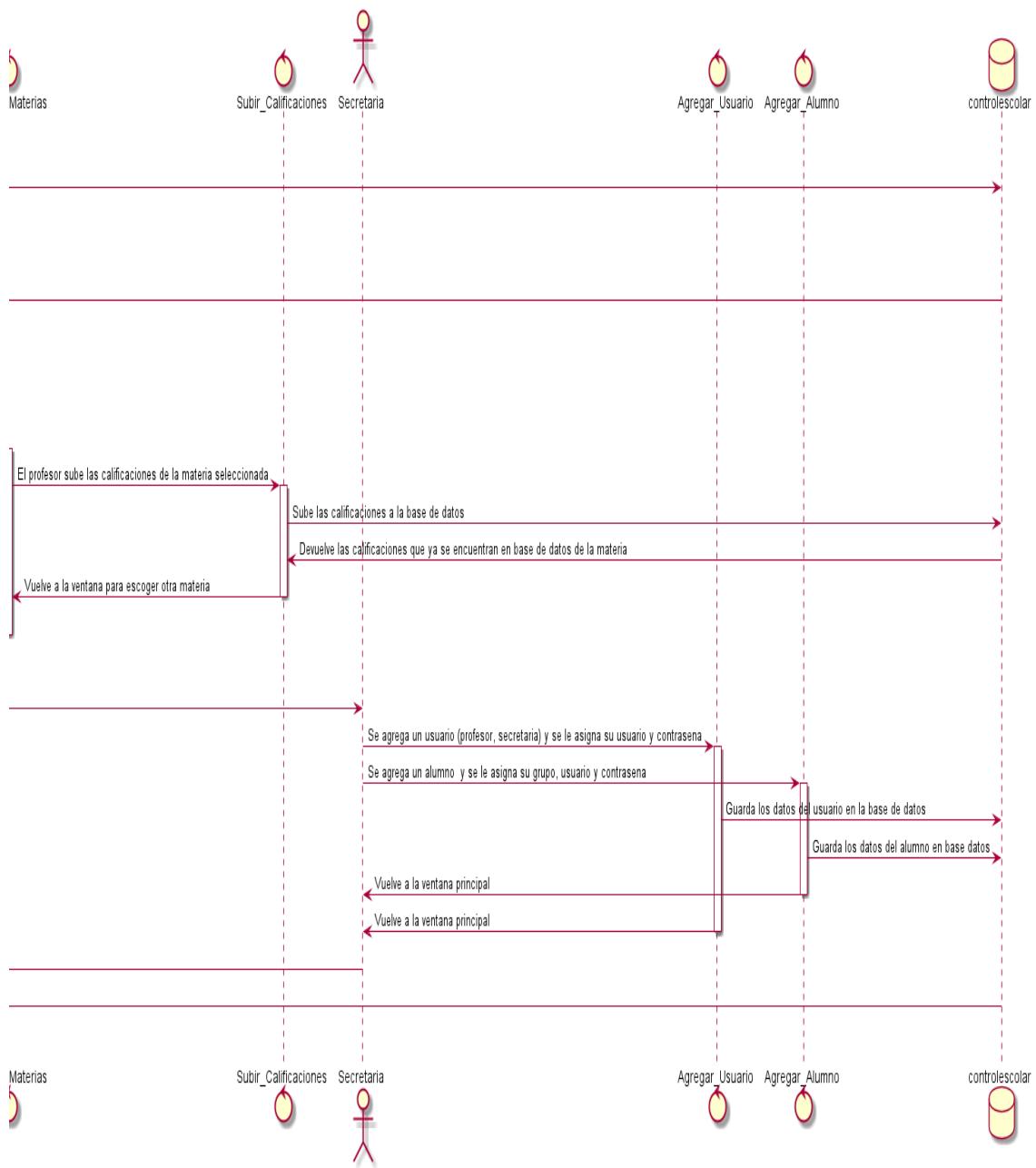


Figura 3.99: Diagrama de Secuencia parte 2

En este diagrama se representa el flujo que deben seguir los usuarios con sus respectivos roles.

```

@startuml
actor Usuario
control Iniciar_Sesion
actor Alumno
control Ver_Calificaciones
actor Profesor
control Escoger_Materias
control Subir_Calificaciones
actor Secretaria
control Agregar_Usuario
control Agregar_Alumno
database controlescolar

Usuario -> Iniciar_Sesion : Ingreso de datos
activate Iniciar_Sesion
Iniciar_Sesion --> controlescolar: Validacion de datos, si los datos son correctos inciara sesion

Iniciar_Sesion --> Alumno : Inciara como alumno si los datos coinciden
activate Iniciar_Sesion
Alumno -> Ver_Calificaciones : Mostrara las calificaciones del alumno
activate Ver_Calificaciones
controlescolar --> Ver_Calificaciones : Devuelve las calificaciones del alumno
Ver_Calificaciones --> Alumno : Volver a la ventana principal
deactivate Ver_Calificaciones
Alumno -> Iniciar_Sesion : Cerrar sesion de alumno
deactivate Iniciar_Sesion

```

Figura 3.100: Diagrama de Secuencia parte 1 código

```

Iniciar_Sesion -> Profesor : Inciara como profesor si los datos coinciden
activate Iniciar_Sesion
Profesor -> Escoger_Materias : El profesor escoge la materia a la que subira calificaciones
activate Escoger_Materias
Escoger_Materias -> Subir_Calificaciones : El profesor sube las calificaciones de la materia seleccionada
activate Subir_Calificaciones
Subir_Calificaciones -> controlescolar : Sube las calificaciones a la base de datos
controlescolar -> Subir_Calificaciones : Devuelve las calificaciones que ya se encuentran en base de datos de la materia
Subir_Calificaciones -> Escoger_Materias : Vuelve a la ventana para escoger otra materia
deactivate Subir_Calificaciones
Escoger_Materias -> Profesor : El profesor vuelve a la ventana de inicio
deactivate Escoger_Materias
Profesor -> Iniciar_Sesion : Cerrar sesion del profesor
deactivate Iniciar_Sesion


Iniciar_Sesion -> Secretaria : Iniciara como secretaria si los datos coinciden
activate Iniciar_Sesion
Secretaria -> Agregar_Usuario : Se agrega un usuario (profesor, secretaria) y se le asigna su usuario y contrasena
activate Agregar_Usuario
Secretaria -> Agregar_Alumno : Se agrega un alumno y se le asigna su grupo, usuario y contrasena
activate Agregar_Alumno
Agregar_Usuario -> controlescolar : Guarda los datos del usuario en la base de datos
Agregar_Alumno -> controlescolar : Guarda los datos del alumno en base datos
Agregar_Alumno -> Secretaria : Vuelve a la ventana principal
deactivate Agregar_Alumno
Agregar_Usuario -> Secretaria : Vuelve a la ventana principal
deactivate Agregar_Usuario
Secretaria -> Iniciar_Sesion : Cerrar sesion secretaria
deactivate Iniciar_Sesion

```

Figura 3.101: Diagrama de Secuencia parte 2 código

```

Iniciar_Sesion -> Profesor : Iniciara como profesor si los datos coinciden
activate Iniciar_Sesion
Profesor -> Escoger_Materias : El profesor escoge la materia a la que subira calificaciones
activate Escoger_Materias
Escoger_Materias -> Subir_Calificaciones : El profesor sube las calificaciones de la materia seleccionada
activate Subir_Calificaciones
Subir_Calificaciones -> controlescolar : Sube las calificaciones a la base de datos
controlescolar -> Subir_Calificaciones : Devuelve las calificaciones que ya se encuentran en base de datos de la materia
Subir_Calificaciones -> Escoger_Materias : Vuelve a la ventana para escoger otra materia
deactivate Subir_Calificaciones
Escoger_Materias -> Profesor : El profesor vuelve a la ventana de inicio
deactivate Escoger_Materias
Profesor -> Iniciar_Sesion : Cerrar sesion del profesor
deactivate Iniciar_Sesion

Iniciar_Sesion -> Secretaria : Iniciara como secretaria si los datos coinciden
activate Iniciar_Sesion
Secretaria -> Agregar_Usuario : Se agrega un usuario (profesor, secretaria) y se le asigna su usuario y contrasena
activate Agregar_Usuario
Secretaria -> Agregar_Alumno : Se agrega un alumno y se le asigna su grupo, usuario y contrasena
activate Agregar_Alumno
Agregar_Usuario -> controlescolar : Guarda los datos del usuario en la base de datos
Agregar_Alumno -> controlescolar : Guarda los datos del alumno en base datos
Agregar_Alumno -> Secretaria : Vuelve a la ventana principal
deactivate Agregar_Alumno
Agregar_Usuario -> Secretaria : Vuelve a la ventana principal
deactivate Agregar_Usuario
Secretaria -> Iniciar_Sesion : Cerrar sesion secretaria
deactivate Iniciar_Sesion

```

Figura 3.102: Diagrama de Secuencia parte 3 código

3.3.4. Diagrama de Colaboración

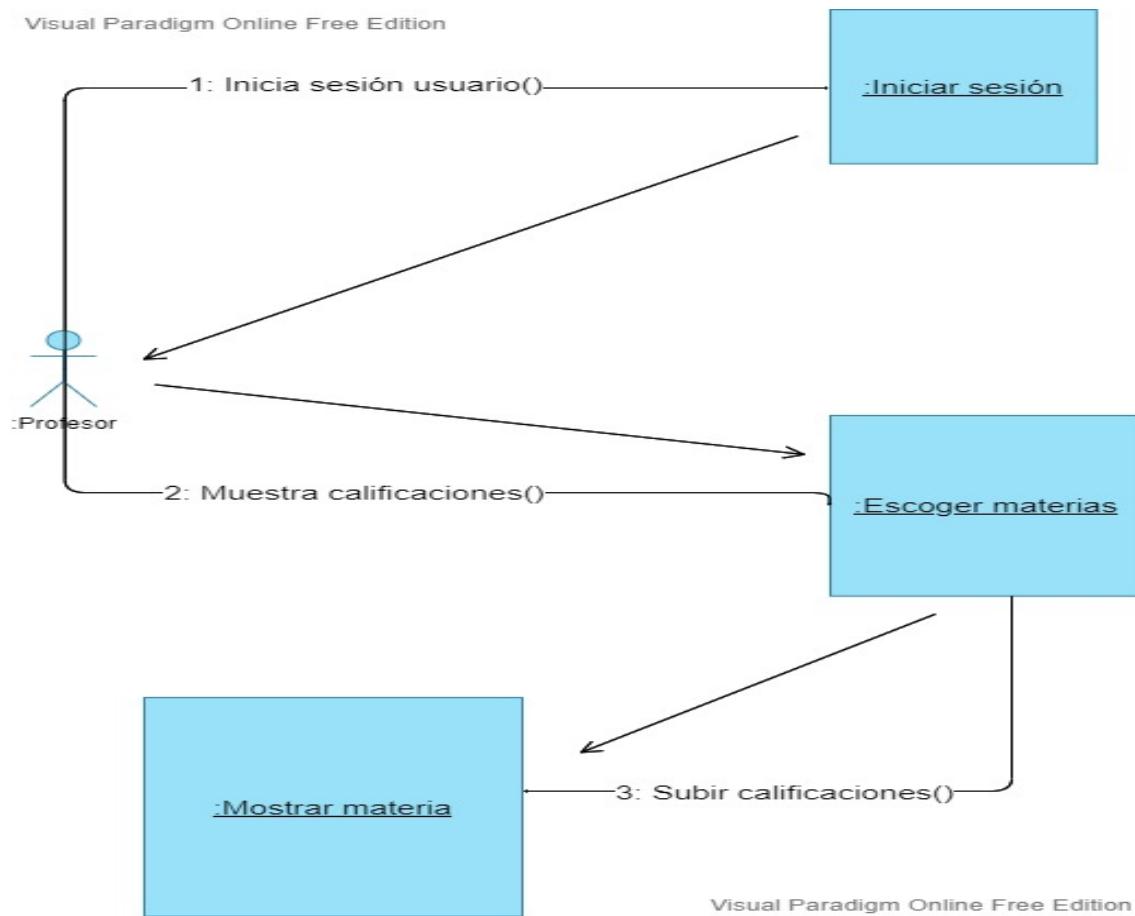


Figura 3.103: Diagrama de Colaboración.

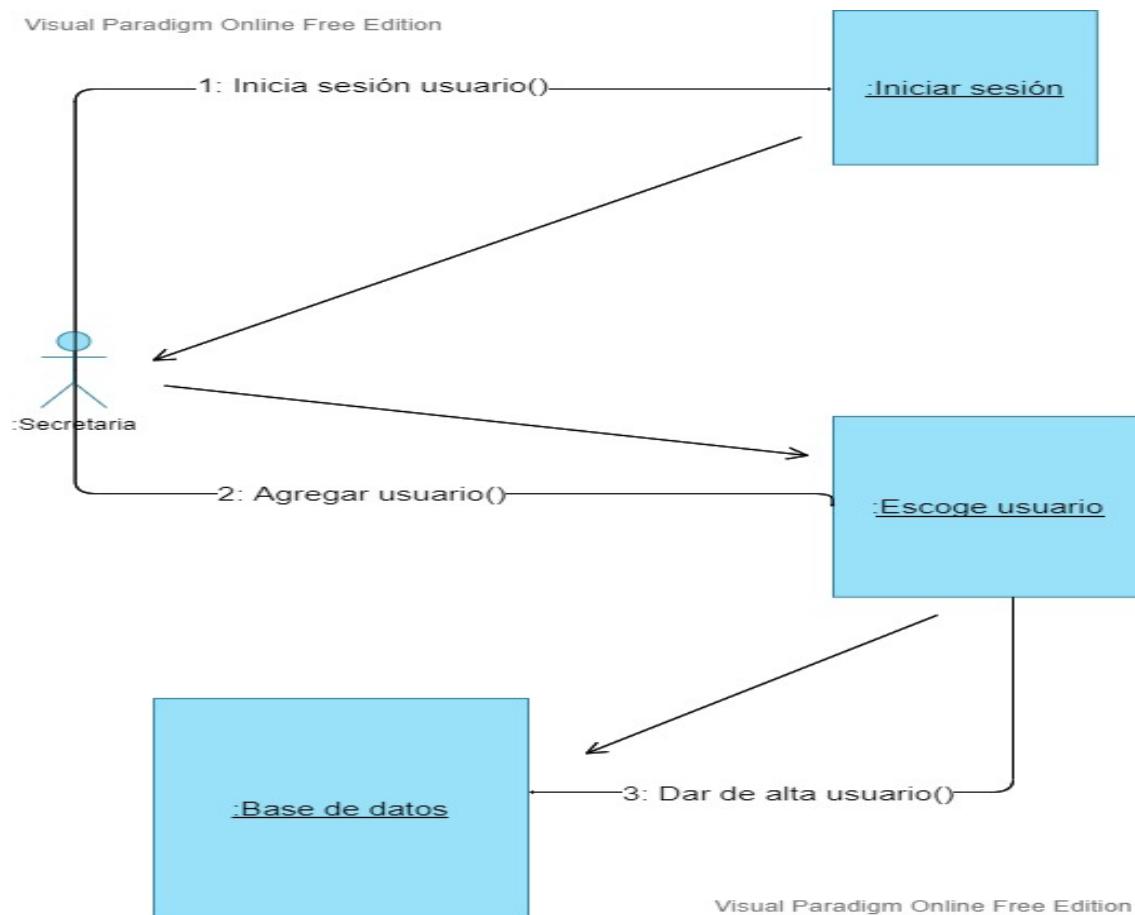


Figura 3.104: Diagrama de Colaboración.

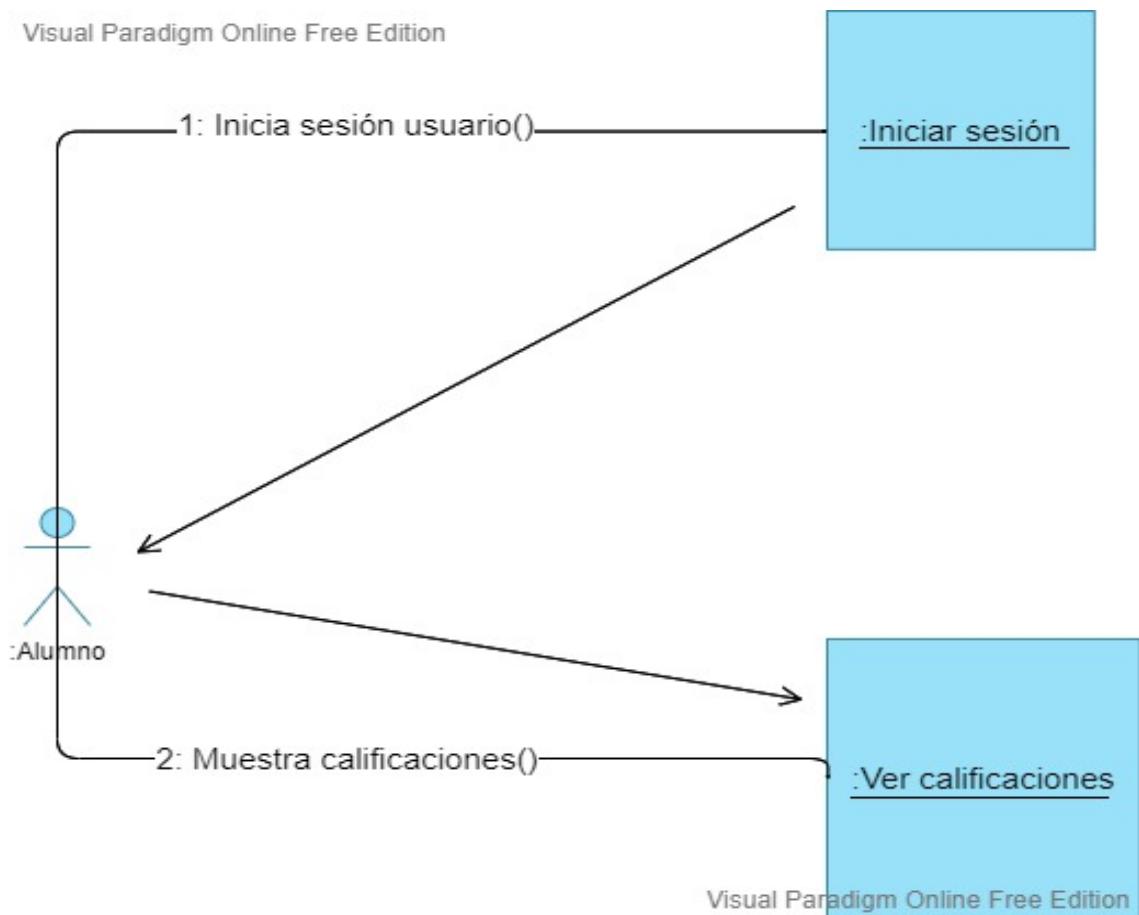


Figura 3.105: Diagrama de Colaboración.

3.3.5. Diagrama de Actividades

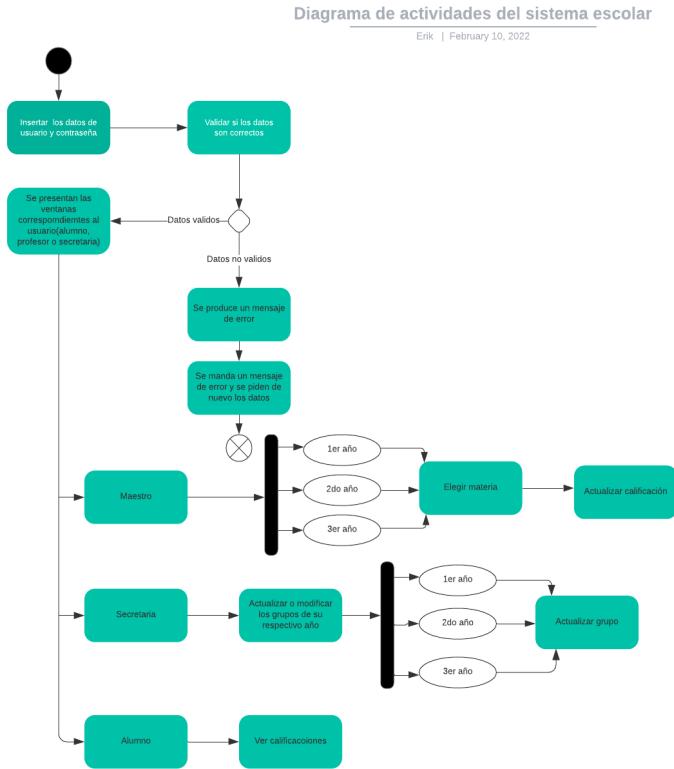


Figura 3.106: Diagrama de Actividades

El diagrama de actividades nos muestra todas la actividades que se pueden relizar en el programa

Capítulo 4

Diseño

4.1. Introducción

En este capítulo se abordara la descripción de nuestro sistema de control escolar en donde se describirán las diversas funcionalidades que implementamos para que pudiera ser funcional y de ayuda para el usuario, en esta sección también se mencionará su arquitectura, de forma física y lógica del sistema de control escolar, también se mostrará el estándar de codificación que utilizamos para la realización de nuestro proyecto desarrollado en java, y se describirá a detalle el diseño de datos de nuestra base de datos, para que pueda haber una mayor comprensión sobre nuestro sistema, de igual manera se vera el el modelo conceptual, el modelo lógico y diccionario de datos, los cuales nos permitieron llevar un mejor desarrollo en el diseño de nuestra aplicación, ya que con todos estos puntos se puede tener un mayor conocimiento conceptual del sistema.

4.2. Descripción del sistema

4.3. Arquitectura del sistema

4.3.1. Arquitectura física

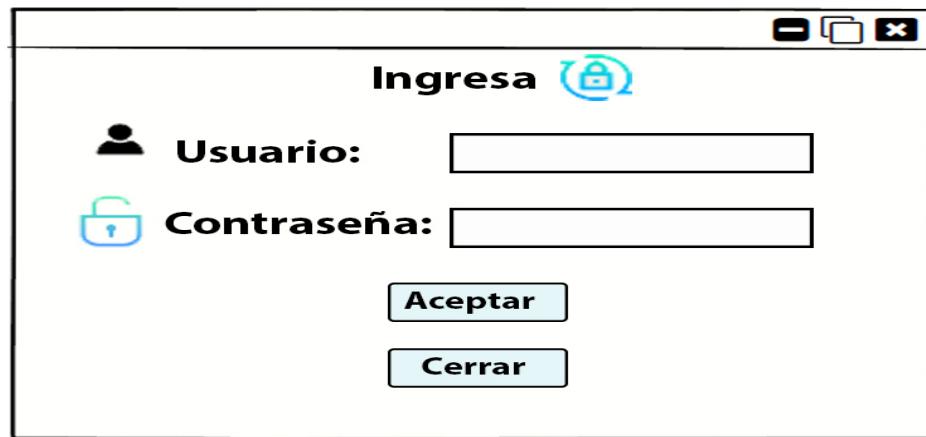


Figura 4.1: Imagen de inicio

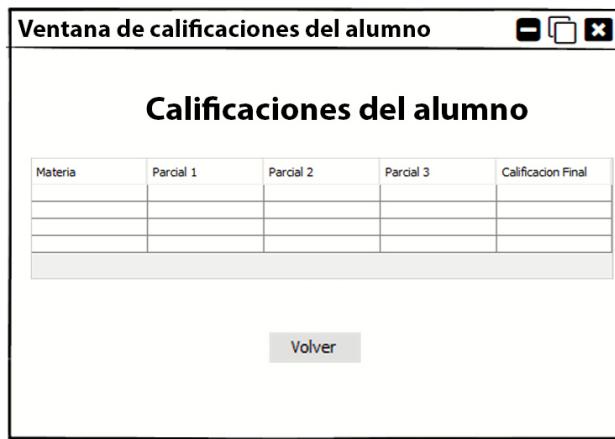


Figura 4.2: Imagen pantalla de alumnos



Figura 4.3: Imagen pantalla de maestros 1

Calificaciones 1

Inicio

Nombre	Grupo:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Subir Calificacion

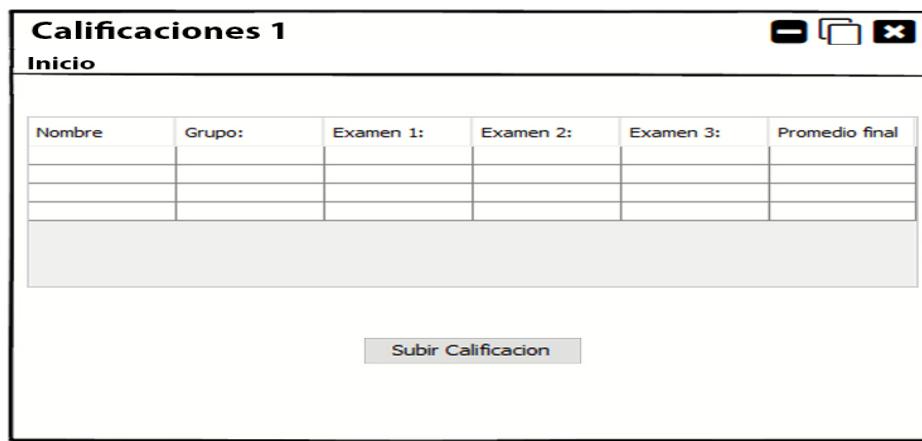


Figura 4.4: Imagen pantalla de subir calificaciones 1

Ventana Maestro2/Profesor

Inicio

Bienvenido a tu perfil

Subir Calificacion

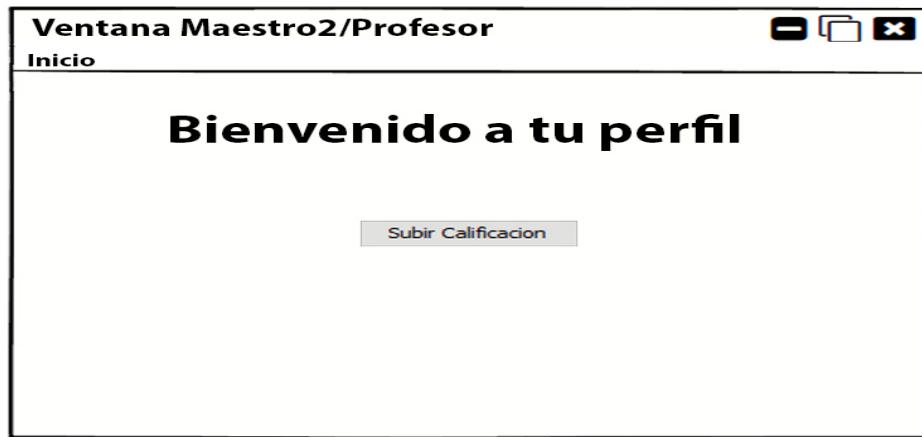


Figura 4.5: Imagen pantalla de maestros 2

Calificaciones 2

Inicio

Nombre	Grupo:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Subir Calificacion

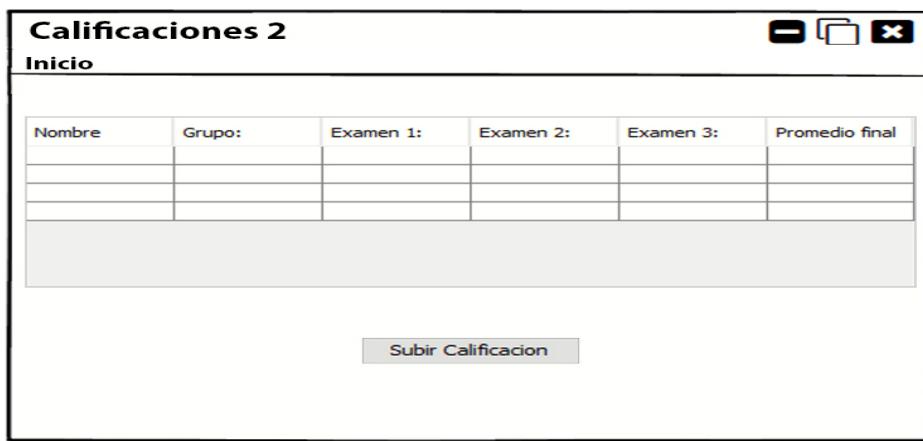


Figura 4.6: Imagen pantalla de subir calificaciones 2



Figura 4.7: Imagen pantalla de maestros 3

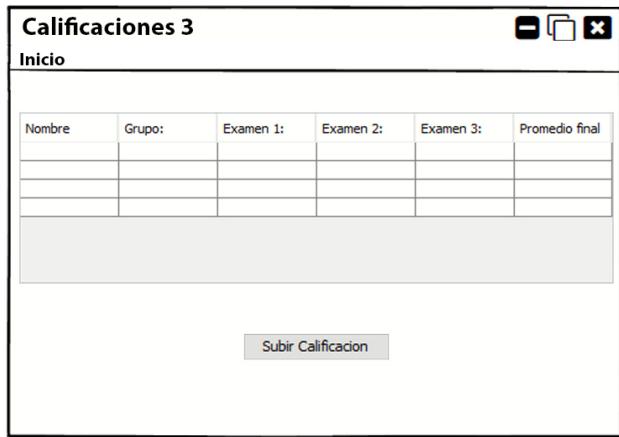


Figura 4.8: Imagen pantalla de subir calificaciones 3



Figura 4.9: Imagen ventana secretaria



Figura 4.10: Imagen modificar grupo



Figura 4.11: Imagen modificar tabla secretaria

Registrar alumno

Nombre:	<input type="text"/>
Apellido Paterno:	<input type="text"/>
Apellido Materno:	<input type="text"/>
Contraseña:	<input type="text"/>
Grupo:	1-A <input type="button" value="▼"/>
<input type="button" value="Aceptar"/>	

Figura 4.12: Imagen registrar alumno



Figura 4.13: Imagen pantalla de materias 1

Calificaciones

Ciencias I

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This screenshot shows a Windows application window titled 'Calificaciones'. The main title bar says 'Calificaciones' and has standard window controls (minimize, maximize, close). Below the title bar, the window is titled 'Ciencias I'. Inside the window, there is a table with columns labeled 'Nombre', 'Grupo:', 'Grado:', 'Examen 1:', 'Examen 2:', 'Examen 3:', and 'Promedio final'. There are five rows available for input. At the bottom of the window is a grey button labeled 'Actualizar'.

Figura 4.14: Imagen ciencias 1

Calificaciones

Geografía I

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This screenshot shows a Windows application window titled 'Calificaciones'. The main title bar says 'Calificaciones' and has standard window controls (minimize, maximize, close). Below the title bar, the window is titled 'Geografía I'. Inside the window, there is a table with columns labeled 'Nombre', 'Grupo:', 'Grado:', 'Examen 1:', 'Examen 2:', 'Examen 3:', and 'Promedio final'. There are five rows available for input. At the bottom of the window is a grey button labeled 'Actualizar'.

Figura 4.15: Imagen geografia

Calificaciones

Matemáticas I

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This figure shows a screenshot of a Microsoft Excel spreadsheet window titled "Calificaciones". The title bar also includes the subject "Matemáticas I". The spreadsheet contains a table with seven columns: Nombre, Grupo:, Grado:, Examen 1:, Examen 2:, Examen 3:, and Promedio final. There are five rows of data, each consisting of a single row of empty cells. At the bottom center of the spreadsheet area is a grey rectangular button labeled "Actualizar". The overall interface is consistent with older versions of Microsoft Office.

Figura 4.16: Imagen matemáticas 1

Calificaciones

Español I

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This figure shows a screenshot of a Microsoft Excel spreadsheet window titled "Calificaciones". The title bar also includes the subject "Español I". The spreadsheet contains a table with seven columns: Nombre, Grupo:, Grado:, Examen 1:, Examen 2:, Examen 3:, and Promedio final. There are five rows of data, each consisting of a single row of empty cells. At the bottom center of the spreadsheet area is a grey rectangular button labeled "Actualizar". The overall interface is consistent with older versions of Microsoft Office.

Figura 4.17: Imagen español 1

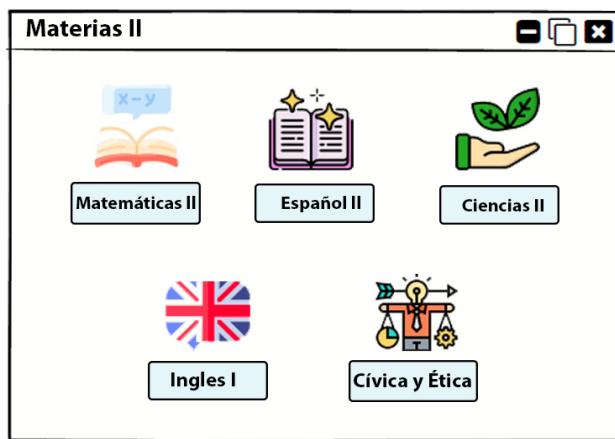


Figura 4.18: Imagen pantalla de materias 2

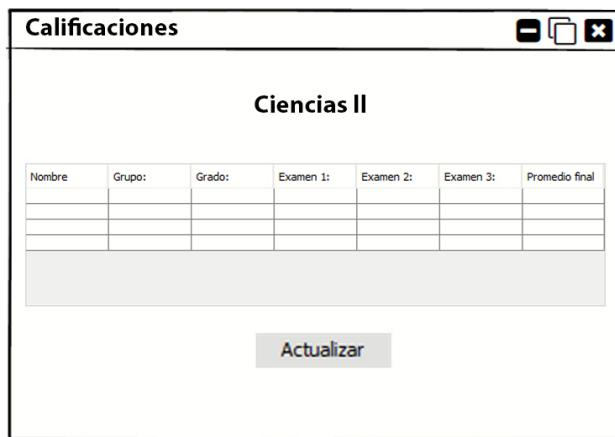


Figura 4.19: Imagen ciencias 2

Calificaciones

Español II

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This screenshot shows a software application window titled "Calificaciones". Inside, there is a section titled "Español II". Below the title is a table with columns for Nombre, Grupo, Grado, Examen 1, Examen 2, Examen 3, and Promedio final. There are five rows available for input, each consisting of a thin black border around the individual cells. A large gray rectangular area is positioned below the table. At the bottom center of this area is a button labeled "Actualizar". The window has standard operating system controls (minimize, maximize, close) at the top right.

Figura 4.20: Imagen español 2

Calificaciones

Cívica y Ética

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This screenshot shows a software application window titled "Calificaciones". Inside, there is a section titled "Cívica y Ética". Below the title is a table with columns for Nombre, Grupo, Grado, Examen 1, Examen 2, Examen 3, and Promedio final. There are five rows available for input, each consisting of a thin black border around the individual cells. A large gray rectangular area is positioned below the table. At the bottom center of this area is a button labeled "Actualizar". The window has standard operating system controls (minimize, maximize, close) at the top right.

Figura 4.21: Imagen cívica y ética

Calificaciones

Ingles I

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar



Figura 4.22: Imagen ingles 1

Calificaciones

Matemáticas II

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

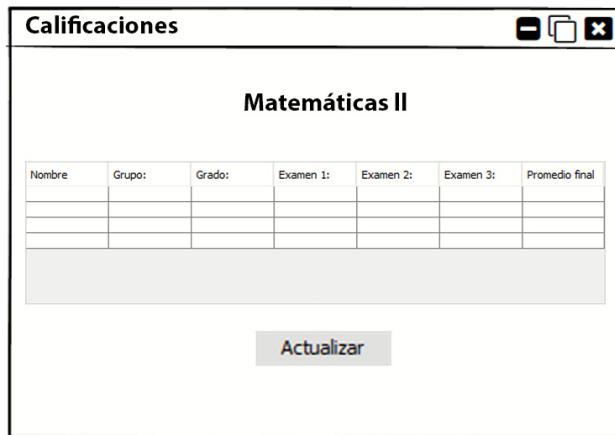


Figura 4.23: Imagen matematicas 2

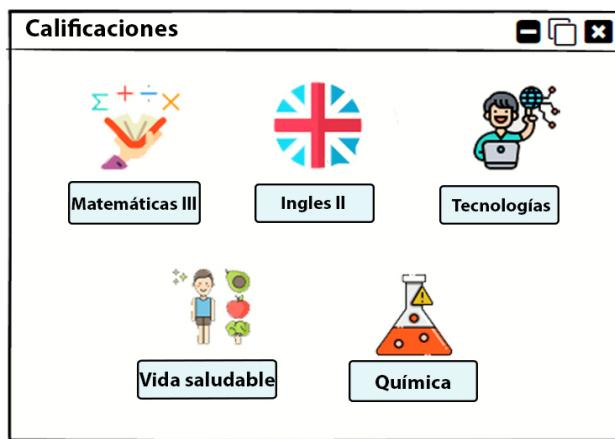


Figura 4.24: Imagen materias 3

The screenshot shows a software window titled "Calificaciones". The title bar also displays "Ingles II". Below the title, there is a table with columns for "Nombre", "Grupo:", "Grado:", "Examen 1:", "Examen 2:", "Examen 3:", and "Promedio final". There are four rows of empty cells for inputting student information. At the bottom of the window is a grey button labeled "Actualizar".

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Figura 4.25: Imagen inglés 2

Calificaciones

Matemáticas III

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This screenshot shows a Microsoft Excel window titled "Calificaciones". Inside, there is a specific sheet for "Matemáticas III". The sheet contains a table with seven columns: Nombre, Grupo:, Grado:, Examen 1:, Examen 2:, Examen 3:, and Promedio final. Below the table are five rows of data, each consisting of empty cells. At the bottom of the sheet is a grey button labeled "Actualizar".

Figura 4.26: Imagen matemáticas 3

Calificaciones

Química

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This screenshot shows a Microsoft Excel window titled "Calificaciones". Inside, there is a specific sheet for "Química". The sheet contains a table with seven columns: Nombre, Grupo:, Grado:, Examen 1:, Examen 2:, Examen 3:, and Promedio final. Below the table are five rows of data, each consisting of empty cells. At the bottom of the sheet is a grey button labeled "Actualizar".

Figura 4.27: Imagen química

Calificaciones

Tecnologías

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This figure shows a screenshot of a Microsoft Excel window titled "Calificaciones". Inside, there is a specific sheet titled "Tecnologías". The sheet contains a table with seven columns: Nombre, Grupo:, Grado:, Examen 1:, Examen 2:, Examen 3:, and Promedio final. Below the table is a large gray rectangular area, and at the bottom center of this area is a button labeled "Actualizar". The window has standard operating system controls (minimize, maximize, close) at the top right.

Figura 4.28: Imagen tecnologías

Calificaciones

Vida saludable

Nombre	Grupo:	Grado:	Examen 1:	Examen 2:	Examen 3:	Promedio final

Actualizar

This figure shows a screenshot of a Microsoft Excel window titled "Calificaciones". Inside, there is a specific sheet titled "Vida saludable". The sheet contains a table with seven columns: Nombre, Grupo:, Grado:, Examen 1:, Examen 2:, Examen 3:, and Promedio final. Below the table is a large gray rectangular area, and at the bottom center of this area is a button labeled "Actualizar". The window has standard operating system controls (minimize, maximize, close) at the top right.

Figura 4.29: Imagen vida saludable

4.3.2. Arquitectura lógica

4.4. Estándar de codificación



Figura 4.30: Estándar de codificación

Para este estándar de codificación se ocupará la notación húngara. La notación húngara hace referencia a la creación de nombre de las variables consiste en prefijos en minúsculas que se añaden a los nombres de las variables y que indican su tipo, ejemplo: intEdad, donde “int” es el tipo de dato el cual es entero y “Edad” es el nombre de la variable. También le hemos agregado que todas las primeras letras del nombre de las variables sean en mayúsculas, para una mejor distinción.

Para los comentarios hemos optado por todas las opciones disponibles para hacerlos “//” y “/**/”. La sangría aun esta en discusión ya que esta depende de cada estructura que ocupemos al igual de las clases y métodos, lo ideal sería hacer distinción entre las llaves de apertura y cierre con la cantidad de un tab en las líneas internas del método, ejemplo:

```
for (int i=1; i <= 10; i++)
System.out.println("El número de recorrido es "+i);
```

Así se vería la estructura for, y así se idearan los formatos de las demás estructuras, clases y métodos que ocuparemos dentro del proyecto.

4.5. Bases de datos

4.5.1. Diseño de datos

4.5.2. Diagrama conceptual

Esta base de datos cuenta con 6 tablas para tener el control del sistema escolar. Antes de crear las tablas, se tiene que crear los datos de tipo enumerado, uno de ellos es el de grupos que contendrá los grupos (1A,2A,3A). El otro tipo enumerado será el de roles que contendrá los roles (1ALU,2PRO,3SEC). Tenemos una tabla llamada Calificaciones que cuenta con las columnas IDC de tipo integer que es una llave primaria, las demás columnas son IDA, IDG, IDM , parcial1, parcial 2, parcial 3 y examen final todos de tipo integer. El IDA hará referencia a IDA de la tabla Alumnos. El IDM hará referencia a IDM de la tabla Materias. Esta tabla nos permitirá guardar las calificaciones de los alumnos en la base. Esta tabla se conectaría con la tabla llamada Grupo que contará con la columna IDG como llave primaria de tipo enumerado grupos. La tabla grupo guardara a los alumnos que pertenecen a cada grupo y con sus respectivos nombres. Además, la tabla Calificaciones tendrá una conexión a la tabla Materia con IDM como llave primaria de tipo integer y NombreMat de tipo varchar() como una columna. El IDG de la tabla Materias hará referencia al IDG de la tabla Grupo. En esta tabla se guardarán los nombres de las materias que se impartirán en la secundaria y con la llave primaria y las conexiones a las otras tablas se irán distribuyendo las materias a su respectivo año correspondiente al igual que su grupo. La tabla Grupo igualmente estará conectada a la tabla de Alumnos con IDA como llave primaria de tipo integer, IDR de tipo enumerado roles, IDG de tipo enumerado grupos, IDU de tipo integer, NombreAlu, ApellidoAlu, ContraseñaAlu estos tres campos serán de tipo varchar() que serán los demás campos de dicha tabla. El IDU de la tabla Alumnos hará referencia al IDU de la tabla Usuarios. El IDR de la tabla Alumnos hará referencia al IDR de la tabla Usuarios. La tabla Alumnos tendrá los datos de todos los alumnos, su nombre, apellido tanto paterno como materno y su contraseña con la que podrán ingresar al sistema para ver sus calificaciones La tabla Alumnos tendrá una conexión a la tabla Roles con IDR como llave primaria, de tipo enumerado, en esta tabla se manejarán los roles de alumno, maestro y secretaria y con ellos se tendrá un control de lo que podrán hacer dentro del sistema cada usuario. La tabla Alumnos también estará conectada Usuarios con IDU como llave primaria de tipo integer, IDR de tipo enumerado roles, NombreUsu, ApellidoUsu, ContraseñaUsu los tres de tipo varchar(). El IDR de la tabla Usuarios hará referencia al IDR de la tabla RolesEn esta tabla se manejarán ya todos los usuarios.

4.5.3. Diagrama Entidad-Relación

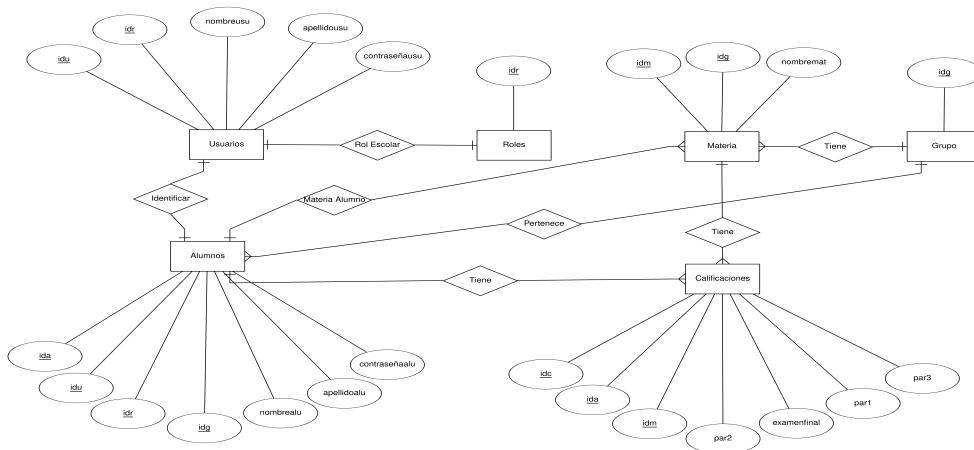


Figura 4.31: Diagrama Entidad-Relación del proyecto del control escolar.

4.5.4. Diagrama de Modelo Relacional

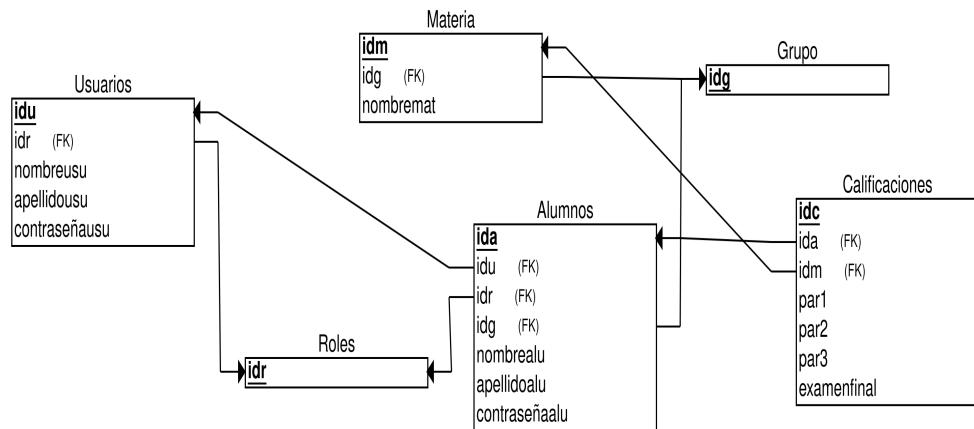


Figura 4.32: Diagrama de modelo Relacional del proyecto del control escolar.

4.5.5. Diagrama físico

```
1 create database if not exists control escolar;
2 use control escolar;
3
4 create type grupos as enum(
5     '1A',
6     '2A',
7     '3A'
8 );
9
10 create type roles as enum(
11     '1ALU',
12     '2PRO',
13     '3SEC'
14 );
15
16 create table Roles(
17     IDR roles
18 );
19
20 create table Usuarios(
21     IDU integer primary key,
22     IDR roles,
23     NombreUsu varchar(20),
24     ApellidoUsu varchar(20),
25     ContraseñaUsu varchar(20),
26     foreign key(IDR) references Roles(IDR) on delete restrict on update restrict
27 );
28
29 create table Alumnos(
30     IDA integer primary key,
31     IDU integer,
32     IDR roles,
33     NombreAlu varchar(20),
34     ApellidoAlu varchar(20),
35     ContraseñaAlu varchar(20),
36     IDG grupos,
37     foreign key(IDU) references Usuarios(IDU) on delete cascade on update cascade,
38     foreign key(IDR) references Usuarios(IDR) on delete cascade on update cascade
39     foreign key(NombreAlu) references Usuarios(NombreUsu) on delete cascade on update cascade
40     foreign key(ApellidoAlu) references Usuarios(ApellidoUsu) on delete cascade on update cascade
41     foreign key(ContraseñaAlu) references Usuarios(ContraseñaUsu) on delete cascade on update cascade
42     foreign key(IDG) references Grupo(IDG) on delete cascade on update cascade
43 );
44
45
46 create table Materias(
47     IDM integer primary key,--Lo ideal sera hacer un tipo de dato materia, aun en proceso
48     IDG integer,
49     NombreMat varchar(20),
50     foreign key(IDG) references Grupo(IDG) on delete restrict on update restrict
51 );
52
53 create table Grupo(
54     IDG grupos
55 ).
```

Figura 4.33: Diagrama físico de la creación de base de datos.

```

13   '3SEC'
14 );
15
16 create table Roles(
17   IDR roles
18 );
19
20 create table Usuarios(
21   IDU integer primary key,
22   IDR roles,
23   NombreUsu varchar(20),
24   ApellidoUsu varchar(20),
25   ContraseñaUsu varchar(20),
26   foreign key(IDR) references Roles(IDR) on delete restrict on update restrict
27 );
28
29 create table Alumnos(
30   IDA integer primary key,
31   IDU integer,
32   IDR roles,
33   NombreAlu varchar(20),
34   ApellidoAlu varchar(20),
35   ContraseñaAlu varchar(20),
36   IDG grupos,
37   foreign key(IDU) references Usuarios(IDU) on delete cascade on update cascade,
38   foreign key(IDR) references Usuarios(IDR) on delete cascade on update cascade
39   foreign key(NombreAlu) references Usuarios(NombreUsu) on delete cascade on update cascade
40   foreign key(ApellidoAlu) references Usuarios(ApellidoUsu) on delete cascade on update cascade
41   foreign key(ContraseñaAlu) references Usuarios(ContraseñaUsu) on delete cascade on update cascade
42   foreign key(IDG) references Grupo(IDG) on delete cascade on update cascade
43 );
44
45
46 create table Materias(
47   IDM integer primary key,--Lo ideal sera hacer un tipo de dato materia, aun en proceso
48   IDG integer,
49   NombreMat varchar(20),
50   foreign key(IDG) references Grupo(IDG) on delete restrict on update restrict
51 );
52
53 create table Grupo(
54   IDG grupos
55 );
56
57 create table Calificaciones(
58   IDC integer primary key,
59   IDA integer,
60   IDM integer,
61   parcial1 integer,
62   parcial2 integer,
63   parcial3 integer,
64   examenfinal integer,
65   foreign key(IDA) references Alumnos(IDA) on delete restrict on update cascade,
66   foreign key(IDM) references Materias(IDM) on delete restrict on update cascade
67 );

```

Figura 4.34: Diagrama físico de la creación de base de datos.

4.5.6. Diccionario de datos

Esta tabla contiene 4 columnas con diferentes tipos de datos, los cuales incluye de tipo integer, varchar y de tipo enumerado, cuenta con solo una llave primaria. Los datos van referenciados a los usuarios y los roles.

TABLA ALUMNOS

Tabla Alumnos

	Columna	Tipo de dato	Descripción
PK	IDA	integer	identificador para los alumnos
FK	IDU	integer	identificador para los usuarios
FK	IDR	roles	Identificador para los roles que existen en la escuela.
FK	NombreAlu	varchar	nombre del alumno
FK	ApellidoAlu	varchar	apellido del alumno
FK	ContrasenaAlu	varchar	contraseña del alumno
	IDG	grupo	identificador para el grupo

Figura 4.35: Tabla Alumnos

Esta tabla contiene 4 columnas con tres tipos de datos, los cuales incluye de tipo varchar, integer y roles. Los datos se desglosan en datos que ayudaran al usuario a ingresar.

TABLA USUARIOS

Tabla Usuarios

	Columna	Tipo de dato	Descripción
PK	IDU	integer	identificador para los usuarios
FK	IDR	roles	Identificador para los roles que existen en la escuela.
	NombreUsu	varchar	nombre del usuario
	ApellidoUsu	varchar	apellido del usuario
	ContrasenaUsu	varchar	contraseña del usuario

Figura 4.36: Tabla Usuarios

Esta tabla contiene 4 columnas con solo un tipo de dato, que es el integer que contiene las llaves primarias.

TABLA CALIFICACIONES

Tabla Calificaciones

	Columna	Tipo de dato	Descripción
PK	IDC	integer	identificador para calificaciones
FK	IDA	integer	identificador de alumnos
FK	IDM	integer	identificador de materias
	parcial1	integer	calificación del parcial 1
	parcial2	integer	calificación del parcial 2
	parcial3	integer	calificación del parcial 3
	examenfinal	integer	calificación del examen final

Figura 4.37: Tabla Calificaiones

Esta tabla cuenta con cuatro columnas y una fila con un solo tipo de dato, que es el integer para identificar a los grupos.

TABLA GRUPO

Tabla Grupo

	Columna	Tipo de dato	Descripción
PK	IDG	integer	identificador para el grupo

Figura 4.38: Tabla Grupo

En esta tabla estan contenidos los nombres de las materias con sus respectivas llaves primarias referenciadas a otras tablas. Cuenta con solo el tipo de dato integer.

TABLA MATERIAS

Tabla Materias

	Columna	Tipo de dato	Descripción
PK	IDM	integer	identificador para las materias
FK	IDG	integer	identificador para el grupo
	NombreMat	varchar	nombre de la materia

Figura 4.39: Tabla Materias

En esta tabla se deseginan los roles que tendrá cada usuario dentro de la base de datos y el programa en sí, y su tipo de dato es de tipo enumerado rol.

TABLA ROLES

Tabla Roles

	Columna	Tipo de dato	Descripción
PK	IDR	roles	Identificador para los roles que existen en la escuela.

Figura 4.40: Tabla Roles

Capítulo 5

Implementación

5.1. Intruducción

La implementación es la parte del proceso en que los desarrolladores programan el código para el proyecto. Las pruebas del software son parte esencial del proceso de desarrollo del software. Esta parte del proceso tiene la función de detectar los errores de software lo antes posible. Durante el desarrollo del software se realiza la documentación del diseño del código interiormente como exteriormente esto con el objetivo de facilitar su mejora y su mantenimiento futuro.

5.2. Código fuente

```
Codigo de Framer ControlEscolar
import javax.swing.JOptionPane;
import javax.swing.UIManager;
import java.sql.*;
import java.util.Random;
import javax.swing.table.DefaultTableModel;

public class Ventana_principal extends javax.swing.JFrame {

    Conexion con = new Conexion();
    String bd = "controlescolar";
    String url = null;
    String table = "prueba";
    DefaultTableModel modelo = new DefaultTableModel();
    DefaultTableModel modelo1 = new DefaultTableModel();

    public Ventana_principal() {
        initComponents();
        this.setTitle("Registro Productos");
        this.setLocationRelativeTo(null);
        modelo = (DefaultTableModel) TablaAlumno.
getModel();
        modelo1 = (DefaultTableModel) CalificacionesMaestro.
getModel();
    }
}
```

```

}

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

    dglAlumno = new javax.swing.JDialog();
    Panel1 = new javax.swing.JPanel();
    btnVolverAlumno = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();
    TablaAlumno = new javax.swing.JTable();
    lblTituloAlumno1 = new javax.swing.JLabel();
    txtAceptarAlum = new javax.swing.JButton();
    dglModificarGrupo = new javax.swing.JDialog();
    lblTituloModificar = new javax.swing.JLabel();
    bttAgregarAlumno = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    mnbMenu2 = new javax.swing.JMenuBar();
    mnbPrograma = new javax.swing.JMenu();
    btnmnSalir2 = new javax.swing.JMenuItem();
    dglNuevoAlumno = new javax.swing.JDialog();
    lblTituloAlumno = new javax.swing.JLabel();
    lblNombre = new javax.swing.JLabel();
    lblApellidoP = new javax.swing.JLabel();
    lblApellidoM = new javax.swing.JLabel();
    lblGrupo = new javax.swing.JLabel();
    txtNombre = new javax.swing.JTextField();
    txtApellidoP = new javax.swing.JTextField();
    txtApellidoM = new javax.swing.JTextField();
    btnAceptarNuevo = new javax.swing.JButton();
    listGrupo = new javax.swing.JComboBox<>();
    jLabel1 = new javax.swing.JLabel();
    txtContraseña1 = new javax.swing.JTextField();
    dglMaterias1 = new javax.swing.JDialog();
    Panel3 = new javax.swing.JPanel();
    btnMatematicas = new javax.swing.JButton();
    btnEspañol = new javax.swing.JButton();
    btnCiencias = new javax.swing.JButton();
    lblMatematicas = new javax.swing.JLabel();
    lblEspañol = new javax.swing.JLabel();
    lblCiencias = new javax.swing.JLabel();
    btnGeoagrafia = new javax.swing.JButton();
    btnHistoria = new javax.swing.JButton();
    lblHistoria = new javax.swing.JLabel();
    lblGeografia = new javax.swing.JLabel();
    mnbMenu4 = new javax.swing.JMenuBar();
    VentanaMaterias1 = new javax.swing.JMenu();
    btnmnSalirMaterias1 = new javax.swing.JMenuItem();
    dglMaterias2 = new javax.swing.JDialog();
}

```

```

Panel4 = new javax.swing.JPanel();
btnMatematicas2 = new javax.swing.JButton();
btnEspañol2 = new javax.swing.JButton();
btnCiencias2 = new javax.swing.JButton();
btnIngles1 = new javax.swing.JButton();
btnCivicayEtica = new javax.swing.JButton();
lblMatematicas2 = new javax.swing.JLabel();
lblIngles1 = new javax.swing.JLabel();
lblCivicayEtica = new javax.swing.JLabel();
lblEspañol2 = new javax.swing.JLabel();
lblCiencias2 = new javax.swing.JLabel();
mnbMenu5 = new javax.swing.JMenuBar();
VentanaMaterias2 = new javax.swing.JMenu();
bttnmnSalirMaterias2 = new javax.swing.JMenuItem();
dglMaterias3 = new javax.swing.JDialog();
Panel5 = new javax.swing.JPanel();
btnMatematicas3 = new javax.swing.JButton();
btnIngles2 = new javax.swing.JButton();
btnTecnologias = new javax.swing.JButton();
btnQuimica = new javax.swing.JButton();
btnVidasaludable = new javax.swing.JButton();
lblIngles2 = new javax.swing.JLabel();
lblMatematicas3 = new javax.swing.JLabel();
lblTecnologias = new javax.swing.JLabel();
lblVidasaludable = new javax.swing.JLabel();
lblQuimica = new javax.swing.JLabel();
mnbMenu6 = new javax.swing.JMenuBar();
VentanaMaterias3 = new javax.swing.JMenu();
bttnmnSalirMaterias3 = new javax.swing.JMenuItem();
dglSecretaria = new javax.swing.JDialog();
txtTituloSecretaria = new javax.swing.JTextField();
bttModificarGrupo = new javax.swing.JButton();
mnbMenu12 = new javax.swing.JMenuBar();
mnPrograma4 = new javax.swing.JMenu();
bttnmnSalirSecretaria = new javax.swing.JMenuItem();
dglCalificaciones = new javax.swing.JDialog();
btnActualizarVidasaludable = new javax.swing.JButton();
jScrollPane2 = new javax.swing.JScrollPane();
CalificacionesMaestro = new javax.swing.JTable();
bttVol1 = new javax.swing.JButton();
bttVol2 = new javax.swing.JButton();
bttVol3 = new javax.swing.JButton();
lblTitulo = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
materiafiltr = new javax.swing.JComboBox();
grupofiltr = new javax.swing.JComboBox();
jButton1 = new javax.swing.JButton();
mnbMenu20 = new javax.swing.JMenuBar();
dglMaestro1 = new javax.swing.JDialog();

```

```
lblTituloMaestro1 = new javax.swing.JLabel();
bttSubirCalificaciónM1 = new javax.swing.JButton();
mnbMenu1 = new javax.swing.JMenuBar();
mnPrograma1 = new javax.swing.JMenu();
btnmnSalirMaestro1 = new javax.swing.JMenuItem();
dglMaestro2 = new javax.swing.JDialog();
lblTituloMaestro2 = new javax.swing.JLabel();
bttSubirCalificaciónM2 = new javax.swing.JButton();
mnbMenu22 = new javax.swing.JMenuBar();
mnPrograma2 = new javax.swing.JMenu();
btnmnSalirMaestro2 = new javax.swing.JMenuItem();
dglMaestro3 = new javax.swing.JDialog();
lblTituloMaestro3 = new javax.swing.JLabel();
bttSubirCalificaciónM3 = new javax.swing.JButton();
mnbMenu23 = new javax.swing.JMenuBar();
mnPrograma3 = new javax.swing.JMenu();
btnmnSalirMaestro3 = new javax.swing.JMenuItem();
CrearUsuario = new javax.swing.JDialog();
ApellidoPusu = new javax.swing.JTextField();
ApellidoMusu = new javax.swing.JTextField();
btnAceptarNuevo1 = new javax.swing.JButton();
listRol = new javax.swing.JComboBox<>();
jLabel4 = new javax.swing.JLabel();
lblTituloAlumno2 = new javax.swing.JLabel();
ContraseñaUsu = new javax.swing.JTextField();
lblNombre1 = new javax.swing.JLabel();
lblApellidoP1 = new javax.swing.JLabel();
lblApellidoM1 = new javax.swing.JLabel();
lblGrupo1 = new javax.swing.JLabel();
NombreUsu = new javax.swing.JTextField();
subircalif = new javax.swing.JDialog();
jLabel3 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
AceptarCalif = new javax.swing.JButton();
idalumno = new javax.swing.JTextField();
examen1 = new javax.swing.JTextField();
examen2 = new javax.swing.JTextField();
examen3 = new javax.swing.JTextField();
examenfinal = new javax.swing.JTextField();
jLabel9 = new javax.swing.JLabel();
materias = new javax.swing.JComboBox<>();
lblTituloInicial = new javax.swing.JLabel();
btnImagen1 = new javax.swing.JLabel();
Panel21 = new javax.swing.JPanel();
lblUsuario = new javax.swing.JLabel();
lblContraseña = new javax.swing.JLabel();
```

```

btnUsuario = new javax.swing.JLabel();
btnContraseña = new javax.swing.JLabel();
txtUsuario = new javax.swing.JTextField();
btnFin = new javax.swing.JButton();
bttnIniciarSesion = new javax.swing.JButton();
txtContraseña = new javax.swing.JPasswordField();

dglAlumno.setTitle("Ventana de calificaciones del alumno");
dglAlumno.setBackground(new java.awt.Color(0, 204, 204));

btnVolverAlumno.setText("Volver");
btnVolverAlumno.addActionListener(new java.awt.event.
event.ActionListener() {
    public void actionPerformed(java.awt.event.
ActionEvent evt) {
        btnVolverAlumnoActionPerformed(evt);
    }
});
});

TablaAlumno.setModel(new javax.swing.table.DefaultTableModel(
new Object [][] {
},
new String [] {
    "Materia", "Parcial 1", "Parcial 2", "Parcial 3",
"Calificacion Final"
}
));
jScrollPane1.setViewportView(TablaAlumno);

lblTituloAlumno1.setFont(new java.awt.Font("Times New Roman",
0, 18)); // NOI18N
lblTituloAlumno1.setText("Calificaciones del alumno");

txtAceptarAlum.setText("Ver Calificaciones");
txtAceptarAlum.addActionListener(new java.awt.event.
ActionListener() {
    public void actionPerformed(java.awt.event.
ActionEvent evt) {
        txtAceptarAlumActionPerformed(evt);
    }
});
});

javax.swing.GroupLayout Panel1Layout = new javax.swing.
GroupLayout(Panel1);
Panel1.setLayout(Panel1Layout);
Panel1Layout.setParallelGroup(
    Panel1Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)

```

```

        .addGroup(Panel1Layout.createSequentialGroup()
            .addGroup(Panel1Layout.createParallelGroup(javax.
swing.GroupLayout.Alignment.LEADING)
                .addGroup(Panel1Layout.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(jScrollPane1, javax.swing.
GroupLayout.PREFERRED_SIZE, 536, javax.swing.GroupLayout.
PREFERRED_SIZE))
                .addGroup(Panel1Layout.createSequentialGroup()
                    .addGap(147, 147, 147)
                    .addComponent(txtAceptarAlum)
                    .addGap(68, 68, 68)
                    .addComponent(btnVolverAlumno)))
            .addContainerGap(javax.swing.GroupLayout.
DEFAULT_SIZE, Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
Panel1Layout.createSequentialGroup()
            .addGap(0, 0, Short.MAX_VALUE)
            .addComponent(lblTituloAlumno1, javax.swing.
GroupLayout.PREFERRED_SIZE, 231, javax.swing.GroupLayout.
PREFERRED_SIZE)
            .addGap(141, 141, 141))
    );
    Panel1Layout.setVerticalGroup(
        Panel1Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.
TRAILING, Panel1Layout.createSequentialGroup()
                .addContainerGap(18, Short.MAX_VALUE)
                .addComponent(lblTituloAlumno1, javax.
swing.GroupLayout.PREFERRED_SIZE, 30, javax.swing.GroupLayout.
PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.UNRELATED)
                .addComponent(jScrollPane1, javax.swing.
GroupLayout.PREFERRED_SIZE, 115, javax.swing.GroupLayout.
PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED)
                .addGroup(Panel1Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(btnVolverAlumno)
                    .addComponent(txtAceptarAlum))
                .addContainerGap())
    );
}

javax.swing.GroupLayout dglAlumnoLayout =
new javax.swing.GroupLayout(dglAlumno.getContentPane());
dglAlumno.getContentPane().setLayout(dglAlumnoLayout);

```

```

        dglAlumnoLayout.setHorizontalGroup(
            dglAlumnoLayout.createParallelGroup(
(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(dglAlumnoLayout.createSequentialGroup()
                .addContainerGap()
                .addComponent(Panel1, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(18, Short.MAX_VALUE))
        );
        dglAlumnoLayout.setVerticalGroup(
            dglAlumnoLayout.createParallelGroup(
(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(dglAlumnoLayout.
createSequentialGroup()
                .addComponent(Panel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(40, Short.MAX_VALUE))
        );

        dglModificarGrupo.setTitle("Ventana para modificar
la tabla");

        lblTituloModificar.setFont(new java.awt.Font
("Times New Roman", 0, 18)); // NOI18N
        lblTituloModificar.setText("Modifica tu tabla");

        bttAgregarAlumno.setText("Aregar Alumno");
        bttAgregarAlumno.addActionListener
(new java.awt.event.ActionListener() {
            public void actionPerformed
(java.awt.event.ActionEvent evt) {
                bttAgregarAlumnoActionPerformed(evt);
            }
        });
        });

        jButton2.setText("Aregar Usuarios");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });
        });

        mnbpPrograma.setText("Inicio");

        btnmnSalir2.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/eliminar.png"))); // NOI18N
        btnmnSalir2.setText("Salir");
    
```

```

        btnmnSalir2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnmnSalir2ActionPerformed(evt);
    }
});
mnbPrograma.add(btnmnSalir2);

mnbMenu2.add(mnbPrograma);

dglModificarGrupo.setJMenuBar(mnbMenu2);

javax.swing.GroupLayout dglModificarGrupoLayout =
new javax.swing.GroupLayout(dglModificarGrupo.getContentPane());
dglModificarGrupo.getContentPane().setLayout
(dglModificarGrupoLayout);
dglModificarGrupoLayout.setHorizontalGroup(
dglModificarGrupoLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(dglModificarGrupoLayout.
createSequentialGroup()
.addContainerGap(82, Short.MAX_VALUE)
.addComponent(bttAgregarAlumno, javax.
swing.GroupLayout.PREFERRED_SIZE, 139, javax.swing.
GroupLayout.PREFERRED_SIZE)
.addGap(18, 18, 18)
.addComponent(jButton2, javax.swing.
GroupLayout.PREFERRED_SIZE, 144, javax.swing.GroupLayout.
PREFERRED_SIZE)
.addGap(92, 92, 92))
.addGroup(dglModificarGrupoLayout.
createSequentialGroup()
.addGap(158, 158, 158)
.addComponent(lblTituloModificar,
javax.swing.GroupLayout.PREFERRED_SIZE, 163, javax.
swing.GroupLayout.PREFERRED_SIZE)
.addContainerGap(javax.swing.GroupLayout.
DEFAULT_SIZE, Short.MAX_VALUE))
);
dglModificarGrupoLayout.setVerticalGroup(
dglModificarGrupoLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(dglModificarGrupoLayout.
createSequentialGroup()
.addGap(20, 20, 20)
.addComponent(lblTituloModificar)
.addGap(88, 88, 88)
.addGroup(dglModificarGrupoLayout.

```

```

createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(bttAgregarAlumno)
    .addComponent(jButton2)
    .addContainerGap(161, Short.MAX_VALUE))
);

dglNuevoAlumno.setTitle("Registrar nuevo alumno");

lblTituloAlumno.setFont(new java.awt.Font
("Times New Roman", 0, 18)); // NOI18N
lblTituloAlumno.setText("Agrega un nuevo alumno");

lblNombre.setText("Nombre:");

lblApellidoP.setText("Apellido Paterno:");

lblApellidoM.setText("Apellido Materno:");

lblGrupo.setText("Grupo:");

txtNombre.addKeyListener
(new java.awt.event.KeyAdapter() {
    public void keyTyped
(java.awt.event.KeyEvent evt) {
        txtNombreKeyTyped(evt);
    }
});

txtApellidoP.addKeyListener
(new java.awt.event.KeyAdapter() {
    public void keyTyped
(java.awt.event.KeyEvent evt) {
        txtApellidoPKeyTyped(evt);
    }
});

txtApellidoM.addKeyListener
(new java.awt.event.KeyAdapter() {
    public void keyTyped
(java.awt.event.KeyEvent evt) {
        txtApellidoMKeyTyped(evt);
    }
});

btnAceptarNuevo.setText("Aceptar");
btnAceptarNuevo.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {

```

```

        btnAceptarNuevoActionPerformed(evt);
    }
});

listGrupo.setModel
(new javax.swing.DefaultComboBoxModel<>
(new String[] { "1A", "1B", "2A", "2B", "3A", "3B" }));
listGrupo.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        listGrupoActionPerformed(evt);
    }
});
};

jLabel1.setText("Contraseña");

txtContraseña1.addKeyListener
(new java.awt.event.KeyAdapter() {
    public void keyTyped
(java.awt.event.KeyEvent evt) {
        txtContraseña1KeyTyped(evt);
    }
});
};

javax.swing.GroupLayout dglNuevoAlumnoLayout =
new javax.swing.GroupLayout(dglNuevoAlumno.
getContentPane());
dglNuevoAlumno.getContentPane().setLayout
(dglNuevoAlumnoLayout);
dglNuevoAlumnoLayout.setHorizontalGroup(
dglNuevoAlumnoLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglNuevoAlumnoLayout.
createSequentialGroup()
    .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(dglNuevoAlumnoLayout.
createSequentialGroup()
            .addGap(41, 41, 41)
            .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(lblApellidoP)
                .addComponent(lblApellidoM)
                .addComponent(lblGrupo)
                .addComponent(jLabel1)
                .addComponent(lblNombre,
javax.swing.GroupLayout.PREFERRED_SIZE, 81, javax.swing.

```

```

        GroupLayout.PREFERRED_SIZE))
                .addGap(44, 44, 44)
                .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
                .addComponent(txtNombre)
                .addComponent(txtApellidoP)
                .addComponent(txtApellidoM)
                .addComponent(listGrupo, 0,
138, Short.MAX_VALUE)
                .addComponent(txtContraseña1)))
            .addGroup(dglNuevoAlumnoLayout.
createSequentialGroup()
                .addGap(132, 132, 132)
                .addComponent(btnAceptarNuevo))
            .addGroup(dglNuevoAlumnoLayout.
createSequentialGroup()
                .addGap(84, 84, 84)
                .addComponent(lblTituloAlumno)))
            .addContainerGap(53, Short.MAX_VALUE))
        );
dglNuevoAlumnoLayout.setVerticalGroup(
    dglNuevoAlumnoLayout.createParallelGroup(
javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglNuevoAlumnoLayout.
createSequentialGroup()
    .addContainerGap()
    .addComponent(lblTituloAlumno, javax.swing.
GroupLayout.PREFERRED_SIZE, 39, javax.swing.GroupLayout.
PREFERRED_SIZE)
    .addGap(18, 18, 18)
    .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(lblNombre)
    .addComponent(txtNombre, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.UNRELATED)
    .addGroup(dglNuevoAlumnoLayout.createParallelGroup(
javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(lblApellidoP)
    .addComponent(txtApellidoP,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addPreferredGap(javax.swing.
LayoutManager.ComponentPlacement.UNRELATED)
    .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(lblApellidoM)
        .addComponent(txtApellidoM,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.
PREFERRED_SIZE))
        .addPreferredGap(javax.swing.
LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)
        .addComponent(jLabel1)
        .addComponent(txtContraseña1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.
swing.GroupLayout.DEFAULT_SIZE, javax.swing.
GroupLayout.PREFERRED_SIZE))
        .addGap(28, 28, 28)
        .addGroup(dglNuevoAlumnoLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(listGrupo,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.
swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.
PREFERRED_SIZE)
        .addComponent(lblGrupo))
        .addGap(55, 55, 55)
        .addComponent(btnAceptarNuevo)
        .addContainerGap(40, Short.MAX_VALUE))
    );
}

dglMaterias1.setTitle("Ventana de las materias");

btnMatematicas.setText("Matematicas1");
btnMatematicas.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnMatematicasActionPerformed(evt);
    }
});

btnEspañol.setText("Español1");
btnEspañol.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnEspañolActionPerformed(evt);
    }
});

btnCiencias.setText("Ciencias1");

```

```

        btnCiencias.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnCienciasActionPerformed(evt);
    }
});

lblMatematicas.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/matematicas (2).png"))); // NOI18N

lblEspañol.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/español.png"))); // NOI18N

lblCiencias.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/ciencias.png"))); // NOI18N

btnGeoografia.setText("Geoografia");
btnGeoagrafia.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnGeoagrafiaActionPerformed(evt);
    }
});

btnHistoria.setText("Historia");
btnHistoria.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnHistoriaActionPerformed(evt);
    }
});

lblHistoria.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/historia.png"))); // NOI18N

lblGeografia.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/geografia.png"))); // NOI18N

javax.swing.GroupLayout Panel3Layout =
new javax.swing.GroupLayout(Panel3);
Panel3.setLayout(Panel3Layout);
Panel3Layout.setHorizontalGroup(
    Panel3Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(Panel3Layout.createSequentialGroup()
            .addContainerGap()
            .addGroup(Panel3Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(Panel3Layout.createSequentialGroup()
                    .addGap(108, 108, 108)
                    .addGroup(Panel3Layout.createParallelGroup(
                        javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(btnCiencias)
                        .addComponent(lblMatematicas)
                        .addComponent(lblEspañol)
                        .addComponent(lblCiencias)
                        .addComponent(btnGeoagrafia)
                        .addComponent(btnHistoria)
                        .addComponent(lblHistoria)
                        .addComponent(lblGeografia))
                    .addGap(108, 108, 108))
                    .addGroup(Panel3Layout.createParallelGroup(
                        javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(btnGeografia)
                        .addComponent(lblGeografia))
                    .addGap(108, 108, 108)))
            .addContainerGap())
);

```

```

        .addGroup(Panel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(lblHistoria,
            javax.swing.GroupLayout.PREFERRED_SIZE, 63, javax.
            swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(btnHistoria))
        .addGap(15, 15, 15)
        .addGroup(Panel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(Panel3Layout.createSequentialGroup()
                .addGap(79, 79, 79)
                .addComponent(lblGeografia))
            .addGroup(Panel3Layout.createSequentialGroup()
                .addGap(68, 68, 68)
                .addComponent(btnGeoagrafia)))
        .addContainerGap())
    .addGroup(Panel3Layout.createSequentialGroup()
        .addGroup(Panel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(Panel3Layout.createSequentialGroup()
                .addGroup(Panel3Layout.createSequentialGroup()
                    .addComponent(btnMatematicas,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 116, javax.swing.
                    GroupLayout.PREFERRED_SIZE)
                    .addGroup(Panel3Layout.createSequentialGroup()
                        .addGroup(Panel3Layout.createSequentialGroup()
                            .addComponent(lblMatematicas,
                            javax.swing.GroupLayout.PREFERRED_SIZE, 63, javax.swing.
                            GroupLayout.PREFERRED_SIZE)))
                    .addGroup(Panel3Layout.createParallelGroup(
                        javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(Panel3Layout.createSequentialGroup()
                            .addGroup(Panel3Layout.createSequentialGroup()
                                .addComponent(btnEspanol, javax.swing.
                                GroupLayout.PREFERRED_SIZE, 91, javax.swing.GroupLayout.PREFERRED_SIZE))
                            .addGroup(Panel3Layout.createSequentialGroup()
                                .addComponent(lblEspanol)))
                        .addGroup(Panel3Layout.createParallelGroup(
                            javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(Panel3Layout.createSequentialGroup()
                                .addGroup(Panel3Layout.createSequentialGroup()
                                    .addComponent(btnCiencias, javax.
                                    swing.GroupLayout.PREFERRED_SIZE, 101, Short.MAX_VALUE))
                                .addGroup(Panel3Layout.createSequentialGroup()
                                    .addComponent(javax.swing.GroupLayout.Alignment.
                                    TRAILING, Panel3Layout.createSequentialGroup()
                                        .addPreferredGap(javax.swing.LayoutStyle.
                                        ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE,
                                        Short.MAX_VALUE)

```

```

        .addComponent(lblCiencias)
        .addGap(22, 22, 22)))
    );
    Panel3Layout.setVerticalGroup(
        Panel3Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
        .addGroup(Panel3Layout.createSequentialGroup()
            .addGroup(Panel3Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(Panel3Layout.createSequentialGroup()
                    .addComponent(lblCiencias)
                    .addGap(12, 12, 12)
                    .addComponent(btnCiencias,
javax.swing.GroupLayout.PREFERRED_SIZE, 75, javax.
swing.GroupLayout.PREFERRED_SIZE)
                    .addPreferredGap(javax.swing.
LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(btnCiencias))
                .addGroup(Panel3Layout.
createSequentialGroup()
                    .addGroup(Panel3Layout.
createParallelGroup(
                        javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(Panel3Layout.
createSequentialGroup()
                                .addComponent(lblMatematicas)
                                .addComponent(lblEspanol,
javax.swing.GroupLayout.PREFERRED_SIZE, 56, javax.swing.
.GroupLayout.PREFERRED_SIZE))
                                .addPreferredGap(javax.swing.
LayoutStyle.ComponentPlacement.RELATED)
                                .addGroup(Panel3Layout.
createParallelGroup(
                                    javax.swing.GroupLayout.Alignment.BASELINE)
                                    .addComponent(btnMatematicas)
                                    .addComponent(btnEspanol)))
                                .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED, 53, Short.MAX_VALUE)
                                .addGroup(Panel3Layout.createParallelGroup(
                                    javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(lblHistoria, javax.swing.
GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.
PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                                    .addComponent(lblGeografia, javax.
swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.
PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE))
                                .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(lblGeografia,
javax.swing.GroupLayout.PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(btnGeografia)))
                                .addGap(18, 18, 18)
                                .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(btnGeografia,
javax.swing.GroupLayout.PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(lblGeografia)))
                    .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(btnGeografia,
javax.swing.GroupLayout.PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(lblGeografia)))
                .addGroup(Panel3Layout.
createSequentialGroup())
                    .addComponent(btnGeografia)
                    .addComponent(lblGeografia)))
            .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(btnGeografia,
javax.swing.GroupLayout.PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(lblGeografia)))
        .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(btnGeografia,
javax.swing.GroupLayout.PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(lblGeografia)))
    .addGroup(Panel3Layout.
createParallelGroup(
                            javax.swing.GroupLayout.Alignment.BASELINE)
                            .addComponent(btnGeografia,
javax.swing.GroupLayout.PREFERRED_SIZE, 72, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(lblGeografia)))
);
}

```

```

Alignment.BASELINE)
        .addComponent(btnGeoografia)
        .addComponent(btnHistoria))
        .addContainerGap())
);

VentanaMaterias1.setText("Ventana");

btmmnSalirMaterias1.setIcon(new javax.
swing.ImageIcon(getClass().getResource
("/Imagen/eliminar.png"))); // NOI18N
btmmnSalirMaterias1.setText("Salir");
btmmnSalirMaterias1.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btmmnSalirMaterias1ActionPerformed(evt);
    }
});
VentanaMaterias1.add(btmmnSalirMaterias1);

mnbMenu4.add(VentanaMaterias1);

dglMaterias1.setJMenuBar(mnbMenu4);

javax.swing.GroupLayout
dglMaterias1Layout = new javax.swing.
GroupLayout(dglMaterias1.getContentPane());
dglMaterias1.getContentPane().
setLayout(dglMaterias1Layout);
dglMaterias1Layout.setHorizontalGroup(
    dglMaterias1Layout.
createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
    .addGroup(dglMaterias1Layout.
createSequentialGroup()
    .addGap(0, 0, Short.MAX_VALUE)
    .addGap(0, 0, Short.MAX_VALUE)
    .addGap(0, 0, Short.MAX_VALUE)
    .addGap(0, 0, Short.MAX_VALUE)
));
dglMaterias1Layout.setVerticalGroup(
    dglMaterias1Layout.
createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
    .addGroup(dglMaterias1Layout.
createSequentialGroup())
);

```

```

        .addComponent(Panel3,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.
swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap())
);

dglMaterias2.setTitle
("Ventana de las Materias");

btnMatematicas2.setText
("Matematicas2");
btnMatematicas2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnMatematicas2ActionPerformed(evt);
    }
});

btnEspañol2.setText("Español2");
btnEspañol2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnEspañol2ActionPerformed(evt);
    }
});

btnCiencias2.setText("Ciencias2");
btnCiencias2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnCiencias2ActionPerformed(evt);
    }
});

btnIngles1.setText("Ingles1");
btnIngles1.addActionListener
(new java.awt.event.ActionListener() {
}  public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnIngles1ActionPerformed(evt);
    }
});

btnCivicayEtica.setText("Civica y Etica");
btnCivicayEtica.addActionListener
(new java.awt.event.ActionListener() {

```

```

        public void actionPerformed
(java.awt.event.ActionEvent evt) {
            btnCivicayEticaActionPerformed(evt);
        }
    });

    lblMatematicas2.setIcon
(new javax.swing.ImageIcon(getClass().
getResource("/Imagen/matematicas (3).png"))); // NOI18N

    lblIngles1.setIcon
(new javax.swing.ImageIcon(getClass().
getResource("/Imagen/idioma-en-ingles.png"))); // NOI18N

    lblCivicayEtica.setIcon
(new javax.swing.ImageIcon(getClass().
getResource("/Imagen/moral (1).png"))); // NOI18N

    lblEspañol2.setIcon
(new javax.swing.ImageIcon(getClass().
getResource("/Imagen/libro-magico.png"))); // NOI18N

    lblCiencias2.setIcon
(new javax.swing.ImageIcon(getClass().
getResource("/Imagen/terapia.png"))); // NOI18N

    javax.swing.GroupLayout Panel4Layout
= new javax.swing.GroupLayout(Panel4);
    Panel4.setLayout(Panel4Layout);
    Panel4Layout.setHorizontalGroup(
        Panel4Layout.createParallelGroup()
            .addGroup(Panel4Layout.createSequentialGroup()
                .addGap(0, 0, 0)
                .addGroup(Panel4Layout.createParallelGroup()
                    .addComponent(lblMatematicas2)
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addGroup(Panel4Layout.createParallelGroup()
                            .addComponent(btnMatematicas2)))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblIngles1))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblCivicayEtica))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblEspañol2))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblCiencias2)))
                .addGap(0, 0, 0)
            )
        .addGap(0, 0, 0)
    );
    Panel4Layout.setVerticalGroup(
        Panel4Layout.createParallelGroup()
            .addGroup(Panel4Layout.createSequentialGroup()
                .addGap(0, 0, 0)
                .addGroup(Panel4Layout.createParallelGroup()
                    .addComponent(lblMatematicas2)
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addGroup(Panel4Layout.createParallelGroup()
                            .addComponent(btnMatematicas2)))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblIngles1))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblCivicayEtica))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblEspañol2))
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGap(0, 0, 0)
                        .addComponent(lblCiencias2)))
                .addGap(0, 0, 0)
            )
        .addGap(0, 0, 0)
    );
}

```

```

createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(lblEspañol2,
    javax.swing.GroupLayout.PREFERRED_SIZE, 66, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btnEspañol2))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 53, Short.MAX_VALUE)
            .addGroup(Panel4Layout.createParallelGroup(
    javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(btnCiencias2)
                .addComponent(lblCiencias2))
            .addGroupGap(35, 35, 35))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
    Panel4Layout.createSequentialGroup()
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(Panel4Layout.createParallelGroup(
    javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(btnIngles1)
                .addComponent(lblIngles1))
            .addGroupPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addGroup(Panel4Layout.createParallelGroup(
    javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addGroup(Panel4Layout.createSequentialGroup()
                            .addComponent(lblCivicayEtica)
                            .addGroupGap(105, 105, 105))
                        .addGroup(Panel4Layout.createSequentialGroup()
                            .addComponent(btnCivicayEtica)
                            .addGroupGap(85, 85, 85)))
                    .addContainerGap())))
    );
    Panel4Layout.setVerticalGroup(
        Panel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(Panel4Layout.createSequentialGroup()
                .addGroupGap(20, 20, 20)
                .addGroup(Panel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                    .addGroup(Panel4Layout.createSequentialGroup()
                        .addComponent(lblEspañol2,
    javax.swing.GroupLayout.PREFERRED_SIZE, 76, javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(lblMatematicas2,
    javax.swing.GroupLayout.PREFERRED_SIZE, 76, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(10, 10, 10)
                        .addComponent(btnMatematicas2,
    javax.swing.GroupLayout.PREFERRED_SIZE, 76, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addContainerGap())))
            .addGroup(Panel4Layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(btnMatematicas2,
    javax.swing.GroupLayout.PREFERRED_SIZE, 76, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap())))
    );

```

```

        GroupLayout.PREFERRED_SIZE))
                .addComponent(lblCiencias2))
                .addGap(18, 18, 18)
                .addGroup(Panel4Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(Panel4Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(btnEspañol2)
                    .addGroup(Panel4Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(btnCiencias2)))
                    .addComponent(btnMatematicas2))
                    .addGap(32, 32, 32)
                    .addGroup(Panel4Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(lblIngles1, javax.swing.
GroupLayout.PREFERRED_SIZE, 60, Short.MAX_VALUE)
                    .addComponent(lblCivicayEtica, javax.
swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
                    .addGap(18, 18, 18)
                    .addGroup(Panel4Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(btnIngles1)
                    .addComponent(btnCivicayEtica))
                    .addContainerGap())
);
;

VentanaMaterias2.setText("Ventana");

bttnmnSalirMaterias2.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/eliminar.png"))); // NOI18N
bttnmnSalirMaterias2.setText("Salir");
bttnmnSalirMaterias2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttnmnSalirMaterias2ActionPerformed(evt);
    }
});
VentanaMaterias2.add(bttnmnSalirMaterias2);

mnbMenu5.add(VentanaMaterias2);

dglMaterias2.setJMenuBar(mnbMenu5);

javax.swing.GroupLayout dglMaterias2Layout
= new javax.swing.GroupLayout(dglMaterias2.getContentPane());
dglMaterias2.getContentPane().setLayout
(dglMaterias2Layout);

```

```

dglMaterias2Layout.setHorizontalGroup(
    dglMaterias2Layout.createParallelGroup(
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(javax.swing.GroupLayout.
Alignment.TRAILING, dglMaterias2Layout.createSequentialGroup()
    .addContainerGap()
    .addComponent(Panel4, javax.swing.
GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.
DEFAULT_SIZE, Short.MAX_VALUE)
    .addContainerGap())
);
dglMaterias2Layout.setVerticalGroup(
    dglMaterias2Layout.createParallelGroup(
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglMaterias2Layout.createSequentialGroup()
    .addComponent(Panel4, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(0, 22, Short.MAX_VALUE))
);

dglMaterias3.setTitle("Ventana de las Materias");

btnMatematicas3.setText("Matematicas3");
btnMatematicas3.addActionListener(new java.awt.event.
event.ActionListener() {
    public void actionPerformed(java.awt.event.
ActionEvent evt) {
        btnMatematicas3ActionPerformed(evt);
    }
});

btnIngles2.setText("Ingles2");
btnIngles2.addActionListener(new java.awt.event.
.event.ActionListener() {
    public void actionPerformed(java.awt.event.
ActionEvent evt) {
        btnIngles2ActionPerformed(evt);
    }
});

btnTecnologias.setText("Tecnologias");
btnTecnologias.addActionListener(new java.awt.event.
event.ActionListener() {
    public void actionPerformed(java.awt.event.
.ActionEvent evt) {
        btnTecnologiasActionPerformed(evt);
    }
});

```

```

        btnQuimica.setText("Quimica");
        btnQuimica.addActionListener(new java.awt.event.
ActionListener() {
            public void actionPerformed(java.awt.event.
ActionEvent evt) {
                btnQuimicaActionPerformed(evt);
            }
        });

        btnVidasaludable.setText("Vida saludable");
        btnVidasaludable.addActionListener(new java.awt.
event.ActionListener() {
            public void actionPerformed(java.awt.event.
ActionEvent evt) {
                btnVidasaludableActionPerformed(evt);
            }
        });

        lblIngles2.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/idioma.png"))); // NOI18N

        lblMatematicas3.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/matematicas (1).png"))); // NOI18N

        lblTecnologias.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/tecnologia.png"))); // NOI18N

        lblVidasaludable.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/estilo-de-vida-saludable.png"))); // NOI18N

        lblQuimica.setIcon(new javax.swing.ImageIcon
(getClass().getResource("/Imagen/productos-quimicos (1).png"))); // NOI18N

        javax.swing.GroupLayout Panel5Layout =
new javax.swing.GroupLayout(Panel5);
        Panel5.setLayout(Panel5Layout);
        Panel5Layout.setHorizontalGroup(
            Panel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(Panel5Layout.createSequentialGroup()
                    .addGap(91, 91, 91)
                    .addComponent(lblVidasaludable))
                .addGroup(Panel5Layout.createSequentialGroup()
                    .addGap(72, 72, 72)
                    .addComponent(btnVidasaludable)))
        );
    }
}

```

```

        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED, javax.swing.GroupLayout.
DEFAULT_SIZE, Short.MAX_VALUE)
        .addGroup(Panel5Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(lblQuimica,
javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(btnQuimica,
javax.swing.GroupLayout.Alignment.TRAILING))
        .addGap(88, 88, 88)
        .addGroup(Panel5Layout.createSequentialGroup()
        .addGroup(Panel5Layout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(Panel5Layout.createSequentialGroup()
                .addGap(58, 58, 58)
                .addComponent(lblMatematicas3))
            .addGroup(Panel5Layout.createSequentialGroup()
                .addGap(26, 26, 26)
                .addComponent(btnMatematicas3,
javax.swing.GroupLayout.PREFERRED_SIZE, 116, javax.swing.
GroupLayout.PREFERRED_SIZE)))
        .addGap(31, 31, 31)
        .addGroup(Panel5Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(lblIngles2)
            .addComponent(btnIngles2))
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED, 28, Short.MAX_VALUE)
        .addGroup(Panel5Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(lblTecnologias)
            .addComponent(btnTecnologias))
        .addGap(40, 40, 40))
    );
    Panel5Layout.setVerticalGroup(
    Panel5Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(Panel5Layout.createSequentialGroup()
            .addGap(20, 20, 20)
            .addGroup(Panel5Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.TRAILING)
                .addGroup(Panel5Layout.createSequentialGroup()
                    .addGap(78, 78, 78)
                    .addComponent(lblMatematicas3,
javax.swing.GroupLayout.PREFERRED_SIZE, 78, javax.swing.
GroupLayout.PREFERRED_SIZE)
                    .addComponent(lblIngles2)
                    .addComponent(lblTecnologias)))

```



```

        GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(Panel5, javax.swing.
        GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap())
        );
        dglMaterias3Layout.setVerticalGroup(
                dglMaterias3Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(dglMaterias3Layout.
        createSequentialGroup()
                .addComponent(Panel5, javax.swing.
        GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 1, Short.MAX_VALUE))
        );
        dglSecretaria.setTitle("Ventana Secretaria");

        txtTituloSecretaria.setFont(new java.awt.Font
        ("Times New Roman", 1, 18)); // NOI18N
        txtTituloSecretaria.setText("ADMINISTRACIÓN");

        bttModificarGrupo.setText("Aregar");
        bttModificarGrupo.addActionListener
        (new java.awt.event.ActionListener() {
            public void actionPerformed
        (java.awt.event.ActionEvent evt) {
            bttModificarGrupoActionPerformed(evt);
        }
    });
    });

    mnPrograma4.setText("Inicio");

    btnmnSalirSecretaria.setIcon
    (new javax.swing.ImageIcon(getClass().getResource
    ("/Imagen/eliminar.png"))); // NOI18N
    btnmnSalirSecretaria.setText("Salir");
    btnmnSalirSecretaria.addActionListener
    (new java.awt.event.ActionListener() {
        public void actionPerformed
    (java.awt.event.ActionEvent evt) {
        btnmnSalirSecretariaActionPerformed(evt);
    }
});
    mnPrograma4.add(btnmnSalirSecretaria);

    mnbMenu12.add(mnPrograma4);

```

```

dglSecretaria.setJMenuBar(mnbMenu12);

javax.swing.GroupLayout dglSecretariaLayout
= new javax.swing.GroupLayout(dglSecretaria.getContentPane());
dglSecretaria.getContentPane().setLayout
(dglSecretariaLayout);
dglSecretariaLayout.setHorizontalGroup
(dglSecretariaLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(dglSecretariaLayout.createSequentialGroup()
.addGroup(dglSecretariaLayout.createParallelGroup()
.addGroup(dglSecretariaLayout.createSequentialGroup()
.addGap(105, 105, 105)
.addComponent(txtTituloSecretaria,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addGroup(dglSecretariaLayout.createSequentialGroup()
.addGap(168, 168, 168)
.addComponent(bttModificarGrupo)))
.addContainerGap(124, Short.MAX_VALUE))
);
dglSecretariaLayout.setVerticalGroup
(dglSecretariaLayout.createSequentialGroup()
.addGroup(dglSecretariaLayout.createSequentialGroup()
.addGap(68, 68, 68)
.addComponent(txtTituloSecretaria,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addGap(55, 55, 55)
.addComponent(bttModificarGrupo)
.addContainerGap(105, Short.MAX_VALUE))
);
dglCalificaciones.setTitle("Calificaciones ");

btnActualizarVidasaludable.setText
("Subir calificaciones");
btnActualizarVidasaludable.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnActualizarVidasaludableActionPerformed(evt);
    }
});

```

```

    CalificacionesMaestro.setModel
(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        },
        new String [] {
            "ID Calificaciones", "ID Alumno",
            "ID Materia", "Examen1", "Examen2", "Examen3",
            "Examen Final"
        }
    ) {
}   boolean[] canEdit = new boolean [] {
} false, false, false,
true, true, true, false
};

} public boolean isCellEditable
(int rowIndex, int columnIndex) {
}   return canEdit [columnIndex];
} }

));
jScrollPane2.setViewportView(CalificacionesMaestro);

bttVol1.setText("Volver");
bttVol1.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttVol1ActionPerformed(evt);
    }
});

bttVol2.setText("Volver");
bttVol2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttVol2ActionPerformed(evt);
    }
});

bttVol3.setText("Volver");
bttVol3.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttVol3ActionPerformed(evt);
    }
});

```

```

    });

    lblTitulo.setFont(new java.awt.Font
("Times New Roman", 0, 18)); // NOI18N

    jLabel2.setText("Filtrar por:");

    materiafiltro.setModel(new javax.swing.
DefaultComboBoxModel<>(new String[] { "Matemáticas 1",
"Español 1", "Ciencias 1", "Historia 1", "Geografía",
"Matemáticas2", "Español 2", "Ciencias 2", "Ingles 1",
"Cívica y Ética", "Matemáticas 3", "Ingles 2",
"Tecnologías", "Vida Saludable", "Química" }));

    grupofiltro.setModel(new javax.swing.
DefaultComboBoxModel<>(new String[] { "1A", "1B",
"2A", "2B", "3A", "3B" }));

    jButton1.setText("Acepatar");
    jButton1.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
dglCalificaciones.setJMenuBar(mnbMenu20);

javax.swing.GroupLayout dglCalificacionesLayout
= new javax.swing.GroupLayout(dglCalificaciones.getContentPane());
dglCalificaciones.getContentPane().setLayout
(dglCalificacionesLayout);
dglCalificacionesLayout.setHorizontalGroup(
    dglCalificacionesLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglCalificacionesLayout.
createSequentialGroup()
        .addGap(88, 88, 88)
        .addComponent(btnActualizarVidasaludable)
        .addGap(388, 388, 388)
        .addComponent(bttVol1)
        .addGap(30, 30, 30)
        .addComponent(bttVol2)
        .addGap(34, 34, 34)
        .addComponent(bttVol3)
        .addGap(Short.MAX_VALUE))
    .addGroup(dglCalificacionesLayout.
createSequentialGroup()
        .addGap(173

```

```
.addGroup(dglCalificacionesLayout.  
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(dglCalificacionesLayout.  
createSequentialGroup()  
        .addGap(39, 39, 39)  
        .addComponent(jLabel12)  
        .addGap(20, 20, 20)  
        .addComponent(materiafiltro,  
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)  
        .addGap(18, 18, 18)  
        .addComponent(grupofiltro,  
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)  
        .addGap(25, 25, 25)  
        .addGroup(dglCalificacionesLayout.  
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
            .addComponent(jButton1)  
            .addComponent(lblTitulo,  
javax.swing.GroupLayout.PREFERRED_SIZE, 293, javax.swing.GroupLayout.PREFERRED_SIZE))  
            .addGroup(dglCalificacionesLayout.  
createSequentialGroup()  
                .addGap(52, 52, 52)  
                .addComponent(jScrollPane2,  
javax.swing.GroupLayout.PREFERRED_SIZE, 889, javax.swing.GroupLayout.PREFERRED_SIZE))  
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))  
        );  
        dglCalificacionesLayout.setVerticalGroup(  
        dglCalificacionesLayout.  
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
            .addGroup(dglCalificacionesLayout.  
createSequentialGroup()  
                .addGroup(dglCalificacionesLayout.  
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
                    .addGroup(dglCalificacionesLayout.  
createSequentialGroup()  
                        .addGap(34, 34, 34)  
                        .addComponent(lblTitulo,  
javax.swing.GroupLayout.PREFERRED_SIZE, 35, javax.swing.GroupLayout.PREFERRED_SIZE)  
                        .addGap(18, 18, 18)
```

```

        .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 183, javax.
swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(dglCalificacionesLayout.
createSequentialGroup()
        .addContainerGap()
        .addGroup(dglCalificacionesLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
            .addComponent(jLabel2)
            .addComponent(materiafiltro,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.
PREFERRED_SIZE)
            .addComponent(grupofiltro,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jButton1)))
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED, 39, Short.MAX_VALUE)
        .addGroup(dglCalificacionesLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.
Alignment.TRAILING, dglCalificacionesLayout.createSequentialGroup()
                .addGroup(dglCalificacionesLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(bttVol1)
                    .addComponent(bttVol2)
                    .addComponent(bttVol3))
                .addContainerGap())
            .addGroup(javax.swing.GroupLayout.
Alignment.TRAILING, dglCalificacionesLayout.createSequentialGroup()
                .addComponent(btnActualizarVidasaludable)
                .addGap(21, 21, 21))))
    );
}

dglMaestro1.setTitle
("Ventana Maestro/Profesor");

lblTituloMaestro1.setFont
(new java.awt.Font("Arial", 0, 18)); // NOI18N
lblTituloMaestro1.setText
("Bienvenido a tu perfil ");

bttSubirCalificaciónM1.setText
("Subir Calificación");
bttSubirCalificaciónM1.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed

```

```

(java.awt.event.ActionEvent evt) {
    bttSubirCalificaciónM1ActionPerformed(evt);
}
});

mnPrograma1.setText("Inicio");

bttnmnSalirMaestro1.setIcon
(new javax.swing.ImageIcon(getClass().getResource
("/Imagen/eliminar.png"))); // NOI18N
bttnmnSalirMaestro1.setText("Salir");
bttnmnSalirMaestro1.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttnmnSalirMaestro1ActionPerformed(evt);
    }
});
mnPrograma1.add(bttnmnSalirMaestro1);

mnbMenu1.add(mnPrograma1);

dglMaestro1.setJMenuBar(mnbMenu1);

javax.swing.GroupLayout dglMaestro1Layout
= new javax.swing.GroupLayout(dglMaestro1.
getContentPane());
dglMaestro1.getContentPane().setLayout
(dglMaestro1Layout);
dglMaestro1Layout.setHorizontalGroup(
dglMaestro1Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglMaestro1Layout.
createSequentialGroup()
    .addGroup(dglMaestro1Layout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(dglMaestro1Layout.
createSequentialGroup()
            .addGap(104, 104, 104)
            .addComponent(lblTituloMaestro1))
        .addGroup(dglMaestro1Layout.
createSequentialGroup()
            .addGap(128, 128, 128)
            .addComponent(bttSubir
CalificaciónM1)))
    .addGap(123, Short.MAX_VALUE))
);
dglMaestro1Layout.setVerticalGroup(
dglMaestro1Layout.createParallelGroup

```

```

(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglMaestro1Layout.
createSequentialGroup()
    .addGap(39, 39, 39)
    .addComponent(lblTituloMaestro1,
javax.swing.GroupLayout.PREFERRED_SIZE, 51, javax.
swing.GroupLayout.PREFERRED_SIZE)
    .addGap(18, 18, 18)
    .addComponent(bttSubirCalificaciónM1)
    .addContainerGap(138, Short.MAX_VALUE))
);

dglMaestro2.setTitle("Ventana Maestro/Profesor");

lblTituloMaestro2.setFont(new java.awt.Font
("Arial", 0, 18)); // NOI18N
lblTituloMaestro2.setText
("Bienvenido a tu perfil ");

bttSubirCalificaciónM2.setText
("Subir Calificación");
bttSubirCalificaciónM2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttSubirCalificaciónM2ActionPerformed(evt);
    }
});

mnPrograma2.setText("Inicio");

bttnmnSalirMaestro2.setIcon(new javax.swing.
ImageIcon(getClass().getResource
("/Imagen/eliminar.png"))); // NOI18N
bttnmnSalirMaestro2.setText("Salir");
bttnmnSalirMaestro2.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        bttnmnSalirMaestro2ActionPerformed(evt);
    }
});
mnPrograma2.add(bttnmnSalirMaestro2);

mnbMenu22.add(mnPrograma2);

dglMaestro2.setJMenuBar(mnbMenu22);

javax.swing.GroupLayout dglMaestro2Layout

```

```

= new javax.swing.GroupLayout(dglMaestro2.getContentPane());
    dglMaestro2.getContentPane().setLayout(dglMaestro2Layout);
    dglMaestro2Layout.setHorizontalGroup(
        dglMaestro2Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(dglMaestro2Layout.createSequentialGroup()
                .addGap(104, 104, 104)
                .addComponent(lblTituloMaestro2)
                .addContainerGap(123, Short.MAX_VALUE))
            .addGroup(dglMaestro2Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(dglMaestro2Layout.createSequentialGroup()
                    .addGap(135, 135, 135)
                    .addComponent(bttSubirCalificaciónM2)
                    .addContainerGap(152, Short.MAX_VALUE)))
        );
    dglMaestro2Layout.setVerticalGroup(
        dglMaestro2Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(dglMaestro2Layout.createSequentialGroup()
                .addGap(39, 39, 39)
                .addComponent(lblTituloMaestro2, javax.swing.GroupLayout.PREFERRED_SIZE, 51, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(187, Short.MAX_VALUE))
            .addGroup(dglMaestro2Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(dglMaestro2Layout.createSequentialGroup()
                    .addGap(123, 123, 123)
                    .addComponent(bttSubirCalificaciónM2)
                    .addContainerGap(131, Short.MAX_VALUE)))
        );
    dglMaestro3.setTitle("Ventana Maestro/Profesor");
    lblTituloMaestro3.setFont(new java.awt.Font
    ("Arial", 0, 18)); // NOI18N
    lblTituloMaestro3.setText("Bienvenido a tu perfil ");

    bttSubirCalificaciónM3.setText("Subir Calificacion");
    bttSubirCalificaciónM3.addActionListener
    (new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            bttSubirCalificaciónM3ActionPerformed(evt);
        }
    });
    mnPrograma3.setText("Inicio");

    btnmnSalirMaestro3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Imagen/eliminar.png"))); // NOI18N

```

```

btmmnSalirMaestro3.setText("Salir");
btmmnSalirMaestro3.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btmmnSalirMaestro3ActionPerformed(evt);
    }
});
mnPrograma3.add(btmmnSalirMaestro3);

mnbMenu23.add(mnPrograma3);

dglMaestro3.setJMenuBar(mnbMenu23);

javax.swing.GroupLayout dglMaestro3Layout =
new javax.swing.GroupLayout(dglMaestro3.getContentPane());
dglMaestro3.getContentPane().setLayout(dglMaestro3Layout);
dglMaestro3Layout.setHorizontalGroup(
    dglMaestro3Layout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(dglMaestro3Layout.createSequentialGroup()
        .addGap(104, 104, 104)
        .addComponent
(lblTituloMaestro3))
    .addGroup(dglMaestro3Layout.createSequentialGroup()
        .addGap(123, 123, 123)
        .addComponent(bttSubirCalificaciónM3)))
    .addGroup(dglMaestro3Layout.createParallelGroup()
        .addGroup(dglMaestro3Layout.createSequentialGroup()
            .addGap(39, 39, 39)
            .addComponent(lblTituloMaestro3,
javax.swing.GroupLayout.PREFERRED_SIZE, 51, javax.
swing.GroupLayout.PREFERRED_SIZE)
            .addGap(27, 27, 27)
            .addComponent(bttSubirCalificaciónM3)
            .addGap(133, Short.MAX_VALUE)))
);
dglMaestro3Layout.setVerticalGroup(
    dglMaestro3Layout.createSequentialGroup()
        .addGroup(dglMaestro3Layout.createParallelGroup()
            .addGroup(dglMaestro3Layout.createSequentialGroup()
                .addGap(123, 123, 123)
                .addComponent(bttSubirCalificaciónM3)))
            .addGroup(dglMaestro3Layout.createSequentialGroup()
                .addGap(39, 39, 39)
                .addComponent(lblTituloMaestro3,
javax.swing.GroupLayout.PREFERRED_SIZE, 51, javax.
swing.GroupLayout.PREFERRED_SIZE)
                .addGap(27, 27, 27)
                .addComponent(bttSubirCalificaciónM3)
                .addGap(133, Short.MAX_VALUE)))
));
ApellidoPUsu.addKeyListener

```

```

(new java.awt.event.KeyAdapter() {
    public void keyTyped
(java.awt.event.KeyEvent evt) {
        ApellidoPUsuKeyTyped(evt);
    }
});

ApellidoMUsu.addKeyListener
(new java.awt.event.KeyAdapter() {
    public void keyTyped
(java.awt.event.KeyEvent evt) {
        ApellidoMUsuKeyTyped(evt);
    }
});

btnAceptarNuevo1.setText("Aceptar");
btnAceptarNuevo1.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        btnAceptarNuevo1ActionPerformed(evt);
    }
});

listRol.setModel(new javax.swing.
DefaultComboBoxModel<>(new String[] { "2PRO",
"2PRO2", "2PRO3", "3SEC" }));
listRol.addActionListener(new java
.awt.event.ActionListener() {
    public void actionPerformed(java.
awt.event.ActionEvent evt) {
        listRolActionPerformed(evt);
    }
});

jLabel4.setText("Contraseña");

lblTituloAlumno2.setFont(new java.awt.
Font("Times New Roman", 0, 18)); // NOI18N
lblTituloAlumno2.setText("Aregar Usuario");

ContraseñaUsu.addKeyListener(new java.
awt.event.KeyAdapter() {
    public void keyTyped(java.awt.
event.KeyEvent evt) {
        ContraseñaUsuKeyTyped(evt);
    }
});

```

```

lblNombre1.setText("Nombre:");

lblApellidoP1.setText("Apellido Paterno:");

lblApellidoM1.setText("Apellido Materno:");

lblGrupo1.setText("Rol:");

NombreUsu.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        NombreUsuKeyTyped(evt);
    }
});

javax.swing.GroupLayout CrearUsuarioLayout
= new javax.swing.GroupLayout(CrearUsuario.getContentPane());
CrearUsuario.getContentPane().setLayout
(CrearUsuarioLayout);
CrearUsuarioLayout.setHorizontalGroup(
    CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(CrearUsuarioLayout.
createSequentialGroup()
    .addGroup(CrearUsuarioLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
        .addGroup(CrearUsuarioLayout.
createSequentialGroup()
            .addGroup(CrearUsuarioLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                .addGroup(CrearUsuarioLayout.
createSequentialGroup()
                    .addGroup(CrearUsuarioLayout.
.createSequentialGroup()
                        .addGroup(CrearUsuarioLayout.
.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                            .addGroup(CrearUsuarioLayout.
.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                                .addComponent(lblApellidoP1)
                                .addComponent(lblApellidoM1)
                                .addComponent(lblGrupo1)
                                .addComponent(jLabel4)
                                .addComponent(lblNombre1,
javax.swing.GroupLayout.PREFERRED_SIZE, 81, javax.swing.
GroupLayout.PREFERRED_SIZE))
                            .addGroup(CrearUsuarioLayout.
.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                                .addComponent(btnAceptarNuevo1)
                                .addGroup(CrearUsuarioLayout.
.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
                                    .addComponent(NombreUsu)
                                    .addComponent(ApellidoPUsu)
                                    .addComponent(ApellidoMUsu)
                                    .addComponent(listRol, 0,

```

```

138, Short.MAX_VALUE)
        .addComponent(ContraseñaUsu)))
    .addGroup(CrearUsuarioLayout.
createSequentialGroup()
        .addGap(118, 118, 118)
        .addComponent(lblTituloAlumno2)))
    .addContainerGap(82, Short.MAX_VALUE))
);
CrearUsuarioLayout.setVerticalGroup(
    CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(CrearUsuarioLayout.
createSequentialGroup()
        .addContainerGap()
        .addComponent(lblTituloAlumno2,
javax.swing.GroupLayout.PREFERRED_SIZE, 39, javax.swing.
GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addGroup(CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(lblNombre1)
        .addComponent(NombreUsu, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.UNRELATED)
        .addGroup(CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(lblApellidoP1)
        .addComponent(ApellidoPUsu,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.UNRELATED)
        .addGroup(CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(lblApellidoM1)
        .addComponent(ApellidoMUsu, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED)
        .addGroup(CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(jLabel4)
        .addComponent(ContraseñaUsu, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(20, 20, 20)

```

```

        .addGroup(CrearUsuarioLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(lblGrupo1)
            .addComponent(listRol, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(27, 27, 27)
        .addComponent(btnAceptarNuevo1)
        .addContainerGap(39, Short.MAX_VALUE))
    );

jLabel3.setText("ID Alumno:");

jLabel5.setText("Examen 1:");

jLabel6.setText("Examen 2:");

jLabel7.setText("Examen 3:");

jLabel8.setText("Examen Final:");

AceptarCalif.setText("Aceptatar");
AceptarCalif.addActionListener
(new java.awt.event.ActionListener() {
    public void actionPerformed
(java.awt.event.ActionEvent evt) {
        AceptarCalifActionPerformed(evt);
    }
});

examen1.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        examen1KeyTyped(evt);
    }
});

examen2.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        examen2KeyTyped(evt);
    }
});

examen3.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        examen3KeyTyped(evt);
    }
});

examenfinal.addKeyListener(new java.awt.event.KeyAdapter() {

```

```

        public void keyTyped(java.awt.event.KeyEvent evt) {
            examenfinalKeyTyped(evt);
        }
    });

    jLabel9.setText("Materia:");

    materias.setModel(new javax.swing.DefaultComboBoxModel
<>(new String[] { "Matemáticas 1", "Español 1", "Ciencias 1",
"Historia 1", "Geografía", "Matemáticas2", "Español 2",
"Ciencias 2", "Ingles 1", "Cívica y Ética", "Matemáticas 3",
"Ingles 2", "Tecnologías", "Vida Saludable", "Química" }));
    materias.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            materiasActionPerformed(evt);
        }
    });
}

javax.swing.GroupLayout subircalifLayout =
new javax.swing.GroupLayout(subircalif.getContentPane());
subircalif.getContentPane().setLayout(subircalifLayout);
subircalifLayout.setHorizontalGroup(
    subircalifLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(subircalifLayout.createSequentialGroup()
            .addGap(107, 107, 107)
            .addComponent(AceptarCalif))
        .addGroup(subircalifLayout.createSequentialGroup()
            .addGap(45, 45, 45)
            .addGroup(subircalifLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(subircalifLayout.createSequentialGroup()
                    .addGap(107, 107, 107)
                    .addComponent(jLabel8)
                    .addGap(18, 18, 18)
                    .addComponent(examenfinal))
                .addGroup(subircalifLayout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(examen3)))
            .addGroup(subircalifLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel7)
                .addGroup(subircalifLayout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel6)
                    .addGap(18, 18, 18)
                    .addComponent(examen4))))));
}

```

```

        .addComponent(examen2))
    .addGroup(subircalifLayout.

createSequentialGroup()
        .addComponent(jLabel5)
        .addGap(18, 18, 18)
        .addComponent(examen1))
    .addGroup(subircalifLayout.

createSequentialGroup()
        .addGroup(subircalifLayout.
createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel3)
            .addComponent(jLabel9))
        .addGap(18, 18, 18)
        .addGroup(subircalifLayout.

createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(idalumno)
            .addComponent(materias, 0,
145, Short.MAX_VALUE))))))
        .addContainerGap(68, Short.MAX_VALUE))
    );
    subircalifLayout.setVerticalGroup(
        subircalifLayout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
        .addGroup(subircalifLayout.createSequentialGroup()
            .addGap(22, 22, 22)
            .addGroup(subircalifLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel3)
                .addComponent(idalumno, javax.swing.
GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(8, 8, 8)
            .addGroup(subircalifLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel9)
                .addComponent(materias, javax.swing.GroupLayout.
PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.
GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.UNRELATED)
            .addGroup(subircalifLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel5)
                .addComponent(examen1, javax.swing.GroupLayout.
PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.
GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)
            .addGroup(subircalifLayout.createParallelGroup
(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(jLabel6)
        .addComponent(examen2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addGroup(subircalifLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel7)
            .addComponent(examen3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addGroup(subircalifLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel8)
                .addComponent(examenfinal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(AceptarCalif)
                .addContainerGap())
            );
    });

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setTitle("Ventana principal (acceder)");
    setBackground(new java.awt.Color(0, 255, 204));

    lblTituloInicial.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
    lblTituloInicial.setText("Ingresa");

    btnImagen1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Imagen/contraseña16px.png"))); // NOI18N

    lblUsuario.setText("Usuario:");

    lblContraseña.setText("Contraseña:");

    btnUsuario.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Imagen/usuario.png"))); // NOI18N

    btnContraseña.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Imagen/candado-aberto.png"))); // NOI18N

    txtUsuario.setBorder(null);
    txtUsuario.addKeyListener(new java.awt.event.KeyAdapter() {
        public void keyTyped(java.awt.event.KeyEvent evt) {

```

```

        txtUsuarioKeyTyped(evt);
    }
});

btnFin.setIcon(new javax.swing.ImageIcon(getClass().
getResource("/Imagen/eliminar.png"))); // NOI18N
btnFin.setText("Terminar programa");
btnFin.addActionListener(new java.awt.event.
ActionListener() {
    public void actionPerformed(java.awt.event.
ActionEvent evt) {
        btnFinActionPerformed(evt);
    }
});
};

bttIniciarSesion.setText("Iniciar Sesion");
bttIniciarSesion.addActionListener(new java.awt.event.
ActionListener() {
    public void actionPerformed(java.awt.event.
ActionEvent evt) {
        bttIniciarSesionActionPerformed(evt);
    }
});
};

txtContraseña.addKeyListener(new java.awt.event.
KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent
evt) {
        txtContraseñaKeyTyped(evt);
    }
});
};

javax.swing.GroupLayout Panel21Layout = new javax.
swing.GroupLayout(Panel21);
Panel21.setLayout(Panel21Layout);
Panel21Layout.setHorizontalGroup(
    Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
    .addGroup(Panel21Layout.createSequentialGroup()
        .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.TRAILING)
            .addComponent(btnContraseña)
            .addComponent(btnUsuario))
        .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
            .addGroup(Panel21Layout.createSequentialGroup()
                .addGap(18, 18, 18)
                .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                    .addGroup(Panel21Layout.createSequentialGroup()
                        .addGap(18, 18, 18)
                        .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                            .addGroup(Panel21Layout.createSequentialGroup()
                                .addGap(18, 18, 18)
                                .addComponent(Panel21Layout.
createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                        .addComponent(Panel21Layout.
createSequentialGroup()
                            .addGap(18, 18, 18)
                            .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                .addComponent(Panel21Layout.
createSequentialGroup()
                                    .addGap(18, 18, 18)
                                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                        .addComponent(Panel21Layout.
createSequentialGroup())
                                            .addGap(18, 18, 18)
                                            .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                .addComponent(Panel21Layout.
createSequentialGroup())
                                                    .addGap(18, 18, 18)
                                                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                        .addComponent(Panel21Layout.
createSequentialGroup())
                                                            .addGap(18, 18, 18)
                                                            .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                                .addComponent(Panel21Layout.
createSequentialGroup())
                                                                    .addGap(18, 18, 18)
                                                                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                                        .addComponent(Panel21Layout.
createSequentialGroup())
                                                                            .addGap(18, 18, 18)
                                                                            .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                                                .addComponent(Panel21Layout.
createSequentialGroup())
                                                                                    .addGap(18, 18, 18)
                                                                                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                                                        .addComponent(Panel21Layout.
createSequentialGroup())
                                                                                            .addGap(18, 18, 18)
                                                                                            .addGroup(Panel21Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
                                                                                                .addComponent(Panel21Layout.
createSequentialGroup())
                                                                

```

```

        .addComponent(btnFin)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    .addGroup(Panel21Layout.createSequentialGroup()
        .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(lblUsuario,
                javax.swing.GroupLayout.PREFERRED_SIZE, 56, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(lblContraseña,
                javax.swing.GroupLayout.PREFERRED_SIZE, 104, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(txtUsuario, javax.swing.GroupLayout.DEFAULT_SIZE, 116, Short.MAX_VALUE)
        .addComponent(txtContraseña))
        .addGap(46, 46, 46)))
    .addGroup(Panel21Layout.createSequentialGroup()
        .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(txtUsuario, javax.swing.GroupLayout.DEFAULT_SIZE, 116, Short.MAX_VALUE)
            .addComponent(txtContraseña))
        .addGroup(Panel21Layout.createSequentialGroup()
            .addGap(126, 126, 126)
            .addComponent(bttIniciarSesion)
            .addGap(0, 0, Short.MAX_VALUE)))
        .addContainerGap()))
);
Panel21Layout.setVerticalGroup(
    Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(Panel21Layout.createSequentialGroup()
            .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(Panel21Layout.createSequentialGroup()
                    .addGap(17, 17, 17)
                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(Panel21Layout.createSequentialGroup()
                            .addGap(17, 17, 17)
                            .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addGroup(Panel21Layout.createSequentialGroup()
                                    .addGap(17, 17, 17)
                                    .addComponent(lblUsuario,
                                        javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                    .addComponent(txtUsuario,
                                        javax.swing.GroupLayout.PREFERRED_SIZE, 20, javax.swing.GroupLayout.PREFERRED_SIZE))
                                .addGroup(Panel21Layout.createSequentialGroup()
                                    .addGap(37, 37, 37)
                                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                                        .addComponent(lblContraseña,
                                            javax.swing.GroupLayout.PREFERRED_SIZE, 20, javax.swing.GroupLayout.PREFERRED_SIZE)
                                        .addComponent(txtContraseña))))))
                .addGroup(Panel21Layout.createSequentialGroup()
                    .addGap(37, 37, 37)
                    .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(lblContraseña,
                            javax.swing.GroupLayout.PREFERRED_SIZE, 20, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(txtContraseña))))))
            .addContainerGap())
        .addGroup(Panel21Layout.createSequentialGroup()
            .addGap(37, 37, 37)
            .addGroup(Panel21Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(lblContraseña,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 20, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(txtContraseña))))))
);

```

```

        .addComponent(txtContraseña,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(Panel21Layout.createSequentialGroup()
        .addComponent(btnUsuario)
        .addPreferredGap(javax.swing.
LayoutStyle.ComponentPlacement.RELATED, javax.swing.
GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(btnContraseña)))
        .addGap(27, 27, 27)
        .addComponent(bttIniciarSesion)
        .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.UNRELATED)
        .addComponent(btnFin)
        .addContainerGap())
    );
}

javax.swing.GroupLayout layout = new javax.swing.
GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
    .addGap(159, 159, 159)
    .addComponent(lblTituloInicial, javax.swing.
GroupLayout.PREFERRED_SIZE, 82, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED)
    .addComponent(btnImagen1)
    .addGap(0, 177, Short.MAX_VALUE))
    .addGroup(javax.swing.GroupLayout.Alignment.
TRAILING, layout.createSequentialGroup()
    .addContainerGap(javax.swing.GroupLayout.
DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(Panel21, javax.swing.GroupLayout.
PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.
GroupLayout.PREFERRED_SIZE)
    .addGap(36, 36, 36))
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
    .addGap(19, 19, 19)
    .addGroup(layout.createParallelGroup(
(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(lblTituloInicial, javax.
swing.GroupLayout.PREFERRED_SIZE, 25, javax.swing.GroupLayout.

```

```

PREFERRED_SIZE)
    .addComponent(btnImagen1, javax.swing.
    GroupLayout.PREFERRED_SIZE, 25, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addPreferredGap(javax.swing.LayoutStyle.
ComponentPlacement.RELATED)
    .addComponent(Panel21, javax.swing.
    GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)
    .addContainerGap(javax.swing.GroupLayout.
    DEFAULT_SIZE, Short.MAX_VALUE))
);

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

private void btnmnSalirMaestro1ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaestro1.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
}

private void btnmnSalir2ActionPerformed(java.awt.
event.ActionEvent evt) {
    dglModificarGrupo.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
    dglSecretaria.setBounds(470, 180, 400, 300);
    dglSecretaria.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    dglSecretaria.setVisible(true);
}

private void btnAceptarNuevoActionPerformed
(java.awt.event.ActionEvent evt) {
    AlumnoNuevo();
}

private void btnVolverAlumnoActionPerformed
(java.awt.event.ActionEvent evt) {
    dglAlumno.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
}

private void btnMatematicasActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(true);
    bttVol2.setVisible(false);
    bttVol3.setVisible(false);
    dglMaterias1.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
}

```

```

        dglCalificaciones.setBounds(470, 180, 951, 355);
        dglCalificaciones.setModal(true); //Este metodo
permite que no se pueda usar otra ventana mientras el JDialog este activo
        lblTitulo.setText("Matematicas 1");
        dglCalificaciones.setVisible(true);
    }

    private void btnEspañolActionPerformed(java.awt.event.
ActionEvent evt) {
    bttVol1.setVisible(true);
    bttVol2.setVisible(false);
    bttVol3.setVisible(false);
    dglMaterias1.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Español");
    dglCalificaciones.setVisible(true);
}

private void btnCienciasActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(true);
    bttVol2.setVisible(false);
    bttVol3.setVisible(false);
    dglMaterias1.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Ciencias");
    dglCalificaciones.setVisible(true);
}

private void btnHistoriaActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(true);
    bttVol2.setVisible(false);
    bttVol3.setVisible(false);
    dglMaterias1.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Historia");
    dglCalificaciones.setVisible(true);
}

```

```

private void btnGeoagrafiaActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(true);
    bttVol2.setVisible(false);
    bttVol3.setVisible(false);
    dglMaterias1.dispose(); //Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Geografia");
    dglCalificaciones.setVisible(true);
}

private void btnmnSalirMaterias1ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaterias1.dispose(); //Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
    dglMaestro1.setBounds(470, 180, 448, 268);
    dglMaestro1.setModal(true); //Este metodo permite que no
se pueda usar otra ventana mientras el JDialog este activo
    dglMaestro1.setVisible(true);
}

private void btnmnSalirMaterias2ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaterias2.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglMaestro2.setBounds(470, 180, 448, 268);
    dglMaestro2.setModal(true); //Este metodo permite que
no se pueda usar otra ventana mientras el JDialog este activo
    dglMaestro2.setVisible(true);
}

private void btnmnSalirMaterias3ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaterias3.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglMaestro3.setBounds(470, 180, 448, 268);
    dglMaestro3.setModal(true); //Este metodo permite que
no se pueda usar otra ventana mientras el JDialog este activo
    dglMaestro3.setVisible(true);
}

private void btnMatematicas3ActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(false);
    bttVol2.setVisible(false);
    bttVol3.setVisible(true);
}

```

```

        dglMaterias3.dispose(); //Hace invisible el JDialog pero
        la ventana no queda en memoria(recogedor de basura)
        dglCalificaciones.setBounds(470, 180, 951, 355);
        dglCalificaciones.setModal(true); //Este metodo permite
        que no se pueda usar otra ventana mientras el JDialog este activo
        lblTitulo.setText("Matematicas 3");
        dglCalificaciones.setVisible(true);
    }

    private void btnIngles2ActionPerformed
    (java.awt.event.ActionEvent evt) {
        bttVol1.setVisible(false);
        bttVol2.setVisible(false);
        bttVol3.setVisible(true);
        dglMaterias3.dispose(); //Hace invisible el JDialog pero
        la ventana no queda en memoria(recogedor de basura)
        dglCalificaciones.setBounds(470, 180, 968, 549);
        dglCalificaciones.setModal(true); //Este metodo permite
        que no se pueda usar otra ventana mientras el JDialog este activo
        lblTitulo.setText("Ingles 2");
        dglCalificaciones.setVisible(true);
    }

    private void btnTecnologiasActionPerformed
    (java.awt.event.ActionEvent evt) {
        bttVol1.setVisible(false);
        bttVol2.setVisible(false);
        bttVol3.setVisible(true);
        dglMaterias3.dispose(); //Hace invisible el JDialog pero
        la ventana no queda en memoria(recogedor de basura)
        dglCalificaciones.setBounds(470, 180, 951, 355);
        dglCalificaciones.setModal(true); //Este metodo permite
        que no se pueda usar otra ventana mientras el JDialog este activo
        lblTitulo.setText("Tecnologias");
        dglCalificaciones.setVisible(true);
    }

    private void btnVidasaludableActionPerformed
    (java.awt.event.ActionEvent evt) {
        bttVol1.setVisible(false);
        bttVol2.setVisible(false);
        bttVol3.setVisible(true);
        dglMaterias3.dispose(); //Hace invisible el JDialog
        pero la ventana no queda en memoria(recogedor de basura)
        dglCalificaciones.setBounds(470, 180, 951, 355);
        dglCalificaciones.setModal(true); //Este metodo permite
        que no se pueda usar otra ventana mientras el JDialog este activo
        lblTitulo.setText("Vida Saludable");
        dglCalificaciones.setVisible(true);
    }

```

```

}

private void btnQuimicaActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(false);
    bttVol2.setVisible(false);
    bttVol3.setVisible(true);
    dglMaterias3.dispose(); //Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo
permite que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Quimica");
    dglCalificaciones.setVisible(true);
}

private void btnMatematicas2ActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(false);
    bttVol2.setVisible(true);
    bttVol3.setVisible(false);
    dglMaterias2.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo
permite que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Matematicas 2");
    dglCalificaciones.setVisible(true);
}

private void btnEspañol2ActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(false);
    bttVol2.setVisible(true);
    bttVol3.setVisible(false);
    dglMaterias2.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Español 2");
    dglCalificaciones.setVisible(true);
}

private void btnCiencias2ActionPerformed
(java.awt.event.ActionEvent evt) {

    bttVol1.setVisible(false);
    bttVol2.setVisible(true);
}

```

```

        bttVol3.setVisible(false);
        dglMaterias2.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
        dglCalificaciones.setBounds(470, 180, 951, 355);
        dglCalificaciones.setModal(true); //Este metodo
permite que no se pueda usar otra ventana mientras el JDialog este activo
        lblTitulo.setText("Ciencias 2");
        dglCalificaciones.setVisible(true);
    }

    private void btnIngles1ActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(false);
    bttVol2.setVisible(true);
    bttVol3.setVisible(false);
    dglMaterias2.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo
permite que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Ingles 2");
    dglCalificaciones.setVisible(true);
}

private void btnCivicayEticaActionPerformed
(java.awt.event.ActionEvent evt) {
    bttVol1.setVisible(false);
    bttVol2.setVisible(true);
    bttVol3.setVisible(false);
    dglMaterias2.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
    dglCalificaciones.setBounds(470, 180, 951, 355);
    dglCalificaciones.setModal(true); //Este metodo
permite que no se pueda usar otra ventana mientras el JDialog este activo
    lblTitulo.setText("Civica y Etica");
    dglCalificaciones.setVisible(true);
}

private void consultar() {//////////TABLA///////////
    /*Object[] ob = new Object[7];
    try{
        Statement s= c.conectar().createStatement();
// pasa el nombre de la bd y crea el satatement
        // con.conectar();
        //Declaramos un objeto de tipo Resulset
        // PreparedStatement ps = con.conectar().
prepareStatement("insert into usuarios
(idu,idr,nombreusu,apellidousu,contraseñausu)values(?,?,?,?,?)");
}

```

```

        ResultSet rs= s.executeQuery("Select * from alumnos;");

        while (rs.next()){
            //    ob[0] = rs.getObject("ida");
            //    ob[1] = rs.getObject("idu");
            //    ob[2] = rs.getObject("idr");
            ob[3] = rs.getObject("nombre");// 
            //    ob[4] = rs.getObject("apellido");
            //    ob[5] = rs.getObject("contraseña");
            ob[6] = rs.getObject("idg");// 
            //dtm.addRow(ob);
            }

            c.conectar().close();
        }catch(Exception e){
            System.out.println(e.getMessage());
        }
        //Calificaciones.setModel(dtm);
// insertar el modelo de la tabla por medio del casteo.. codigo arriba
/*
}

private void btnmnSalirMaestro2ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaestro2.dispose();//Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
}

private void btnmnSalirMaestro3ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaestro3.dispose();//Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
}

private void btnmnSalirSecretariaActionPerformed
(java.awt.event.ActionEvent evt) {
    dglSecretaria.dispose();//Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
}

private void bttSubirCalificaciónM1ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaestro1.dispose();//Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
    dglMaterias1.setBounds(470, 180, 427, 326);
//Se le da el tamaño al JDialog con el que aparecera cuando se invoque
    dglMaterias1.setModal(true);
}

```

```

//Este metodo permite que no se pueda usar
otra ventana mientras el JDialog este activo
    dglMaterias1.setVisible(true);
//Hace visible el JDialog
}

private void bttSubirCalificaciónM2ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaestro2.dispose();//Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
    dglMaterias2.setBounds(470, 180, 427, 326);
//Se le da el tamaño al JDialog con el que aparecera
cuando se invoque
    dglMaterias2.setModal(true);//Este metodo
permite que no se pueda usar otra ventana mientras
el JDialog este activo
    dglMaterias2.setVisible(true);//Hace visible el JDialog
}

private void bttSubirCalificaciónM3ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglMaestro3.dispose();//Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
    dglMaterias3.setBounds(470, 180, 427, 326);
//Se le da el tamaño al JDialog con el que aparecera cuando se invoque
    dglMaterias3.setModal(true);//Este metodo permite que
no se pueda usar otra ventana mientras el JDialog este activo
    dglMaterias3.setVisible(true);//Hace visible el JDialog
}

private void bttModificarGrupoActionPerformed
(java.awt.event.ActionEvent evt) {
    dglSecretaria.dispose();//Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglModificarGrupo.setBounds(470, 180, 475, 335);
//Se le da el tamaño al JDialog con el que aparecera cuando se invoque
    dglModificarGrupo.setModal(true);//Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
    dglModificarGrupo.setVisible(true);//Hace visible el JDialog
}

private void bttAgregarAlumnoActionPerformed
(java.awt.event.ActionEvent evt) {
    dglModificarGrupo.dispose();//Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
    dglNuevoAlumno.setBounds(470, 180, 360, 344);//Se le da
el tamaño al JDialog con el que aparecera cuando se invoque
    dglNuevoAlumno.setModal(true);//Este metodo permite que
no se pueda usar otra ventana mientras el JDialog este activo

```

```

        dglNuevoAlumno.setVisible(true); //Hace visible el JDialog
    }

    private void jButton2ActionPerformed
(java.awt.event.ActionEvent evt) {
    dglModificarGrupo.dispose(); //Hace invisible el
JDialog pero la ventana no queda en memoria(recogedor de basura)
    CrearUsuario.setBounds(470, 180, 389, 307); //Se le da
el tamaño al JDialog con el que aparecera cuando se invoque
    CrearUsuario.setModal(true); //Este metodo permite que no
se pueda usar otra ventana mientras el JDialog este activo
    CrearUsuario.setVisible(true); //Hace visible el JDialog
}

private void bttIniciarSesionActionPerformed
(java.awt.event.ActionEvent evt) {
    Login();
}

private void btnFinActionPerformed
(java.awt.event.ActionEvent evt) {
    //Muestra un comentario de fin de programa
    JOptionPane.showMessageDialog
(null, "Eso ha sido todo, gracias por usar este programa");
    //Se cierra el programa
    System.exit(0);
}

private void btnAceptarNuevo1ActionPerformed
(java.awt.event.ActionEvent evt) {
    UsuarioNuevo();
}

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

}

private void AceptarCalifActionPerformed
(java.awt.event.ActionEvent evt) {
    Calificacion();
}

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

}

private void listRolActionPerformed

```

```

(java.awt.event.ActionEvent evt) {

}

    private void txtAceptarAlumActionPerformed
(java.awt.event.ActionEvent evt) {
    ConsultarCalificaciones();
}

    private void txtUsuarioKeyTyped
(java.awt.event.KeyEvent evt) {
    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    boolean espacio = key == 32;
    if (!(minusculas || mayusculas || espacio)) {
        evt.consume();
    }
}

    private void txtContraseñaKeyTyped
(java.awt.event.KeyEvent evt) {
//    int key = evt.getKeyChar();
//    boolean mayusculas = key >= 65 && key <= 90;
//    boolean minusculas = key >= 97 && key <= 122;
//    boolean espacio = key == 32;
//    if (!(minusculas || mayusculas || espacio)) {
//        evt.consume();
//    }
//    boolean numeros = key >= 48 && key <= 57;
//    if (!numeros) {
//        evt.consume();
//    }
//    if (txtContraseña.getText().trim().length() == 10) {
//        evt.consume();
//    }
}
}

private void examen1KeyTyped(java.awt.event.KeyEvent evt) {
//Comprobacion de datos que sean solo numeros

    int key = evt.getKeyChar();
    boolean numeros = key >= 48 && key <= 57;
    if (!numeros) {
        evt.consume();
    }
    if (examen1.getText().trim().length() == 10) {
        evt.consume();
    }
}

```

```

}

private void examen2KeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo numeros

    int key = evt.getKeyChar();
    boolean numeros = key >= 48 && key <= 57;
    if (!numeros) {
        evt.consume();
    }
    if (examen2.getText().trim().length() == 10) {
        evt.consume();
    }
}

private void examen3KeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo numeros

    int key = evt.getKeyChar();
    boolean numeros = key >= 48 && key <= 57;
    if (!numeros) {
        evt.consume();
    }
    if (examen3.getText().trim().length() == 10) {
        evt.consume();
    }
}

private void examenfinalKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo numeros

    int key = evt.getKeyChar();
    boolean numeros = key >= 48 && key <= 57;
    if (!numeros) {
        evt.consume();
    }
    if (examenfinal.getText().trim().length() == 10) {
        evt.consume();
    }
}

private void txtNombreKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras

    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    if (!(minusculas || mayusculas)) {
        evt.consume();
    }
}

```

```

        }
    }

private void txtApellidoPKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras

    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    if (!(minusculas || mayusculas)) {
        evt.consume();
    }
}

private void txtApellidoMKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras

    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    if (!(minusculas || mayusculas)) {
        evt.consume();
    }
}

private void txtContraseña1KeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras y numeros

    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    boolean numeros = key >= 48 && key <= 57;
    if (!(minusculas || mayusculas || numeros)) {
        evt.consume();
    }
}

private void NombreUsuKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras

    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    if (!(minusculas || mayusculas)) {
        evt.consume();
    }
}

private void ApellidoPUsuKeyTyped(java.awt.event.KeyEvent evt) {

```

```

//Comprobacion de datos que sean solo letras

int key = evt.getKeyChar();
boolean mayusculas = key >= 65 && key <= 90;
boolean minusculas = key >= 97 && key <= 122;
if (!(minusculas || mayusculas)) {
    evt.consume();
}
}

private void ApellidoMuertoKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras

    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    if (!(minusculas || mayusculas)) {
        evt.consume();
    }
}

private void ContraseñaUsuKeyTyped(java.awt.event.KeyEvent evt) {
    //Comprobacion de datos que sean solo letras y numeros
    int key = evt.getKeyChar();
    boolean mayusculas = key >= 65 && key <= 90;
    boolean minusculas = key >= 97 && key <= 122;
    boolean numeros = key >= 48 && key <= 57;
    if (!(minusculas || mayusculas || numeros)) {
        evt.consume();
    }
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String grupo, califID, materiaid = "";

    grupo = grupofiltro.getSelectedItem().toString();
    if (materiafiltro.getSelectedItem() == "Matemáticas 1")
    {//Identifica la materia seleccionada asignando a una variable el
    nombre del Id del campo
        materiaid = "mat1";
    }
    if (materiafiltro.getSelectedItem() == "Español 1")
    {//Identifica la materia seleccionada asignando a una variable el
    nombre del Id del campo
        materiaid = "esp1";
    }
    if (materiafiltro.getSelectedItem() == "Ciencias 1")
    {//Identifica la materia seleccionada asignando a una variable el
    nombre del Id del campo

```

```

        materiaid = "cien1";
    }
    if (materiafiltro.getSelectedItem() == "Historia 1")
//Identifica la materia seleccionada asignando a una variable el
nombre del Id del campo
        materiaid = "hist1";
    }
    if (materiafiltro.getSelectedItem() == "Geografía")
//Identifica la materia seleccionada asignando a una variable el
nombre del Id del campo
        materiaid = "geo1";
    }
    if (materiafiltro.getSelectedItem() == "Matemáticas 2")
//Identifica la materia seleccionada asignando a una variable el
nombre del Id del campo
        materiaid = "mate2";
    }
    if (materiafiltro.getSelectedItem() == "Español 2")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "esp2";
    }
    if (materiafiltro.getSelectedItem() == "Ciencias 2")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "cien2";
    }
    if (materiafiltro.getSelectedItem() == "Ingles 1")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "ingl1";
    }
    if (materiafiltro.getSelectedItem() == "Cívica y Ética")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "civyeti";
    }
    if (materiafiltro.getSelectedItem() == "Matemáticas 3")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "mate3";
    }
    if (materiafiltro.getSelectedItem() == "Ingles 2")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "ingl2";
    }
    if (materiafiltro.getSelectedItem() == "Tecnologías")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "tec1";
    }
    if (materiafiltro.getSelectedItem() == "Vida Saludable")
//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo
        materiaid = "vidsalu1";
    }

```

```

        if (materiafiltro.getSelectedItem() == "Química")
    {//Identifica la materia seleccionada asignando a una variable el nombre del Id del campo

        materiaid = "quimi1";
    }
//Se realiza la consulta para filtrar los datos seleccionados
    String consulta = "select * from materias as m, grupo as g,
calificaciones as c where "
        + "m.idm=c.idm and m.idg=g.idg and g.idg=" + grupo + ", "
        + "and c.idm=" + materiaid + "';";

    Object[] objeto = new Object[7];
//imprime en el cuadro los datos de calificaciones de los alumnos
    try {
        Statement s = con.conectaBase(bd).createStatement();
        ResultSet rs = s.executeQuery(consulta);
        while (rs.next()) {
            objeto[0] = rs.getObject("idc");
            objeto[1] = rs.getObject("ida");
            objeto[2] = rs.getObject("idm");
            objeto[3] = rs.getObject("parcial1");
            objeto[4] = rs.getObject("parcial2");
            objeto[5] = rs.getObject("parcial3");
            objeto[6] = rs.getObject("examenfinal");
            //modelo.addRow(ob);
        }
        modelo1.addRow(objeto);
        con.conectaBase(bd).close();//se cierra la conexión a la BD
    } catch (Exception e) {
        System.out.println(e.getMessage());
    }
    CalificacionesMaestro.setModel(modelo1);
}

private void bttVol3ActionPerformed(java.awt.event.ActionEvent evt) {
    dglCalificaciones.dispose();
//Hace invisible el JDialog pero la ventana no queda
en memoria(recogedor de basura)
    dglMaterias3.setBounds(470, 180, 430, 370);
    dglMaterias3.setModal(true);//Este metodo permite que
no se pueda usar otra ventana mientras el JDialog este activo
    dglMaterias3.setVisible(true);
}

private void bttVol2ActionPerformed(java.awt.event.ActionEvent evt) {
    dglCalificaciones.dispose();//Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
    dglMaterias2.setBounds(470, 180, 430, 370);
}

```

```

        dglMaterias2.setModal(true); //Este metodo permite que no se
pueda usar otra ventana mientras el JDialog este activo
        dglMaterias2.setVisible(true);
    }

    private void btVol1ActionPerformed(java.awt.event.ActionEvent evt) {
        dglCalificaciones.dispose(); //Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
        dglMaterias1.setBounds(470, 180, 430, 370);
        dglMaterias1.setModal(true); //Este metodo permite que no se
pueda usar otra ventana mientras el JDialog este activo
        dglMaterias1.setVisible(true);
    }

    private void btnActualizarVidasaludableActionPerformed
(java.awt.event.ActionEvent evt) {
        dglCalificaciones.dispose(); //Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
        subircalif.setBounds(470, 180, 329, 277);
        subircalif.setModal(true); //Este metodo permite que no se
pueda usar otra ventana mientras el JDialog este activo
        subircalif.setVisible(true);
    }

}

public void Calificacion() {
} //obtenemos los datos de las cajas seleccionadas de subir calificaiones
    String idalu = idalumno.getText();
    String ex1 = examen1.getText();
    String ex2 = examen2.getText();
    String ex3 = examen3.getText();
    String exfinal = examenfinal.getText();
    String califID = "";
    String materiaid = "";
//convertimos los datos a entero
    int exa1 = Integer.parseInt(ex1);
    int exa2 = Integer.parseInt(ex2);
    int exa3 = Integer.parseInt(ex3);
    int exafinal = Integer.parseInt(exfinal);

    if (materias.getSelectedItem() == "Matemáticas 1")
{//Identifica la materia seleccionada asignando a una variable
el nombre del Id del campo más el nombre
        califID = "Matemáticas 1";
        //        materiaid = materias.getSelectedItem().toString();
        materiaid = "mat1";
    }
    if (materias.getSelectedItem() == "Español 1")
}

```

```

{ //Identifica la materia seleccionada asignando a una variable
el nombre del Id del campo más el nombre
    califID = "Español 1";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "esp1";
}
if (materias.getSelectedItem() == "Ciencias 1")
{ //Identifica la materia seleccionada asignando a una variable
el nombre del Id del campo más el nombre
    califID = "Ciencias 1";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "cien1";
}
if (materias.getSelectedItem() == "Historia 1")
{ //Identifica la materia seleccionada asignando a una variable
el nombre del Id del campo más el nombre
    califID = "Historia 1";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "hist1";
}
if (materias.getSelectedItem() == "Geografía")
{ //Identifica la materia seleccionada asignando a una variable
el nombre del Id del campo más el nombre
    califID = "Geografía";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "geo1";
}
if (materias.getSelectedItem() == "Matemáticas 2")
{ //Identifica la materia seleccionada asignando a una variable el
nombre del Id del campo más el nombre
    califID = "Matemáticas 2";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "mate2";
}
if (materias.getSelectedItem() == "Español 2") { //Identifica
la materia seleccionada asignando a una variable el nombre del Id del
campo más el nombre
    califID = "Español 2";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "esp2";
}
if (materias.getSelectedItem() == "Ciencias 2") { //Identifica la
materia seleccionada asignando a una variable el nombre del Id del campo
más el nombre
    califID = "Ciencias 2";
    //        materiaid = materias.getSelectedItem().toString();
    materiaid = "cien2";
}
if (materias.getSelectedItem() == "Ingles 1") { //Identifica la

```

```

materia seleccionada asignando a una variable el nombre del Id del campo
más el nombre
    califID = "Ingles 1";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "ingl1";
}
if (materias.getSelectedItem() == "Cívica y Ética") {//Identifica
la materia seleccionada asignando a una variable el nombre del Id del
campo más el nombre
    califID = "Cívica y Ética";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "civyeti";
}
if (materias.getSelectedItem() == "Matemáticas 3") {//Identifica
la materia seleccionada asignando a una variable el nombre del Id del
campo más el nombre
    califID = "Matemáticas 3";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "mate3";
}
if (materias.getSelectedItem() == "Ingles 2") {//Identifica la
materia seleccionada asignando a una variable el nombre del Id del campo
más el nombre
    califID = "Ingles 2";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "ingl2";
}
if (materias.getSelectedItem() == "Tecnologías") {//Identifica
la materia seleccionada asignando a una variable el nombre del Id del
campo más el nombre
    califID = "Tecnologías";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "tec1";
}
if (materias.getSelectedItem() == "Vida Saludable") {//Identifica
la materia seleccionada asignando a una variable el nombre del Id del
campo más el nombre
    califID = "Vida Saludable";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "vidsalu1";
}
if (materias.getSelectedItem() == "Química") {//Identifica la
materia seleccionada asignando a una variable el nombre del Id del
campo más el nombre
    califID = "Química";
    //      materiaid = materias.getSelectedItem().toString();
    materiaid = "quimi1";
}

```

```

        int numero = 0;
        for (int i = 1; i <= 2; i++) {
            //ciclo para crear numeros de 0 a 999
            Random r = new Random();
            numero = r.nextInt(1000);
        }
        califID = califID + numero;
        try {
            //INSERTAMOS CALIFICACIONES
            PreparedStatement ps = con.conectaBase(bd).prepareStatement
("insert into calificaciones (idc,ida,idm,parcial1,parcial2,parcial3,examenfinal) "
+ "values(?,?,?,?,?,?);");
            ps.setString(1, califID);
            ps.setString(2, idalu);
            ps.setString(3, materiaid);
            ps.setInt(4, exa1);
            ps.setInt(5, exa2);
            ps.setInt(6, exa3);
            ps.setInt(7, exafinal);
            ps.executeUpdate();
            JOptionPane.showMessageDialog(null, "Datos insertados correctamente");
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }

    public void ConsultarCalificaciones() {
        //consultas para mostrar las calificaciones y el nombre de la materia
        String sql2 = "select nombremat from materias as m,
alumnos as a, calificaciones as c where "
+ "m.idm=c.idm and a.ida=c.ida and c.ida='"
+ txtUsuario.getText() + "';";
        String sql = "select parcial1,parcial2,parcial3,examenfinal
from calificaciones as c, "
+ "alumnos as a where c.ida=a.ida and c.ida='"
+ txtUsuario.getText() + "';

        Object[] ob = new Object[5];
        try {
            //nos conectamos a las base de datos
            Statement s = con.conectaBase(bd).createStatement();
            Statement s1 = con.conectaBase(bd).createStatement();
            ResultSet rs = s.executeQuery(sql);//se hace la consulta
            ResultSet r = s1.executeQuery(sql2);//se hace la consulta
            //imprime los datos
            while (r.next()) {
                ob[0] = r.getObject("nombremat");
            }
        }
    }
}

```

```

        while (rs.next()) {
            //ob[0] = r.getObject("nombremat");
            ob[1] = rs.getObject("parcial1");
            ob[2] = rs.getObject("parcial2");
            ob[3] = rs.getObject("parcial3");
            ob[4] = rs.getObject("examenfinal");
            //modelo.addRow(ob);
        }
        modelo.addRow(ob);
        con.conectaBase(bd).close();
    } catch (Exception e) {
        System.out.println(e.getMessage());
    }
    TablaAlumno.setModel(modelo);
}

public void UsuarioNuevo() {
    //se asignan los datos a variables que se obtuvieron de las cajas
    String apellido1, apellido2, nombre2;
    String nombre = NombreUsu.getText();
    String apellidop = ApellidoPUsu.getText();
    String apellidom = ApellidoMUsu.getText();
    String password = ContraseñaUsu.getText();
    String rol = listRol.getSelectedItem().toString();
    String idu1;
    String apellido = apellidop + " " + apellidom;
    apellido.toUpperCase();
    String NombreTabla = nombre;
    NombreTabla.toUpperCase();

    int numero = 0;
    nombre2 = nombre.trim(); //quitamos los espacios en blanco
    apellido1 = apellidop.trim(); //quitamos los espacios en blanco
    apellido2 = apellidom.trim(); //quitamos los espacios en blanco
    apellido1 = apellido1.substring(0, 2); //segunda letra del primer apellido
    apellido2 = apellido2.substring(0, 2); //segunda letra del segundo apellido
    nombre2 = nombre2.substring(0, 2); //segunda letra del nombre
    apellido1.toUpperCase(); //ponemos las letras en mayuscula
    apellido2.toUpperCase(); //ponemos las letras en mayuscula
    nombre2.toUpperCase(); //ponemos las letras en mayuscula
    for (int i = 1; i <= 2; i++) { //ciclo para crear numeros de 0 a 999
        Random r = new Random();
        numero = r.nextInt(1000);
    }
    String num = String.valueOf(numero);
    idu1 = rol + nombre2 + apellido1 + apellido2 + num; // creamos el idu
    idu1.toUpperCase();
    //INSERTAMOS DATOS
    try {

```

```

//con.conectaServer();
PreparedStatement ps = con.conectaBase(bd).prepareStatement
("insert into usuarios (idu,idr,nombreusu,apellidousu,contraseñausu)values(?,?,?,?,?,?)");
ps.setString(1, idu1);
ps.setString(2, listRol.getSelectedItem().toString());
ps.setString(3, NombreTabla);
ps.setString(4, apellido);
ps.setString(5, password);
ps.executeUpdate();
JOptionPane.showMessageDialog(null, "Datos insertados correctamente");
} catch (Exception ex) {
    System.out.println(ex);
}

CrearUsuario.dispose(); //Hace invisible el JDialog pero
la ventana no queda en memoria(recogedor de basura)
dglModificarGrupo.setBounds(470, 180, 430, 370); //Se le da
el tamaño al JDialog con el que aparecerá cuando se invoque
dglModificarGrupo.setModal(true); //Este método permite que
no se pueda usar otra ventana mientras el JDialog esté activo
dglModificarGrupo.setVisible(true); //Hace visible el JDialog
}

public void Login() {
    //con.conectaServer();
    int a = 0;
    int b = 0;
    String SQL1 = "select * from usuarios where idu= '" +
txtUsuario.getText() + "' and contraseñausu = '" +
txtContraseña.getText() + "';";
    String sql = "select idu from usuarios where idu='"
+ txtUsuario.getText() + "';";
    String IdentificadorUsuarioAlumno = "select * from
usuarios where idr='1ALU' and idu='" + txtUsuario.getText() +
"' and contraseñausu= '" + txtContraseña.getText() + "';";
    String IdentificadorUsuarioProfesor = "select * from
usuarios where idr='2PRO' and idu='" + txtUsuario.getText() +
"' and contraseñausu= '" + txtContraseña.getText() + "';";
    String IdentificadorUsuarioProfesor2 = "select * from
usuarios where idr='2PRO2' and idu='" + txtUsuario.getText() +
"' and contraseñausu= '" + txtContraseña.getText() + "';";
    String IdentificadorUsuarioProfesor3 = "select * from
usuarios where idr='2PR03' and idu='" + txtUsuario.getText() +
"' and contraseñausu= '" + txtContraseña.getText() + "';";
    String IdentificadorUsuarioSecretaria = "select * from
usuarios where idr='3SEC' and idu='" + txtUsuario.getText() +
"' and contraseñausu= '" + txtContraseña.getText() + "';";
    String p = null;
}

```

```

int s = 1;

try {
    if (con.ejecutarConsulta(bd, IdentificadorUsuarioAlumno).next()) {
        b = 1;
    }
    if (con.ejecutarConsulta(bd, IdentificadorUsuarioProfesor).next()) {
        b = 2;
    }
    if (con.ejecutarConsulta(bd, IdentificadorUsuarioProfesor2).next()) {
        b = 3;
    }
    if (con.ejecutarConsulta(bd, IdentificadorUsuarioProfesor3).next()) {
        b = 4;
    }
    if (con.ejecutarConsulta(bd, IdentificadorUsuarioSecretaria).next()) {
        b = 5;
    }
    switch (b) {
        case 1:
            if (b == 1) {
                dglAlumno.setBounds(470, 180, 584, 254);
                dglAlumno.setModal(true); //Este metodo
                permite que no se pueda usar otra ventana mientras el JDialog este activo
                dglAlumno.setVisible(true);
            }
            break;
        case 2:
            if (b == 2) {
                dglMaestro1.setBounds(470, 180, 400, 299);
                dglMaestro1.setModal(true); //Este metodo
                permite que no se pueda usar otra ventana mientras el JDialog este activo
                dglMaestro1.setVisible(true);
            }
            break;
        case 3:
            if (b == 3) {
                dglMaestro2.setBounds(470, 180, 400, 299);
                dglMaestro2.setModal(true); //Este metodo
                permite que no se pueda usar otra ventana mientras el JDialog este activo
                dglMaestro2.setVisible(true);
            }
            break;
        case 4:
            if (b == 4) {
                dglMaestro3.setBounds(470, 180, 400, 299);
                dglMaestro3.setModal(true); //Este metodo
                permite que no se pueda usar otra ventana mientras el JDialog este activo

```

```

        dglMaestro3.setVisible(true);
    }
    break;
case 5:
    if (b == 5) {
        dglSecretaria.setBounds(470, 180, 400, 300);
        dglSecretaria.setModal(true);
        dglSecretaria.setVisible(true);
    }
    break;

default:
    JOptionPane.showMessageDialog
(null, "Usuario o contraseña erronea");
    break;
}
} catch (Exception e) {
    System.out.println(e);
}
}

public void AlumnoNuevo() {
//con.conectaServer();
String apellido1, apellido2, nombre2;
String nombre = txtNombre.getText();
String apellidop = txtApellidoP.getText();
String apellidom = txtApellidoM.getText();
String password = txtContraseña1.getText();
String idu1;
String apellido = apellidop + " " + apellidom;
String rol = "1ALU";

int numero = 0;
nombre2 = nombre.trim(); //quitamos los espacios en blanco
apellido1 = apellidop.trim(); //quitamos los espacios en blanco
apellido2 = apellidom.trim(); //quitamos los espacios en blanco
apellido1.toUpperCase(); //ponemos las letras en mayuscula
apellido2.toUpperCase(); //ponemos las letras en mayuscula
nombre2.toUpperCase(); //ponemos las letras en mayuscula
apellido1 = apellido1.substring(0, 2); //segunda letra del primer apellido
apellido2 = apellido2.substring(0, 2); //segunda letra del segundo apellido
nombre2 = nombre2.substring(0, 2); //segunda letra del nombre
apellido1.toUpperCase(); //ponemos las letras en mayuscula
apellido2.toUpperCase(); //ponemos las letras en mayuscula
nombre2.toUpperCase(); //ponemos las letras en mayuscula
for (int i = 1; i <= 2; i++) { //ciclo para crear numeros de 0 a 999
    Random r = new Random();
    numero = r.nextInt(1000);
}
}

```

```

String num = String.valueOf(numero);
idu1 = "1ALU" + nombre2 + apellido1 + apellido2 + num;
String idu2 = idu1;
try {
    PreparedStatement ps = con.conectaBase(bd).
prepareStatement("insert into usuarios (idu,idr,nombreusu,
apellidousu,contraseñausu)values(?,?,?,?,?,?)");
    ps.setString(1, idu1);
    ps.setString(2, rol);
    ps.setString(3, txtNombre.getText());
    ps.setString(4, apellido);
    ps.setString(5, txtContraseña1.getText());

    ps.executeUpdate();
} catch (Exception ex) {
    System.out.println(ex);
}
try {
    PreparedStatement ps = con.conectaBase(bd).
prepareStatement("insert into alumnos (ida,idu,idr,nombrealu,
apellidoalu,contraseñaalu,idg)values(?,?,?,?,?,?,?)");
    ps.setString(1, idu1);
    ps.setString(2, idu1);
    ps.setString(3, rol);
    ps.setString(4, txtNombre.getText());
    ps.setString(5, apellido);
    ps.setString(6, txtContraseña1.getText());
    ps.setString(7, listGrupo.getSelectedItem().toString());
    ps.executeUpdate();
    JOptionPane.showMessageDialog(null, "Datos guardados correctamente");
} catch (Exception ex) {
    System.out.println(ex);
}
dglNuevoAlumno.dispose(); //Hace invisible el JDialog
pero la ventana no queda en memoria(recogedor de basura)
dglModificarGrupo.setBounds(470, 180, 430, 370); //Se
le da el tamaño al JDialog con el que aparecera cuando se invoque
dglModificarGrupo.setModal(true); //Este metodo permite
que no se pueda usar otra ventana mientras el JDialog este activo
dglModificarGrupo.setVisible(true); //Hace visible el JDialog
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    // <editor-fold defaultstate="collapsed" desc=" Look and

```

```

feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available,
    stay with the default look and feel.
     * For details see http://download.oracle.com/javase/
tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        UIManager.setLookAndFeel(new com.nilo.plaf.nimrod.
NimRODLookAndFeel());
    } catch (Exception ex) {
        java.util.logging.Logger.getLogger(Ventana_principal.
class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    @Override
    public void run() {
        new Ventana_principal().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JButton AceptarCalif;
private javax.swing.JTextField ApellidoMUsu;
private javax.swing.JTextField ApellidoPUsu;
private javax.swing.JTable CalificacionesMaestro;
private javax.swing.JTextField ContraseñaUsu;
private javax.swing.JDialog CrearUsuario;
private javax.swing.JTextField NombreUsu;
private javax.swing.JPanel Panel1;
private javax.swing.JPanel Panel21;
private javax.swing.JPanel Panel3;
private javax.swing.JPanel Panel4;
private javax.swing.JPanel Panel5;
private javax.swing.JTable TablaAlumno;
private javax.swing.JMenu VentanaMaterias1;
private javax.swing.JMenu VentanaMaterias2;
private javax.swing.JMenu VentanaMaterias3;
private javax.swing.JButton btnAceptarNuevo;
private javax.swing.JButton btnAceptarNuevo1;
private javax.swing.JButton btnActualizarVidasaludable;
private javax.swing.JButton btnCiencias;
private javax.swing.JButton btnCiencias2;
private javax.swing.JButton btnCivicayEtica;
private javax.swing.JLabel btnContraseña;
private javax.swing.JButton btnEspañol;

```

```
private javax.swing.JButton btnEspañol2;
private javax.swing.JButton btnFin;
private javax.swing.JButton btnGeoagrafia;
private javax.swing.JButton btnHistoria;
private javax.swing.JLabel btnImagen1;
private javax.swing.JButton btnIngles1;
private javax.swing.JButton btnIngles2;
private javax.swing.JButton btnMatematicas;
private javax.swing.JButton btnMatematicas2;
private javax.swing.JButton btnMatematicas3;
private javax.swing.JButton btnQuimica;
private javax.swing.JButton btnTecnologias;
private javax.swing.JLabel btnUsuario;
private javax.swing.JButton btnVidasaludable;
private javax.swing.JButton btnVolverAlumno;
private javax.swing.JMenuItem btnmnSalir2;
private javax.swing.JMenuItem btnmnSalirMaestro1;
private javax.swing.JMenuItem btnmnSalirMaestro2;
private javax.swing.JMenuItem btnmnSalirMaestro3;
private javax.swing.JMenuItem btnmnSalirMaterias1;
private javax.swing.JMenuItem btnmnSalirMaterias2;
private javax.swing.JMenuItem btnmnSalirMaterias3;
private javax.swing.JMenuItem btnmnSalirSecretaria;
private javax.swing.JButton bttAregarAlumno;
private javax.swing.JButton bttIniciarSesion;
private javax.swing.JButton bttModificarGrupo;
private javax.swing.JButton bttSubirCalificaciónM1;
private javax.swing.JButton bttSubirCalificaciónM2;
private javax.swing.JButton bttSubirCalificaciónM3;
private javax.swing.JButton bttVol1;
private javax.swing.JButton bttVol2;
private javax.swing.JButton bttVol3;
private javax.swing.JDialog dglAlumno;
private javax.swing.JDialog dglCalificaciones;
private javax.swing.JDialog dglMaestro1;
private javax.swing.JDialog dglMaestro2;
private javax.swing.JDialog dglMaestro3;
private javax.swing.JDialog dglMaterias1;
private javax.swing.JDialog dglMaterias2;
private javax.swing.JDialog dglMaterias3;
private javax.swing.JDialog dglModificarGrupo;
private javax.swing.JDialog dglNuevoAlumno;
private javax.swing.JDialog dglSecretaria;
private javax.swing.JTextField examen1;
private javax.swing.JTextField examen2;
private javax.swing.JTextField examen3;
private javax.swing.JTextField examenfinal;
private javax.swing.JComboBox<String> grupofiltro;
private javax.swing.JTextField idalumno;
```

```
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JLabel lblApellidoM;
private javax.swing.JLabel lblApellidoM1;
private javax.swing.JLabel lblApellidoP;
private javax.swing.JLabel lblApellidoP1;
private javax.swing.JLabel lblCiencias;
private javax.swing.JLabel lblCiencias2;
private javax.swing.JLabel lblCivicayEtica;
private javax.swing.JLabel lblContraseña;
private javax.swing.JLabel lblEspanol;
private javax.swing.JLabel lblEspanol2;
private javax.swing.JLabel lblGeografia;
private javax.swing.JLabel lblGrupo;
private javax.swing.JLabel lblGrupo1;
private javax.swing.JLabel lblHistoria;
private javax.swing.JLabel lblIngles1;
private javax.swing.JLabel lblIngles2;
private javax.swing.JLabel lblMatematicas;
private javax.swing.JLabel lblMatematicas2;
private javax.swing.JLabel lblMatematicas3;
private javax.swing.JLabel lblNombre;
private javax.swing.JLabel lblNombre1;
private javax.swing.JLabel lblQuimica;
private javax.swing.JLabel lblTecnologias;
private javax.swing.JLabel lblTitulo;
private javax.swing.JLabel lblTituloAlumno;
private javax.swing.JLabel lblTituloAlumno1;
private javax.swing.JLabel lblTituloAlumno2;
private javax.swing.JLabel lblTituloInicial;
private javax.swing.JLabel lblTituloMaestro1;
private javax.swing.JLabel lblTituloMaestro2;
private javax.swing.JLabel lblTituloMaestro3;
private javax.swing.JLabel lblTituloModificar;
private javax.swing.JLabel lblUsuario;
private javax.swing.JLabel lblVidasaludable;
private javax.swing.JComboBox<String> listGrupo;
private javax.swing.JComboBox<String> listRol;
```

```
private javax.swing.JComboBox<String> materiafiltro;
private javax.swing.JComboBox<String> materias;
private javax.swing.JMenu mnPrograma1;
private javax.swing.JMenu mnPrograma2;
private javax.swing.JMenu mnPrograma3;
private javax.swing.JMenu mnPrograma4;
private javax.swing.JMenuBar mnbMenu1;
private javax.swing.JMenuBar mnbMenu12;
private javax.swing.JMenuBar mnbMenu2;
private javax.swing.JMenuBar mnbMenu20;
private javax.swing.JMenuBar mnbMenu22;
private javax.swing.JMenuBar mnbMenu23;
private javax.swing.JMenuBar mnbMenu4;
private javax.swing.JMenuBar mnbMenu5;
private javax.swing.JMenuBar mnbMenu6;
private javax.swing.JMenu mnbPrograma;
private javax.swing.JDialog subircalif;
private javax.swing.JButton txtAceptarAlum;
private javax.swing.JTextField txtApellidoM;
private javax.swing.JTextField txtApellidoP;
private javax.swing.JPasswordField txtContraseña;
private javax.swing.JTextField txtContraseña1;
private javax.swing.JTextField txtNombre;
private javax.swing.JTextField txtTituloSecretaria;
private javax.swing.JTextField txtUsuario;
// End of variables declaration
}
```

Capítulo 6

Conclusión

El desarrollo de un sistema escolar es un proyecto muy ambicioso, el cuál lo decidimos al inicio del curso, bajo la tutela de tres de nuestros docentes. Sus respectivas materias ayudarian a tener un mejor desarrollo y visión del proyecto. Las materias involucradas fueron Base de datos impartida por la profesora Lirio Cruz Guerra, ingeniería de software I imparida por el profesor Rolando Pedro Gabriel y Paradigmas de programación impartida por Everardo De Jesus Pacheco. Con su ayuda pudimos tener un mejor desarrollo conforme avanzaba el curso y así al final del mismo tener terminado el proyecto o al menos en su mayoría. Al realizar este proyecto nos llevamos muchos conocimientos, de que u proyecto de software no solo es codificación sino que también requiere de mucho análisis y planeación, ver si es factible para llevar a cabo o si el tiempo dado será suficiente para terminarlo. Este proyecto buscada dar una ayuda a los profesores de una escuela secundaria para así agilizar muchos trámites y evitar errores que a veces se cometen al hacer trámites manuales.

6.1. Referencias

- [1] S.W. Ambler. The Elements of UML 2.0 Style. 1st Edición, Edición Kindle. 9 Mayo 2005
- [2] S. Martina, S. Mariano, H. Christian, K. Gerti. UML @ Classroom. Edición 2015. Springer. 9 marzo 2015
- [3] O. Antoni. Conceptual Modeling of Information Systems. 2007th Edición. Springer. 28 Agosto 200
- [4] R. Pressman, B. Maxim and L. Medellin Serna, Ingenieria de l software , 7th ed. Mexico [etc.]: McGraw-Hill, 2010, p. 20.
- [5] I. SOMMERVILLE, Ingeniería del software , 7th ed. Madrid(españa): Miguel Martín romo, 2005, p. 15.
- [6] J. Pilataxi, "UML Gota una Gota de Martin Fowler FEBRERO", Academia.edu de 1999. [En línea]. Disponible:https://www.academia.edu/9309108/UML_Gota_a_Gota_Martin_Fowler_FEB. [Consultado El: 02 - Feb - 2022].
- [7] Bruegge B, Dutoit A. Ingenieria de software orientado a objetos. Mexico: Pearson Educacion; 2002.
- [8] Booch G, Rumbaugh J, Jacobson I, Garcia Molina J, Saez Martin ezJ. El lenguaje unificado de modelado. Madrid: Pearson Educacion; 2010.
- [9] Pantaleo G, Rinaudo L, German Asenjo A. Ingenieria de software . Mexico: Alfaomega; 2015.
- [10] P. Rigby y M. Norris, Ingenieria de Software Explicada / Software Engineering Explained. Limusa, 1994.
- [11] McKinney, D. and Jimenez, A., n.d. Un dia de trabajo d e desarrollador de software.
- [12] T. Erler, UML. Bonn: Bhv, 2002.

6.2. Anexo 1.PSP

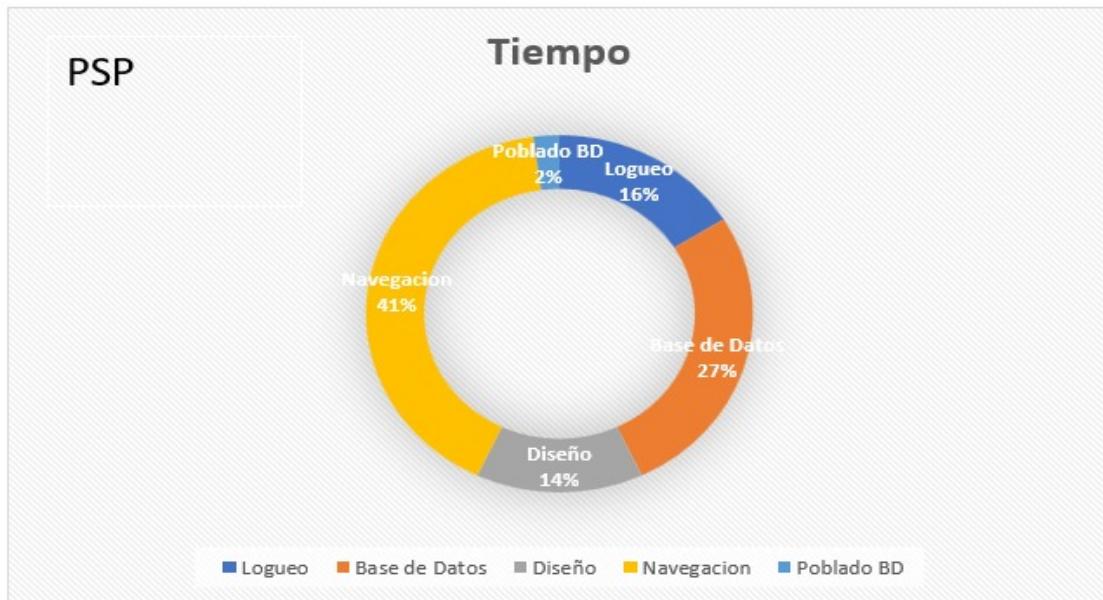


Figura 6.1: Gráfica del usu de PSP

Grafico donde se muestran los tiempos que se tomo en desarrollar los módulos del sistema, a gran medida lo que mas tiempo tomo fue la parte de navegación de ventanas y el desarrollo de la base de datos

En la gráfica se puede apreciar a ver que la parte de ventas nos llevó mucho tiempo aplicarlo.

6.3. Anexo 2.Galería de imágenes



Figura 6.2: Evidencia de la colaboración del equipo



Figura 6.3: Evidencia de la colaboración de Guillermo y Erik



Figura 6.4: Evidencia de la colaboración Lennin y Erik



Figura 6.5: Evidencia de la colaboración de Lennin y Erik