



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

SISTEMA WEB PARA GESTIÓN ACADÉMICA DE UN POSGRADO

T E S I S

PARA OBTENER EL TÍTULO DE

LICENCIADA EN INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN

PRESENTA

ALEXANDRA BAUTISTA OTERO

ASESORA

DRA. JOSEFINA GUERRERO GARCÍA

OCTUBRE, 2023

AGRADECIMIENTOS

A mis padres, que con amor y dedicación me formaron como una persona tenaz y trabajadora, a ellos que con mucho trabajo me dieron una vida llena de oportunidades para lograr mis objetivos y sueños.

A mí hermano, mi persona favorita en este mundo, le agradezco el apoyo y el cariño incondicional que siempre me ha ofrecido.

A mí misma, por nunca rendirme, por esforzarme al máximo y por siempre creer en mí.

Finalmente doy las gracias a los profesores que contribuyeron a mi proceso de aprendizaje y a mi crecimiento personal, así como a la Benemérita Universidad Autónoma de Puebla y a la Facultad de Ciencias de la Computación por brindarme las herramientas, oportunidades y conocimientos necesarios para mi crecimiento académico y profesional.

DEDICATORIA

Con todo mi amor y cariño les dedico a mis padres y hermano el esfuerzo y el resultado del trabajo realizado en el presente proyecto de Tesis.

Resumen

Gestionar un programa educativo en posgrado es esencial para garantizar su eficacia y calidad. La gestión permite coordinar la estructura curricular, asignar recursos adecuados, asegurar la calidad de la enseñanza, llevar un seguimiento del progreso académico y evaluar los resultados.

El presente trabajo de tesis tiene como objetivo diseñar y desarrollar un sistema web que proporcione herramientas que permitan al usuario interactuar y analizar información de estudiantes, docentes y asignaturas de un programa doctoral de la BUAP de forma clara y fácil. La aplicación facilitará el registro, eliminación y modificación, así como, exportar en formato Excel e imprimir información de acuerdo con criterios seleccionados por el usuario.

Acorde a las características del proyecto, se utilizó el método Kanban para llevarlo a cabo, puesto que contribuye a tener buena organización, así como procesos productivos y buen rendimiento; logrando así un producto final de calidad.

Usar un sistema web para la gestión académica facilita el acceso remoto a la información, agiliza los procesos administrativos, mejora la comunicación entre la coordinación, la planta académica y los estudiantes, y permite un seguimiento más eficiente del progreso académico. Además, centraliza los datos contribuyendo a la sostenibilidad y organización del programa.

De acuerdo con los puntos establecidos se obtuvo un sistema web robusto y eficiente que favorece la simplificación del proceso de gestión académica.

Índice

Tabla de contenido

Resumen	2
Índice	3
Índice de Figuras	4
Índice de Tablas	8
Introducción	10
Capítulo 1	12
Estado del Arte	12
Capítulo 2	16
Marco teórico metodológico	16
Marco Teórico	16
Análisis, Diseño y Modelado	16
Modelo MVC	22
Programación	23
Interfaces de Usuario	25
Pruebas	29
Marco metodológico	32
Método Kanban	32
Modelo Incremental	33
Capítulo 4	34
Implementación	34
Incremento 1	34
Incremento 2	94
Interfaces de Usuario	107
Capítulo 5	125
Pruebas	125
Pruebas de Caja Blanca	153
Conclusión	182
Referencias	183
Anexos	187

Índice de Figuras

Figura 1 Pantalla principal administración	12
Figura 2 Listado de cursos	13
Figura 3 Interfaz Registrar un nuevo usuario	14
Figura 4 Interfaz de horario de los docentes	15
Figura 5 Módulo de mantenimiento de materias	15
Figura 6 Clasificación de los diagramas de UML	18
Figura 7 Modelo del ciclo de vida Incremental	33
Figura 8 Diagrama de casos de uso del sistema web	37
Figura 9 Diagrama de secuencia Registrar Alumno	57
Figura 10 Diagrama de secuencia Eliminar Alumno	58
Figura 11 Diagrama de secuencia Modificar Alumno	59
Figura 12 Diagrama de secuencia Crear Expediente de Alumno	60
Figura 13 Diagrama de secuencia Consultar Alumnos Registrados	61
Figura 14 Diagrama de secuencia Registrar Docente	62
Figura 15 Diagrama de secuencia Eliminar Docente	63
Figura 16 Diagrama de secuencia Modificar Docente	64
Figura 17 Diagrama de secuencia Consultar Docentes Registrados	65
Figura 18 Diagrama de secuencia Agendar Coloquio	66
Figura 19 Diagrama de secuencia Modificar Coloquio	67
Figura 20 Diagrama de secuencia Coloquios Registrados	68
Figura 21 Diagrama de secuencia Asignar Cursos a Alumnos	69
Figura 22 Diagrama de secuencia Asignar Cursos a Docentes	70
Figura 23 Diagrama de secuencia Consultar Lista de Cursos	71
Figura 24 Diagrama de secuencia Consultar Información de Asignación	72
Figura 25 Mapa de navegación del sistema web	73
Figura 26 Diagrama entidad relación del sistema web	74
Figura 27 Estructura de archivos del proyecto web	75
Figura 28 Estructura general del formulario de Inicio de Sesión	77
Figura 29 Estructura general de la página principal	77
Figura 30 Estructura general de formularios	78
Figura 31 Estructura general de tablas	78
Figura 32 Estructura general de la página Asignar Cursos a Docentes	79
Figura 33 Estructura general de la página Consultar Información de Asignación	79
Figura 34 Código del método Iniciar Sesión (clase controlador)	80
Figura 35 Código del método Registrar Alumno (clase controlador)	81
Figura 36 Código del método Registrar Docente (clase controlador)	82
Figura 37 Código del método Registrar Coloquio (clase controlador)	82
Figura 38 Código del método Expediente de Alumno (clase controlador)	83
Figura 39 Código del método Eliminar (clase controlador)	84
Figura 40 Código del método Modificar (clase controlador)	84
Figura 41 Código del método Asignar Alumno (clase controlador)	85
Figura 42 Código del método Asignar Tutoría (clase controlador)	86
Figura 43 Código del método Asignar Materia (clase controlador)	87
Figura 44 Código del método Mostrar Tabla (clase controlador)	87
Figura 45 Código del método Registrar Alumno (clase modelo)	88
Figura 46 Código del método Registrar Docente (clase modelo)	88

Figura 47 Código del método Registrar Coloquio (clase modelo)	89
Figura 48 Código del método Expediente de Alumno (clase modelo)	89
Figura 49 Código del método Eliminar (clase modelo)	90
Figura 50 Código del método Modificar (clase modelo)	90
Figura 51 Código del método Asignar Alumno (clase modelo)	91
Figura 52 Código del método Asignar Tutoría (clase modelo)	91
Figura 53 Código del método Asignar Materia (clase modelo)	92
Figura 54 Código del método Mostrar Tabla (clase modelo)	92
Figura 55 Código del método Validar (clase modelo)	93
Figura 56 Mapa de navegación actualizado del sistema web	95
Figura 57 Estructura general actualizada del formulario de Inicio de Sesión	96
Figura 58 Estructura general actualizada de la página principal	97
Figura 59 Estructura general actualizada de formularios	97
Figura 60 Estructura general actualizada de tablas	98
Figura 61 Estructura general actualizada de la página Asignar Cursos a Docentes	98
Figura 62 Estructura general actualizada de la página Consultar Asignación	99
Figura 63 Código del método Buscar Curso (clase controlador)	100
Figura 64 Código del método Buscar Alumno (clase controlador)	101
Figura 65 Código del método Selección Alumno (clase controlador)	101
Figura 66 Código del método Selección Docente (clase controlador)	102
Figura 67 Código del método Mostrar Selección (clase controlador)	102
Figura 68 Código del método Limpiar Selección (clase controlador)	103
Figura 69 Código del método Buscar Curso (clase modelo)	104
Figura 70 Código del método Buscar Alumno (clase modelo)	104
Figura 71 Código del método Almacenar Selección (clase modelo)	105
Figura 72 Código del método Mostrar Selección (clase modelo)	105
Figura 73 Código del método Limpiar Selección (clase modelo)	106
Figura 74 Código del método Validar Curso (clase modelo)	106
Figura 75 Interfaz de Inicio de Sesión	107
Figura 76 Interfaz de página principal	108
Figura 77 Área de información	108
Figura 78 Interfaz Barra de Navegación	108
Figura 79 Interfaz Registrar Alumno	110
Figura 80 Interfaz Eliminar Alumno	111
Figura 81 Interfaz Modificar Información de Alumno	111
Figura 82 Interfaz Crear Expediente de Alumno	112
Figura 83 Interfaz Alumnos Registrados	113
Figura 84 Interfaz Selección de Información de Alumnos	113
Figura 85 Interfaz Información Seleccionada	114
Figura 86 Extensión Imprimir	114
Figura 87 Extensión Exportar	115
Figura 88 Interfaz Registrar Docente	115
Figura 89 Interfaz Eliminar Docente	116
Figura 90 Interfaz Modificar Información de Docentes	117
Figura 91 Interfaz Docentes Registrados	117
Figura 92 Interfaz Selección de Información de Docentes	118
Figura 93 Interfaz Agendar Coloquio	119
Figura 94 Interfaz Modificar Información de Coloquio	119

Figura 95 Interfaz Coloquios Registrados	120
Figura 96 Interfaz Asignar Curso a Alumno	121
Figura 97 Interfaz Asignar Curso a Docente	121
Figura 98 Interfaz formulario de Asignación de Tutoría	122
Figura 99 Interfaz formulario para Asignar Materias	122
Figura 100 Interfaz Información de Asignación	123
Figura 101 Interfaz Lista de Cursos	124
Figura 102 Iniciar Sesión r1 entrada	126
Figura 103 Iniciar Sesión r1 salida	126
Figura 104 Iniciar Sesión r2,r3,r4 entrada	126
Figura 105 Iniciar Sesión r2,r3,r4 salida	127
Figura 106 Registrar Alumno r1 entrada	128
Figura 107 Registrar Alumno r1 salida	128
Figura 108 Registrar Alumno r2 entrada	129
Figura 109 Registrar Alumno r2 salida	129
Figura 110 Modificar información de alumno r1 entrada	130
Figura 111 Modificar información de alumno r1 salida	130
Figura 112 Modificar información de alumno r2 entrada	131
Figura 113 Modificar información de alumno r2 salida	131
Figura 114 Eliminar alumno r1 entrada	132
Figura 115 Eliminar alumno r1 salida	132
Figura 116 Eliminar alumno r2 entrada	133
Figura 117 Eliminar alumno r2 salida	133
Figura 118 Crear expediente de alumno r1 entrada	134
Figura 119 Crear expediente de alumno r1 salida	134
Figura 120 Crear expediente de alumno r2 entrada	134
Figura 121 Crear expediente de alumno r2 salida	135
Figura 122 Selección de Información de Alumnos r1 entrada	136
Figura 123 Selección de Información de Alumnos r1 salida	136
Figura 124 Selección de Información de Alumnos r2 entrada	136
Figura 125 Selección de Información de Alumnos r2 salida	137
Figura 126 Registrar docente r1 salida	138
Figura 127 Registrar docente r1 entrada	138
Figura 128 Registrar docente r2 salida	139
Figura 129 Registrar docente r2 entrada	139
Figura 130 Modificar información de docente r1 entrada	140
Figura 131 Modificar información de docente r1 salida	140
Figura 132 Modificar información de docente r2 entrada	141
Figura 133 Modificar información de docente r2 salida	141
Figura 134 Eliminar docente r1 entrada	142
Figura 135 Eliminar docente r1 salida	142
Figura 136 Eliminar docente r2 entrada	143
Figura 137 Eliminar docente r2 salida	143
Figura 138 Selección de información de docente r1 entrada	144
Figura 139 Selección de información de docente r1 salida	144
Figura 140 Selección de información de docente r2 entrada	144
Figura 141 Selección de información de docente r2 salida	145
Figura 142 Modificar información de coloquio r1 entrada	146

Figura 143 Modificar información de coloquio r1 salida	146
Figura 144 Modificar información de coloquio r2 entrada	147
Figura 145 Modificar información de coloquio r2 salida	147
Figura 146 Asignar Cursos a Alumnos r1 entrada	148
Figura 147 Asignar Cursos a Alumnos r1 salida	148
Figura 148 Asignar Cursos a Alumnos r2 entrada	149
Figura 149 Asignar Cursos a Alumnos r2 salida	149
Figura 150 Asignar Cursos a Docentes r1 entrada	150
Figura 151 Asignar Cursos a Docentes r1 salida	150
Figura 152 Asignar Cursos a Docentes r2 entrada	151
Figura 153 Asignar Cursos a Docentes r2 salida	151
Figura 154 Buscar Información de Asignación r1 entrada y salida	152
Figura 155 Buscar Información de Asignación r2 entrada y salida	152
Figura 156 Diagrama de flujo del proceso Iniciar sesión	153
Figura 157 Grafo de flujo del proceso Iniciar sesión	154
Figura 158 Diagrama de flujo del proceso Registrar	155
Figura 159 Grafo de flujo del proceso Registrar	156
Figura 160 Diagrama de flujo del proceso Modificar	157
Figura 161 Grafo de flujo del proceso Modificar	158
Figura 162 Diagrama de flujo del proceso Eliminar	159
Figura 163 Grafo de flujo del proceso Eliminar	160
Figura 164 Diagrama de flujo del proceso de Crear Expediente de alumno	162
Figura 165 Grafo de flujo del proceso Crear Expediente de Alumno	163
Figura 166 Diagrama de flujo del proceso de Agendar Coloquios	164
Figura 167 Grafo de flujo del proceso Agendar Coloquios	165
Figura 168 Diagrama de flujo del proceso Asignación de Cursos	166
Figura 169 Grafo de flujo del proceso Asignación de Cursos	167
Figura 170 Diagrama de flujo del proceso de Consultar Información	168
Figura 171 Grafo de flujo del proceso Consultar Información	169
Figura 172 Diagrama de flujo del proceso Imprimir y seleccionar información	171
Figura 173 Grafo de flujo del proceso imprimir y seleccionar información	172
Figura 174 Diagrama de flujo del proceso Exportar y seleccionar información	174
Figura 175 Grafo de flujo del proceso Exportar y Seleccionar Información	175
Figura 176 Diagrama de flujo del proceso Imprimir información	177
Figura 177 Grafo de flujo del proceso Imprimir Información	178
Figura 178 Diagrama de flujo del proceso Exportar Información	180
Figura 179 Grafo de flujo del proceso Exportar Información	181

Índice de Tablas

Tabla 1 Esqueleto de un documento HTML	26
Tabla 2 Requerimientos funcionales	35
Tabla 3 Descripción del caso de uso Iniciar Sesión	38
Tabla 4 Descripción del caso de uso Registrar Alumno	39
Tabla 5 Descripción del caso de uso Registrar Docente	40
Tabla 6 Descripción del caso de uso Crear Expediente de Alumno	41
Tabla 7 Descripción del caso de uso Modificar Información de Alumno	42
Tabla 8 Descripción del caso de uso Modificar Información de Docente	43
Tabla 9 Descripción del caso de uso Modificar Información de Coloquio	44
Tabla 10 Descripción del caso de uso Eliminar Alumno	45
Tabla 11 Descripción del caso de uso Eliminar Docente	46
Tabla 12 Descripción del caso de uso Asignar Cursos a Alumnos	47
Tabla 13 Descripción del caso de uso Asignar Cursos a Docentes	48
Tabla 14 Descripción del caso de uso Agendar Coloquio	49
Tabla 15 Descripción del caso de uso Exportar Información	50
Tabla 16 Descripción del caso de uso Imprimir Información	51
Tabla 17 Descripción del caso de uso Consultar Información de Alumno	52
Tabla 18 Descripción del caso de uso Consultar Información de Docentes	53
Tabla 19 Descripción del caso de uso Consultar Información de Coloquios	54
Tabla 20 Descripción del caso de uso Consultar Información de Cursos	55
Tabla 21 Descripción del caso de uso Consultar Información de Asignación	56
Tabla 22 Descripción de carpetas del proyecto web	76
Tabla 23 Descripción de archivos del proyecto web	76
Tabla 24 Requerimientos no funcionales	94
Tabla 25 Recursos utilizados en el diseño del sistema web	96
Tabla 26 Tabla de decisión Iniciar Sesión	125
Tabla 27 Tabla de decisión Registrar Alumno	127
Tabla 28 Tabla de decisión Modificar Información de Alumno	130
Tabla 29 Tabla de decisión Eliminar Alumno	132
Tabla 30 Tabla de decisión Crear Expediente de Alumno	134
Tabla 31 Tabla de decisión Seleccionar Información de Alumno	135
Tabla 32 Tabla de decisión Registrar Docentes	137
Tabla 33 Tabla de decisión Modificar información de Docente	140
Tabla 34 Tabla de decisión Eliminar Docente	142
Tabla 35 Tabla de decisión Seleccionar Información de Docente	143
Tabla 36 Tabla de decisión Modificar información de Coloquio	145
Tabla 37 Tabla de decisión Asignar Curso a Alumno	148
Tabla 38 Tabla de decisión Asignar Curso a Docente	150
Tabla 39 Tabla de decisión Buscar Información de Asignación	152
Tabla 40 Complejidad ciclomática del proceso Iniciar Sesión	154
Tabla 41 Caminos básicos del proceso Iniciar Sesión	154
Tabla 42 Casos de prueba del proceso Iniciar Sesión	154
Tabla 43 Complejidad ciclomática del proceso Registrar	156
Tabla 44 Caminos básicos del proceso Registrar	156
Tabla 45 Casos de prueba del proceso Registrar	157
Tabla 46 Complejidad ciclomática del proceso Modificar	158

Tabla 47 Caminos básicos del proceso Modificar	158
Tabla 48 Casos de prueba del proceso Modificar	159
Tabla 49 Complejidad ciclomática del proceso Eliminar	160
Tabla 50 Caminos básicos del proceso Eliminar	161
Tabla 51 Casos de prueba del proceso Eliminar	161
Tabla 52 Complejidad ciclomática del proceso Crear Expediente de Alumno	163
Tabla 53 Caminos básicos del proceso Crear Expediente de Alumno	163
Tabla 54 Casos de prueba del proceso Crear Expediente de Alumno	164
Tabla 55 Complejidad ciclomática del proceso Agendar Coloquios	165
Tabla 56 Caminos básicos del proceso Agendar Coloquios	165
Tabla 57 Casos de prueba del proceso Agendar Coloquios	165
Tabla 58 Complejidad ciclomática del proceso Asignación de Cursos	167
Tabla 59 Caminos básicos del proceso Asignación de Cursos	167
Tabla 60 Casos de prueba del proceso Asignación de Cursos	168
Tabla 61 Complejidad ciclomática del proceso Consultar Información	169
Tabla 62 Caminos básicos del proceso Consultar Información	169
Tabla 63 Casos de prueba del proceso Consultar Información	170
Tabla 64 Complejidad ciclomática del proceso Imprimir y Seleccionar Información	172
Tabla 65 Caminos básicos del proceso Imprimir y Seleccionar Información	172
Tabla 66 Casos de prueba del proceso Imprimir y Seleccionar Información	173
Tabla 67 Complejidad ciclomática del proceso Exportar y Seleccionar Información	175
Tabla 68 Caminos básicos del proceso Exportar y Seleccionar Información	175
Tabla 69 Casos de prueba del proceso Exportar y Seleccionar Información	176
Tabla 70 Complejidad ciclomática del proceso Imprimir Información	178
Tabla 71 Caminos básicos del proceso Imprimir Información	178
Tabla 72 Casos de prueba del proceso Imprimir información	179
Tabla 73 Complejidad ciclomática del proceso Exportar Información	181
Tabla 74 Caminos básicos del proceso Exportar Información	181
Tabla 75 Casos de prueba del proceso Exportar Información	181

Introducción

La gestión es un proceso pilar para cualquier institución en cualquier rubro, específicamente la gestión aplicada a instituciones académicas toma el nombre de gestión académica, la cual es llevada a cabo por el coordinador académico. Este personaje se encarga de administrar los recursos de la organización, así como de gestionar la información de todos los individuos que conforman el organismo, ya sean, estudiantes, maestros o administrativos.

Contar con una correcta gestión permite obtener buenos resultados y cumplir metas planteadas.

Dado que es una tarea que se conforma de numerosos procesos y que se desenvuelve en diversas áreas, se puede convertir en una labor difícil de ejecutar y puede conllevar a afectaciones negativas a la institución.

Como ya se ha mencionado es un proceso fundamental para que la organización funcione correctamente y otorgue una educación de calidad, por ello, es de vital importancia utilizar herramientas que faciliten su realización.

La gestión académica de un posgrado conlleva la organización y control de distintos procesos como: convocatoria de nuevo ingreso, asignación de cursos a docentes y alumnos, coloquios, control de publicaciones, solicitud de oficios y constancias de docentes y alumnos, entre otros. A nivel de coordinación no existe un sistema que permita apoyar en la gestión de las actividades mencionadas. Es por ello por lo que surge la necesidad de crear un sistema web que apoye en la gestión académica de un posgrado.

Objetivo General

Diseñar y desarrollar una aplicación web para coordinar un programa doctoral en la Benemérita Universidad Autónoma de Puebla.

Objetivos Específicos

- Diseñar y crear una base de datos que le proporcione integridad y exactitud a la información almacenada.
- Asegurar y controlar el acceso al sistema web por medio de un usuario y contraseña asignados.
- Programar módulos funcionales que faciliten la alta, baja, modificación y salida de información, logrando así una gestión eficaz.
- Diseñar y programar un sistema web que permita gestionar información sin tener que realizar procesos tediosos.
- Diseñar y programar una interfaz intuitiva para el usuario.

El trabajo está estructurado en capítulos organizados de la siguiente manera:

Capítulo I. Está referido a la revisión del estado del arte, donde se realiza una breve descripción de trabajos previos que abordan aspectos similares en comparación con este trabajo de tesis.

Capítulo II. Marco teórico metodológico, donde se presentan conceptos fundamentales que sustentan esta tesis, así como una revisión de las metodologías, enfoques y técnicas utilizadas en el desarrollo de sistemas web.

Capítulo III. Se describe la implementación de la solución tecnológica siguiendo el proceso de diseño y desarrollo del sistema web.

Capítulo IV. Se describen las pruebas realizadas para validar y verificar el sistema web desarrollado.

Posteriormente, se presentan las conclusiones del trabajo poniendo de manifiesto los logros obtenidos, también se presentan las sugerencias a ser consideradas como líneas de investigación que podrían explorarse en el futuro. Seguidamente, las referencias bibliográficas y las referencias digitales a través de direcciones de internet; concluyendo el documento con los Anexos que incluyen un glosario de términos y una lista de abreviaturas.

Capítulo 1

Estado del Arte

La aplicación de tecnologías de la información y comunicación posibilitan la creación de instrumentos de trabajo que ofrecen una considerable oportunidad de progresar a cualquier tipo de organización. Esta tecnología posibilita el procesamiento y almacenamiento de datos, así como el acceso remoto de todos los usuarios al sistema.

Estos sistemas ayudan a la toma acertada de decisiones, al igual que, garantizar el éxito, limitar riesgos, reducir costos e incrementar la confiabilidad en la organización.

Un ejemplo claro se ubica en el trabajo de (Zurita Lara, 2020) cuyo objetivo fue crear una plataforma web que proporcione diversas herramientas para gestionar de forma eficiente los procesos académicos y administrativos correspondientes a la organización DIENAV. Algunos de los beneficios de este trabajo fueron la agilidad de los procesos, la seguridad operativa, el ahorro en material de oficina y el registro adecuado de información (Figura 1).



Figura 1 Pantalla principal administración, (Back - End). Fuente: Zurita Lara, 2020.

Dentro de la aplicación, el usuario puede realizar las siguientes acciones según el tipo de perfil que tenga:

ALUMNOS

Inscripción.
Matriculación.
Registro de calificaciones.
Tramitar información.

DOCENTES

Tramitar información

ADMINISTRADOR

Creación de usuarios adaptables según su perfil.
Asignación de materias

Para fundamentar el proyecto se realizaron acciones como revisión de documentos de la institución, encuestas y entrevistas a los diferentes tipos de usuarios que usarían el sistema. La metodología empleada para el desarrollo de la plataforma web fue XP, ya que posibilita el trabajo en equipo y reducción de riesgos, con el fin de garantizar la satisfacción del cliente.

De acuerdo con el autor los resultados obtenidos fueron positivos ya que se crearon procesos automatizados de forma incremental y por iteraciones, los cuales cubren los requerimientos más urgentes del cliente.

Otro trabajo para destacar es el de (Mantilla Oliva, 2019) donde se describe una herramienta que automatiza procesos de gestión escolar en la universidad de Piura, Perú. La mencionada institución ya hace uso del software Sistema Integral de Gestión Académica “SIGA”, el proyecto que se presenta se ofrece como complemento al mencionado sistema (Figura 2).

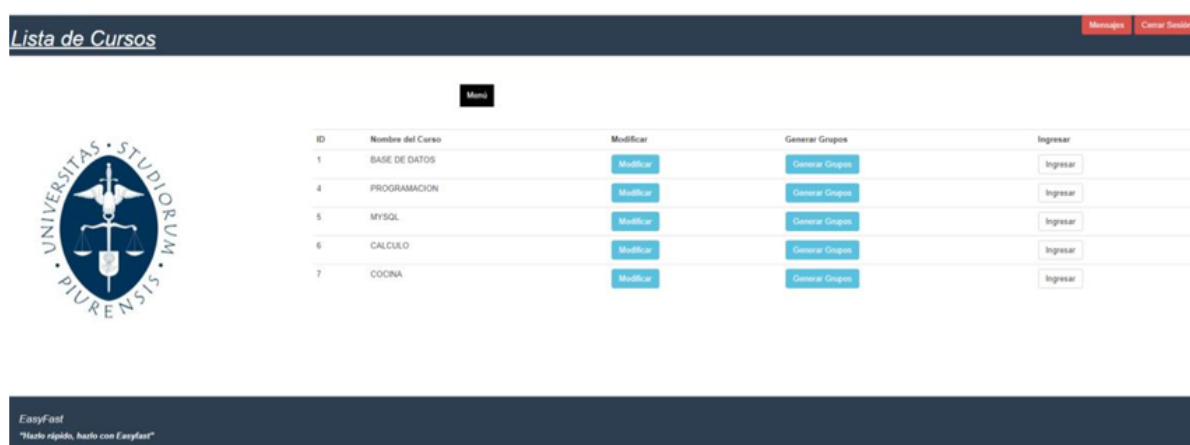


Figura 2 Listado de cursos. Fuente: Mantilla Oliva, 2019.

SIGA es un software orientado a alumnos y docentes, el cual, tiene la finalidad de planificar procesos pertenecientes a la gestión educativa de la institución, tales como: manejo de cursos, usuarios, talleres, evaluaciones, ingreso de alumnos, gestión de docente y administración de cada curso. De igual manera sirve como herramienta para administrar todos los aspectos académicos-administrativos como son, el control en los procesos de preinscripción, inscripción, selección y admisión de todos los aspirantes a cada programa académico, hasta el proceso de graduación.

Finalmente, se menciona el uso del nombrado sistema, para el manejo del archivo de los estudiantes, conformado por el manejo de los grados y el registro de las calificaciones de cada uno.

El objetivo del trabajo fue mostrar el desarrollo del prototipo de una plataforma de gestión académica, enfocada en la interrelación entre alumnos y catedráticos de la universidad de Piura. Dicha plataforma complementaria emerge de las necesidades y dificultades que han enfrentado docentes y alumnos de la organización.

Las ventajas de utilizar un sistema de estas características fortalecen el desarrollo de las actividades académicas al tener la información requerida por los estudiantes de manera correcta, oportuna y almacenada en una base de datos de fácil acceso (Mantilla Oliva, 2019, p.51).

En (Barrientos Rodríguez et al., 2022) se presenta la implementación de un sistema web para la gestión de la información relacionada con los programas de maestrías de una institución en La Haba, Cuba. Para el desarrollo del sistema se empleó el Proceso Unificado de Desarrollo (RUP) como metodología de desarrollo de software para el análisis, implementación y documentación de sistemas y que se divide en las fases de inicio, elaboración, construcción y transición; el Lenguaje Unificado de Modelado (UML), ya que está consolidado como el lenguaje estándar en el análisis y diseño de sistemas; WampServer 2.0f como entorno de desarrollo ya que su uso permite servir páginas html a internet y poder gestionar datos en ellas, al mismo tiempo proporciona lenguajes de programación para el desarrollo de aplicaciones web. WAMP es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas Apache, como servidor web; MySQL, como gestor de bases de datos y PHP, Perl o Python, como lenguajes de programación (Figura 3).

Figura 3 Interfaz Registrar un nuevo usuario. Fuente: Barrientos Rodríguez et al., 2022

Apolaya Ñaupá (2018) desarrolló una propuesta informática para la mejora de la gestión académica en un Instituto de Educación Superior Tecnológico Público considerando el proceso de matrícula, elaboración de carga lectiva, horario de clase, designación de ambientes y notas de los estudiantes mediante el desarrollo de una aplicación web, logrando la mejora en el proceso en automatizar, simplificar y agilizar los procesos. La aplicación web fue desarrollada siguiendo la metodología RUP y modelando la propuesta con UML; se utilizó Visual Basic 2017 y en la base de datos MSSQL y aplicación Android (Figura 4).

Figura 4 Interfaz de horario de los docentes: Fuente: Apolaya Ñaupá, 2018.

Mendoza Rivilla (2017) presentó un Sistema web para la gestión y control de los procesos de una unidad de titulación, el cual brinda apoyo a los estudiantes, tutores, coordinadores y docentes, agiliza el proceso y genera evidencias de las actividades de este. El programa fue desarrollado con PHP para la lógica del aplicativo, MySQL como gestor de la base de datos, y complementos para la parte del diseño como JQuery (biblioteca javascript), Bootstrap (framework para el maquetado CSS), entre otros. Para la arquitectura del aplicativo se usó el patrón de diseño M.V.C. (Modelo, Vista, Controlador). Se usó el método de lista de chequeo para controlar el cumplimiento de los escenarios de pruebas con respecto a los requisitos planteados por el usuario (Figura 5).

No.	Materia	Carrera	Estado	Action
1	Bases de datos I	Ingeniería Sistemas	A	[Pencil] [Check] [X]
2	Programación I	Ingeniería Sistemas	A	[Pencil] [Check] [X]
3	Programación II	Ingeniería Sistemas	A	[Pencil] [Check] [X]
4	Sistemas de Información I	Ingeniería Sistemas	A	[Pencil] [Check] [X]
5	Sistemas de Información II	Ingeniería Sistemas	A	[Pencil] [Check] [X]
6	TCPIA	Ingeniería Sistemas	A	[Pencil] [Check] [X]

Figura 5 Módulo de mantenimiento de materias. Fuente: Mendoza Rivilla, 2017.

Capítulo 2

Marco teórico metodológico

El marco teórico está constituido por un conjunto de teorías, enfoques teóricos y conceptos que se consideran válidos para el encuadre correcto de este trabajo de tesis. El marco metodológico es la descripción de los pasos, etapas, fases, y actividades que se llevarán a cabo para el diseño de la solución.

Marco Teórico

Análisis, Diseño y Modelado

Análisis de Requerimientos

De acuerdo con Kotonya, & Sommerville (2000) los requisitos del software expresan las necesidades y los apremios colocados en un producto de software que contribuye a la solución de un cierto problema del mundo real.

El análisis de requerimientos (Núñez, 2018) es la primera etapa y la más importante en el desarrollo de cualquier proyecto de software, ya que se precisan los cimientos en lo que se basaran las fases posteriores. Este análisis permite conocer las necesidades que el sistema debe solucionar, así como los costos y el plan de desarrollo a efectuar.

Los requerimientos son capacidades o características que debe contener el sistema para lograr el objetivo del proyecto. Deben ser descritos de forma clara y precisa con el objetivo de que sean entendibles para cualquier participante del proyecto.

Tipos de Requerimientos

Requerimientos funcionales: describen las funciones que el sistema debe proveer a los usuarios finales. Este tipo de requerimientos sirven para entender qué es lo que el sistema debe hacer.

Requerimientos no Funcionales: especifican las propiedades del sistema, como son, sus límites y necesidades. Este tipo de requerimientos permiten conocer cómo debe actuar el sistema.

Clasificación

- Requerimientos de producto: describen la conducta del sistema, como la eficiencia, fiabilidad y portabilidad.
- Requerimientos de organización: describen los acuerdos procedentes de la organización entre el desarrollador y el cliente, como son el tiempo de entrega, el tiempo de implementación y los estándares.

- **Requerimientos externos:** describen los requisitos procedentes de factores externos al sistema, como son los requerimientos de interoperabilidad, éticos y legislativos.

El proceso de análisis de requerimientos inicia con la identificación y definición del problema, después se recopilan los requisitos de las partes interesadas, como usuarios, clientes y otras partes a las que afectará el sistema de software. A continuación, se documentan y organizan los requisitos y se resuelve cualquier incoherencia o ambigüedad. El paso final en el proceso de análisis de requisitos es validar los requisitos, lo que incluye probar y evaluar los requisitos para garantizar que son completos, coherentes y alcanzables.

El modelado de requisitos es un proceso crítico para documentar y gestionar los requisitos de un sistema. Los beneficios del modelado de requisitos incluyen comunicación mejorada, ambigüedad reducida y una comprensión común de los requisitos del sistema. Existen varias técnicas para el modelado de requisitos, incluidos casos de uso, modelos de datos y gráficos de estado. La técnica más popular es el lenguaje de modelado unificado (UML).

Diseño de software

El diseño es el proceso para definir la arquitectura, los componentes, las interfaces y otras características de un sistema o un componente (IEEE Std 610.12-90). El resultado de un buen diseño de software debe describir la arquitectura del software describiendo la estructura del software y organización e identificar a nivel superior los componentes; el diseño detallado describe cada componente suficientemente para tener en cuenta su construcción.

Algunos conceptos importantes del diseño de software son (Pressman, 2010):

- **Abstracción:** característica de la programación orientada a objetos que va a permitir que los objetos puedan interactuar sin necesidad de conocer los detalles del funcionamiento. De esta forma, el enfoque se centra en aspectos relevantes de las entidades.
- **Arquitectura:** estructura general del software y las formas en las que ésta da integridad conceptual a un sistema.
- **Patrones:** un patrón de diseño describe una estructura de diseño que resuelve un problema particular de diseño dentro de un contexto específico y entre “fuerzas” que afectan la manera en la que se aplica y en la que se utiliza dicho patrón.
- **Modularidad:** el software se divide en componentes con nombres distintos y abordables por separado (módulos) que se integran para satisfacer los requerimientos del problema.
- **Ocultamiento de información:** sugiere que los módulos se caractericen por decisiones de diseño que se oculten (cada una) de las demás. Esto permite hacer cumplir las restricciones de acceso tanto a los detalles de procedimiento como a cualquier estructura de datos local que utilice el módulo.
- **Independencia funcional:** es el resultado de la separación de problemas y de los conceptos de abstracción y ocultamiento de información. La independencia se evalúa con el uso de la cohesión y el acoplamiento. Un módulo cohesivo debe hacer sólo una cosa; el acoplamiento es una indicación de la interconexión entre módulos en una

estructura de software, y depende de la complejidad de la interfaz entre módulos, del grado en el que se entra o se hace referencia a un módulo y de qué datos pasan a través de la interfaz.

Modelo del diseño

Un modelo de software es una representación abstracta de un sistema o aplicación, que se utiliza para comprender, diseñar y comunicar aspectos clave del software antes de su desarrollo. Puede incluir diagramas, especificaciones y descripciones que ayudan a los desarrolladores a visualizar la estructura, funciones y relaciones del software (Fowler y Scott, 1999).

UML

El Lenguaje de Modelado Unificado – del inglés Unified Modeling Language (UML) – es un lenguaje basado en diagramas para la especificación, visualización, construcción y documentación de cualquier sistema complejo, aunque nosotros nos centraremos en el caso específico de sistemas software (Booch, Rumbaugh, y Jacobson, 1999).

UML en su versión 2.5.1 cuenta con dos clasificaciones de diagramas: los diagramas estructurales y los diagramas de comportamiento (ver figura 6).

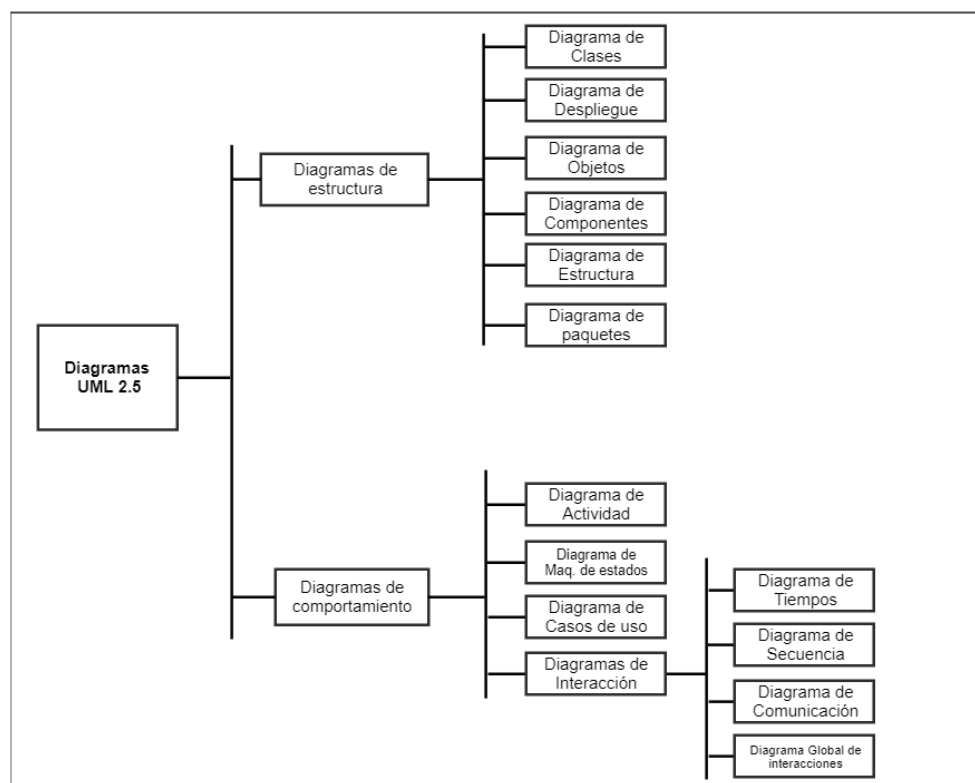


Figura 6 Clasificación de los diagramas de UML. Fuente: <https://diagramasuml.com/>

Diagramas de Casos de Uso

La técnica de casos de uso fue desarrollada por Ivar Jacobson en 1987 y posteriormente anexada al Lenguaje Unificado de Modelado. Dicha técnica muestra la relación entre el sistema y los usuarios finales. UML emplea los casos de uso para describir el funcionamiento de un sistema, subsistema o clase desde la percepción del usuario que interactúa con la aplicación (García Peñalvo y García Holgado, 2018).

Componentes

- Actor: elemento que representa a los usuarios externos que interactúan con el sistema.
- Casos de Uso: componente que representa los requisitos funcionales del sistema.
- Sujeto: elemento que representa el sistema a modelar.
- Relación: componente que representa el tipo de interacción que tiene el usuario con un caso de uso.

Diagramas de Secuencia

De acuerdo con (García Peñalvo y Vázquez Ingelmo, 2019) el diagrama de secuencia forma parte de los diagramas de interacción de UML, tiene como objetivo mostrar de forma gráfica la secuencia cronológica de mensajes que llevan a cabo las diferentes partes lógicas del sistema, para realizar algún evento que haya solicitado el usuario final.

El mencionado diagrama trabaja con dos dimensiones; el eje horizontal representa los objetos y el eje vertical representa el tiempo. El transcurso del tiempo se simboliza de la parte superior del diagrama a la parte inferior.

Componentes

- Objetos: cualquier elemento software que desempeñe una tarea en específico dentro del sistema.
- Línea de vida: representa el paso del tiempo a medida que se extiende hacia abajo. Esta línea vertical discontinua representa eventos secuenciales que le ocurren a un objeto durante el proceso graficado. Las líneas de vida pueden comenzar con una figura rectangular etiquetada o un símbolo de actor.
- Activación: periodo de tiempo que requiere un objeto para finalizar una acción.
- Loop: se emplea para modelar escenarios del tipo "Si... entonces...", es decir, una circunstancia que solo sucederá en determinadas condiciones.
- Alternative: simboliza una decisión (que, por lo general, es mutuamente exclusiva) entre dos o más secuencias de mensajes. Para representar alternativas, emplea la figura rectangular etiquetada con una línea discontinua en su interior.
- Mensajes: comunicación entre el conjunto de objetos que forman parte del sistema.
 - Simples: envío de información de un objeto a otro, la comunicación entre los objetos es unidireccional.

- Síncronos: el objeto emisor transmite información al objeto receptor y queda a espera de la respuesta para poder continuar su labor.
- Asíncronos: el objeto emisor transmite información al objeto receptor y no espera respuesta para continuar su trabajo.
- Rechazado: acción transmitida no se puede ejecutar.

Mapa de Navegación

Gráfico que sirve para visualizar la estructura general de un sistema o aplicación de software; permite planificar y organizar la distribución de los componentes de la interfaz del proyecto a trabajar. El mapa de navegación no es flexible, por lo que no da lugar a que existan errores (Quenguan Acosta, 2011).

Esté diagrama es aplicable en cualquier metodología ágil, ya que funge como plano para la creación del sistema, lo cual conlleva a crear una navegación óptima y a diseñar un sistema intuitivo para el usuario final.

Tipos de Estructuras

- Estructura lineal: el recorrido de las páginas que componen al sistema se realiza de la página inicial a la página final, lo cual genera una navegación intuitiva pero tardía.
- Estructura jerárquica: estructura compuesta por niveles, los cuales se encuentran enlazados entre sí como niveles y subniveles, este tipo de estructura le permite al usuario controlar la navegación dentro del sistema.
- Estructura radial: estructura que relaciona las páginas secundarias con la página principal, pero entre ellas no existe ninguna relación.
- Estructura de red: estructura que interconecta todas las páginas, por lo que el usuario no tiene límites en la navegación.

Bases de Datos

El concepto de bases de datos de acuerdo con Gutiérrez Díaz (s.f.) hace referencia a un repositorio digital, que permite almacenar de manera organizada un conjunto de datos estructurados. Las bases de datos tienen como objetivo satisfacer las necesidades de almacenamiento de una organización, proporcionando fácil acceso a la información.

El Sistema de Gestión de Base de Datos (Marqués, 2011) es un programa software que proporciona los servicios necesarios al usuario para gestionar la información almacenada en una base de datos y brinda al usuario control de acceso al repositorio.

El SGBD permite controlar el acceso a la base de datos, así como definir su estructura y el tipo de información que contendrá, de igual forma posibilita crear la base de datos, insertar, modificar, eliminar y consultar información del repositorio.

Las bases de datos relacionales siguen el modelo relacional, el cual fue propuesto en 1970 por Edgar Frank Codd y sigue el concepto matemático de relación. El modelo relacional considera a una base de datos como una agrupación de relaciones, donde cada relación representa una tabla y la tabla a su vez representa información referente a una entidad del mundo real (Azure, s.f.).

La estructura de la tabla está compuesta por columnas y filas, las columnas simbolizan los atributos del objeto que se quiere almacenar y las filas contienen los valores de dichos atributos. Cada tupla tiene una clave única que permite identificarla y garantiza que no exista la duplicidad de registros.

Al utilizar este tipo de estructura se logra evitar que se genere redundancia de información, proporciona un acceso sencillo a los datos, permite trabajar con grandes cantidades de información y mantener la integridad de los datos.

Diagrama Entidad Relación

El modelo Entidad Relación fue introducido por Peter Pin-Shan Chen en 1976 y tiene el objetivo de modelar y diseñar bases de datos relacionales siguiendo los requisitos del proyecto a realizar.

El diagrama permite visualizar las relaciones e interacciones de las entidades antes de implementar la base de datos, funge como guía para construir la estructura lógica del repositorio (Marqués, 2011).

Componentes

- Entidad: objeto del mundo real sobre el cual se reúne información.
- Entidad Fuerte: entidad que se puede identificar por sus atributos.
- Entidad débil: entidad que no se puede identificar con sus atributos, este tipo de entidad requiere de otra entidad para poder existir.
- Atributo: propiedades o características de interés del objeto.
- Atributos opcionales: no es necesario conocer el valor del atributo en todo momento.
- Atributos obligatorios: es necesario conocer el valor del atributo en todo momento.
- Clave candidata: propiedad aspirante a ser clave primaria.
- Clave primaria: propiedad principal por la que se identificara una instancia de la tabla de forma inequívoca.
- Clave foránea: propiedad que relaciona una tabla con otra, es decir el valor de la clave foránea en la tabla T2 corresponde al valor de clave primaria en la tabla T1 (considerando dos tablas).

Tipos de Relaciones entre Entidades

Rocha (2017) define la relación entre entidades como la representación de la interacción entre dos o más entidades y la cardinalidad como el número de veces que una entidad se puede relacionar con otra.

Considerando dos tablas T1 y T2, se describe la relación 1 a 1 cuando un registro perteneciente a T1 se relaciona únicamente con un registro de T2 y viceversa. La relación 1 a N hace referencia a que un registro de T1 se relaciona con cero o muchas instancias de T2 y un registro perteneciente a T2 se relaciona únicamente con una instancia de T1. La relación de N a 1 se crea cuando una instancia de T1 se relaciona únicamente con un registro de T2 y un registro de T2 se relaciona con cero o muchos registros de T1, por último la relación N a N se produce cuando un registro perteneciente a T1 se relaciona con cero o muchos registros de T2 y viceversa.

Modelo MVC

El Modelo Vista Controlador (MVC) es un patrón de diseño y arquitectura de software, se basa en dividir el proyecto a desarrollar en 3 componentes lógicos, cada módulo es totalmente independiente de los otros y solo realiza una función en específico. Esta estructura les permite a los desarrolladores del sistema contar con mayor control en la implementación del proyecto y genera sistemas de fácil comprensión y de alta calidad (Armao et al.,2016).

Componentes

- Modelo: componente responsable de generar los cambios o consultas solicitadas por el usuario.
- Vista: elemento encargado de obtener los eventos o acciones de entrada en el sistema que genere el usuario.
- Controlador: componente que comunica al modelo los eventos obtenidos en la vista para ser procesados.

Descripción de la interacción de los componentes lógicos de MVC

1. El usuario genera un evento en la interfaz.
2. La vista recibe la acción de entrada y la canaliza al controlador.
3. El controlador recibe el evento y lo comunica al modelo.
4. El modelo realiza la acción o consulta y devuelve la información resultante al controlador.
5. El controlador envía la nueva información a la vista.
6. La vista obtiene los datos provenientes del modelo y genera una nueva vista.
7. La interfaz de usuario queda a espera de un nuevo evento por parte del usuario.

Estructura de un sistema web implementado con MVC

- Vista: módulo encargado de la estructura de la página HTML y estilos CSS.
- Controlador: módulo responsable del código que obtiene la información de entrada y genera el contenido HTML.
- Modelo: módulo que se encuentra en contacto directo con la BD del sistema y trabaja con las reglas y restricciones que permiten transformar la información.

Programación

PHP

Lenguaje de programación orientado principalmente al desarrollo de proyectos web y se caracteriza por ser un lenguaje de código abierto, así como de ser ejecutado únicamente del lado del servidor. PHP permite implementar las instrucciones que hacen funcionable a la plataforma web, así como de comunicar la interfaz gráfica y el servidor del sistema. Todas las instrucciones codificadas en el mencionado lenguaje deben estar contenidas entre delimitadores PHP para poder ser interpretadas. Asimismo, es capaz de interactuar con bases de datos, permitiendo la gestión de información de cualquier sitio web (PHP, s. f.).

SQL

De acuerdo con Oppel y Sheldon (2010) SQL es un lenguaje de consulta estructurado que permite administrar la información contenida en una base de datos relacional. Las operaciones que se pueden realizar con esta herramienta son, creación de bases de datos y tablas, inserción de información, modificación de información y tablas, eliminación de tablas e información, así como consultar la información contenida en la base de datos.

MySQL

Gestor de bases de datos relacionales, que emplea el lenguaje de consulta estructurado para administrar la información de una base de datos. Esta herramienta actúa como intermediario entre un sitio web y la base de datos implementada en él. Permite gestionar de manera ordenada y eficiente el conjunto de datos que maneja la aplicación, así como garantizar la integridad y consistencia de la información. MySQL se caracteriza por ser de código abierto y por ser adaptable a diferentes entornos de desarrollo y lenguajes de programación (Robledano, 2023).

XAMPP

Herramienta orientada al sector de desarrollo web, se caracteriza por ser un software multiplataforma, gratuito y de código libre; permite probar y examinar el funcionamiento de proyectos web implementados de manera local en un ordenador personal. Este paquete utiliza MariaDB o MySQL como gestor de base de datos, como servidor local emplea Apache y usa PHP y Perl como lenguajes de programación (Apache Friends, s. f.).

Sistemas operativos compatibles con XAMPP: Windows, GNU/Linux, Mac Os

PDO

Objetos de Datos de PHP es una extensión de PHP que actúa como nivel de abstracción sobre el proceso de acceso a cualquier gestor de base de datos; contiene controladores con características de diferentes SGBD que permiten realizar consultas y recuperación de información empleando siempre las mismas funciones (PHP: PDO, s. f.). Para realizar consultas en PDO la consulta debe seguir la sintaxis del sistema gestor de base de datos en el que se creó el repositorio.

Gestores de bases de datos compatibles con PDO: MySQL, Oracle, Microsoft SQL Server, SQLite

Clases componentes de PDO

- PDO: realiza la conexión entre la base de datos y el sistema web.
- PDOStatement: permite trabajar con sentencias SQL para la inserción, modificación, eliminación y recuperación de información de una base de datos.
- PDOException: gestiona las fallas generadas por PDO.

Visual Studio Code

VSC (Visual Studio Code, s.f.) permite editar, ejecutar y depurar instrucciones codificadas en cualquier lenguaje de programación. Esta herramienta de codificación se caracteriza por ser gratuita, multiplataforma y de código abierto, así como de ser personalizable de acuerdo con las necesidades de cada usuario.

Extensiones de Visual Studio Code para desarrollo de Sitios Web

- Live Server: extensión que brinda un servidor de desarrollo para ejecutar proyectos web.
- ESLint: extensión orientada a encontrar y solucionar automáticamente errores de sintaxis originados en código JavaScript.
- PHP IntelliSense: extensión que permite autocompletar código PHP.

- PRETTIER: extensión que sirve para formatear el código de acuerdo con las reglas del lenguaje en el que fue implementado.

Sistemas operativos compatibles con Visual Studio Code: Windows, GNU/Linux, Mac Os

phpMyAdmin

Software desarrollado por The phpMyAdmin Project; este sistema fue implementado en el lenguaje de programación PHP; proporciona herramientas que permiten exportar datos en distintos formatos, efectuar sentencias SQL para la gestión de información contenida en una base de datos, así como de gestionar, configurar y facilitar el mantenimiento a servidores (Contributors, s. f.).

phpMyAdmin es una aplicación web que permite acceder a los gestores de bases de datos MySQL y MariaDB a través de un navegador web; el acceso a este sistema se puede realizar mediante XAMPP o a través de un buscador web.

Interfaces de Usuario

HTML

El Lenguaje de Marcas de Hipertexto (MDN, s. f. a) es un lenguaje informático que utiliza etiquetas para definir el formato que tendrá el texto contenido en un archivo. Este tipo de lenguaje funge como el conjunto de reglas que los navegadores web deben seguir para mostrar el contenido de una página o aplicación web. Los elementos que contiene un sistema web son definidos por el conjunto de marcas que emplea HTML, así como la estructura visual de la página. Los elementos que pueden conformar una página web son texto, imágenes, videos, listas, tabla de datos, entre otros.

Estructura de una marca o etiqueta

- Etiqueta de apertura: contiene el nombre de la marca y los atributos que tendrá ese elemento; su objetivo es establecer dónde se empieza a aplicar el efecto de la etiqueta.
- Contenido: información o texto que será afectado por el formato o estilo que defina esa marca.
- Etiqueta de cierre: contiene la barra de cierre y el nombre de la etiqueta; su objetivo es establecer hasta donde se aplica el efecto de la marca.

En la Tabla 1 se observan los elementos de un documento en HTML.

CSS

Hojas de estilo en cascada es un lenguaje de diseño gráfico creado con el objetivo de afinar y estilizar los elementos estructurados en un documento HTML. La sintaxis que maneja CSS son reglas, las cuales definen las propiedades visuales que modificaran a un elemento HTML en específico. Con CSS es posible aplicar estilos a más de un archivo creado con el Lenguaje de Marcas de Hipertexto (MDN, s.f. b).

Las características que se pueden personalizar son: color, tamaño, tipografía, posición, así como la visualización del sistema web en diferentes dispositivos electrónicos. Las hojas de estilo en cascada se pueden aplicar a los archivos HTML de diferentes maneras.

Tabla 1 Esqueleto de un documento HTML

Etiqueta /Marca	Funcionalidad
<!DOCTYPE html>	Declaración que notifica al navegador web, que versión de HTML se utilizó para desarrollar el archivo a ejecutar, brinda la información necesaria para poder interpretar correctamente la página o aplicación web.
<html></html>	Elemento raíz de la estructura que comprende a las demás marcas que conforman el esqueleto del documento HTML..
<head></head>	Permite añadir información necesaria para el correcto funcionamiento de la página o aplicación, como hojas de estilo y scripts.
<meta charset="utf-8">	La etiqueta meta sirve para proporcionar información sobre el archivo HTML. La propiedad charset permite definir el conjunto de caracteres que procesara el proyecto web.El valor utf-8 hace referencia a la agrupación de caracteres que permite a cualquier sistema web mostrar su contenido en cualquier idioma.
<title> </title>	Define el título que se visualizará en la pestaña del buscador web al ejecutar y mostrar la página .
<body></body>	Contiene todos los elementos que mostrará el proyecto web, como imágenes, videos, listas, botones etc.

Métodos para aplicar estilos a archivos HTML

- Interno: el código CSS se agrega directamente al archivo HTML a personalizar, ubicando las reglas de estilo dentro de la etiqueta `<head>` `</head>`. Las reglas creadas de esta forma solo afectan al documento de hipertexto en el que fueron creadas, y se ejecutarán cada vez que el archivo sea actualizado en el navegador web.
- Inline: el código CSS se implementa dentro de los elementos que estructuran el archivo de hipertexto, utilizando la etiqueta `<style>` `</style>`. Al emplear este método se debe personalizar de forma independiente cada componente que estructura el documento.
- Externo: el código CSS se añade al archivo HTML por medio de una dirección URL dentro de la etiqueta `<head>` `</head>`. La dirección hace referencia a la ubicación de un archivo externo de extensión `.css`, en el cual se alojan todas las reglas de estilo creadas. Esta forma de agregar estilos permite personalizar diferentes archivos de hipertexto.

Sintaxis de una regla en CSS

- Selector: elemento HTML al que se le aplicaran propiedades para estilizar su aspecto visual.
- Bloque de declaración: área limitada por corchetes, que contiene las declaraciones que afectan al elemento HTML indicado por el selector.
- Declaración: componente formado por la propiedad a modificar y el valor a efectuar.

JavaScript

Lenguaje de programación (MDN, s.f. c) desarrollado con la finalidad de validar formularios implementados en HTML. Posteriormente, su funcionalidad se amplió a dotar de interactividad a cualquier página web estática. Este lenguaje fue diseñado para ejecutarse en navegadores web, por lo que no requiere ser compilado, ya que el buscador interpreta sucesivamente línea por línea las instrucciones contenidas en el archivo JavaScript.

Los documentos generados con JavaScript reciben el nombre de scripts y su estructura consta mínimamente de una clase pública con una función. Es considerado como un lenguaje de programación del lado del cliente, porque los scripts son interpretados sólo por el buscador del usuario que desea ingresar al sistema web.

JavaScript permite añadir interactividad al comportamiento de cualquier proyecto web, como mostrar animaciones, crear efectos y generar eventos, con el objetivo de mejorar la interacción entre el sistema y el usuario. Los eventos dentro del contexto de JavaScript son acontecimientos derivados de una acción generada por el usuario en la interfaz gráfica de la plataforma web.

Métodos para incorporar código JavaScript a archivos HTML.

- Interno: el código JavaScript se agrega directamente al archivo HTML a trabajar; las instrucciones se colocan dentro de la etiqueta `<script> </script>`, la cual puede estar posicionada dentro del bloque limitado por la etiqueta `<head> </head>` o por la marca `<body></body>`. El código JavaScript implementado de esta manera sólo afecta al documento de hipertexto en el que fue agregado.
- Externo: el código JavaScript se incorpora al archivo HTML por medio de una dirección URL dentro del bloque limitado por la etiqueta `<head> </head>` o por la marca `<body></body>`. La dirección hace referencia a la ubicación de un archivo externo de extensión `.js`, en el cual se encuentran todas las instrucciones que volverán interactivo al proyecto web. Esta forma de agregar código JavaScript permite trabajar con varios archivos HTML, así como de mejorar la productividad del sistema web.

Framework

Un marco de trabajo (del inglés *framework*) es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

Estos marcos de trabajo se emplean para ahorrar tiempo y esfuerzo en el desarrollo de aplicaciones, ya que proporcionan una estructura básica que se puede utilizar como punto de partida. Además, los frameworks también ofrecen soluciones a problemas comunes en el desarrollo de software, lo que significa que los desarrolladores pueden centrarse en las funcionalidades específicas de su aplicación en lugar de perder tiempo resolviendo problemas técnicos (<https://es.wikipedia.org/wiki/Framework>).

Figma

Plataforma web (Garcés, 2022) orientada al diseño de prototipos o maquetas de interfaces gráficas, tanto de aplicaciones móviles como de aplicaciones web. Esta herramienta está enfocada al trabajo colaborativo, por lo que facilita la edición del diseño en tiempo real, así como de visualizar los cambios que realiza cada integrante del equipo.

La particularidad más importante es que aloja los proyectos en la nube, permitiendo al usuario el acceso a estos archivos desde cualquier dispositivo con accesos a internet. Los recursos gráficos que maneja Figma son texto, formas, iconos, imágenes, elementos interactivos entre otros. Todos los proyectos desarrollados en Figma tienen la facultad de generar el código CSS del diseño, así como de ser compartidos a cualquier persona por medio de una URL.

Bootstrap

Framework que tiene como objetivo agilizar el desarrollo de la interfaz gráfica de una aplicación o página web (Urrutia, 2021). Permite crear interfaces que se puedan adaptar

automáticamente a las dimensiones del dispositivo donde se esté visualizando el sistema web. De esta manera facilita al usuario el acceso a los recursos contenidos en la aplicación. Este framework está compuesto por elementos de CSS y JAVASCRIPT los cuales permiten crear estructuras visuales limpias, interactivas e intuitivas para el usuario final. Ofrece elementos predefinidos como botones, menús desplegables, formularios, mensajes de alerta, carrusel de imágenes, listas desplegables, entre otros.

Es compatible con los siguientes navegadores: Chrome, Firefox, Edge, Safari, Opera

Bootstrap Icons

Extensión de bootstrap que proporciona de manera gratuita iconos SVG, con la posibilidad de personalizar su color y su tamaño sin perder calidad en el proceso.

Para acceder a los iconos es necesario agregar la extensión al proyecto, sin embargo, no es necesario emplear Bootstrap en el proyecto para poder utilizar la mencionada librería (<https://icons.getbootstrap.com/>).

Pexels

Repositorio web (Pexels, s. f.) que contiene imágenes y vídeos de alta calidad que se pueden utilizar de forma gratuita en cualquier tipo de proyecto. Todos los archivos contenidos en el repositorio son ingresados por los autores de estos. El buscador que contiene la plataforma, permite ingresar palabras clave para realizar el proceso de búsqueda, con el fin de filtrar las imágenes y vídeos contenidos hasta encontrar resultados relacionados.

Pruebas

De acuerdo con Sánchez Peño (2015) las pruebas permiten evaluar la seguridad y el funcionamiento del sistema o proyecto informático, también puntualiza la diferencia entre error, defecto y fallo. El error hace referencia a un evento originado por el usuario, el defecto es el resultado de una implementación errónea en el sistema y el fallo es el resultado de ejecutar el sistema con un defecto.

Pruebas de Caja Negra

Las pruebas de Caja Negra (Maquieira, 2023) proporcionan información acerca de la funcionalidad de un software, sirven para verificar que el sistema cumpla con las funciones declaradas en la etapa de levantamiento de requerimientos del proyecto realizado.

Las pruebas funcionales únicamente analizan los valores de entrada y de salida de un proceso, ignorando completamente la estructura lógica del mismo. Este tipo de pruebas permiten detectar defectos en la funcionalidad de este.

Para implementar el mencionado mecanismo de tester se requiere de un conjunto de datos con valores válidos e inválidos para el sistema, el cual será utilizado como información de entrada para generar diferentes respuestas por parte del proceso a analizar. Las respuestas generadas por el sistema representan parámetros que permitirán mejorar el funcionamiento del software.

Estrategias para implementar pruebas de caja negra

Partición de Equivalencia

Técnica que utiliza casos de prueba para encontrar fallas de funcionamiento en el sistema, este método es eficiente, sencillo y rápido de emplear (Gómez, 2023).

Procedimiento

1. Seccionar los datos de entrada en grupos de igual tamaño.
2. Generar un caso de prueba de cada conjunto de datos.
3. Emplear los casos de prueba obtenidos en el sistema.
4. Cada caso de prueba está enfocado a detectar un tipo específico de error.

Análisis de Valores Límites

Técnica que emplea el método de Partición de equivalencia y utiliza los valores límites de los datos de entrada para diseñar casos de prueba, con la finalidad de identificar fallas en un proceso (Gómez, 2023).

Procedimiento

1. Dividir los datos de entrada en particiones equivalentes.
2. Probar en el sistema las instancias contenidas en los conjuntos, utilizando los valores límites de la partición.

Tabla de Decisión

Método que emplea tablas de decisión para generar casos de prueba y encontrar errores en un sistema o proceso. Las tablas de decisión permiten representar gráficamente condiciones que determinan soluciones para un conjunto de datos de entrada (Gómez, 2023).

Las condiciones comprenden tres valores válidos, se cumple, no se cumple e indistinto. Las acciones comprenden dos valores válidos, se realiza o no se realiza.

Procedimiento

1. Definir condiciones.
2. Definir acciones factibles.

3. Definir alternativas de cada condición.
4. Forma tabla.

Pruebas de Caja Blanca

Las pruebas de caja blanca (Singureanu, 2023) se centran en verificar el funcionamiento de la lógica y el diseño del sistema implementado. Este tipo de pruebas permiten hallar errores de programación, seguridad y usabilidad, favoreciendo la calidad del producto, así como su mantenibilidad, flexibilidad y modularidad. Las pruebas de caja blanca se enfocan en cómo funciona el código para verificar el flujo de entrada y salida de los procesos que componen al sistema.

Para elaborar pruebas de caja blanca es necesario tener conocimientos de programación informática y sobre la estructura interna del software, por lo que generalmente son realizadas por desarrolladores o ingenieros de software.

Estrategia para implementar pruebas de caja blanca

Prueba del Camino Básico

Método que consiste en comprobar que todas las instrucciones del código de un programa se ejecuten por lo menos una vez, con la finalidad de medir la complejidad lógica de un diseño procedimental (Sánchez Peño, 2015).

Caminos independientes: caminos de ejecución que el sistema llevará a cabo por lo menos una vez.

Complejidad ciclomática: métrica de software que permite medir la complejidad lógica de un programa, permite definir el número de caminos independientes.

Formas de cálculo

1. Formula: $V(G) = a - n + 2$

donde: a = número de aristas y n = número de nodos

2. Formula $V(G) = P + 1$

donde:

P = número de nodos predicado-contenidos en el grafo (nodos de los cuales se desprenden dos aristas de él).

Grafo de flujo: representación gráfica del flujo de control del sistema a analizar.

Casos de prueba: conjunto de datos de prueba, condiciones de ejecución y resultados esperados para un proceso determinado.

1. Generar el diagrama de flujo correspondiente al programa a analizar
2. Generar el grafo de flujo correspondiente al diagrama de flujo
3. Calcular complejidad ciclomática
4. Determinar el conjunto básico de caminos independientes
5. Definir casos de prueba

Marco metodológico

Método Kanban

El método Kanban (Anderson, s.f.) se escogió como la técnica de organización predeterminada para llevar a cabo el presente proyecto, ya que la serie de fases que presenta permite controlar el avance de cada proceso a ejecutar; este método es flexible ante cualquier cambio que se deba aplicar al sistema y de manera regular se puede comprobar el avance del proyecto.

Kanban es un método visual utilizado para la gestión de proyectos, el cual ayuda a tener un rendimiento y ritmo de trabajo óptimo. Este método permite que la circulación de trabajo sea entendible para cualquier miembro del equipo, así como tener una buena organización entre sus elementos. Gracias a su forma de organización logra reducir retrasos, tener retroalimentación temprana y fomenta el cambio progresivo y la mejora continua, logrando así que el proyecto sea flexible a cambios (Castellano Lendínez, 2019).

1. Definir las tareas a desarrollar en tarjetas o notas adhesivas, las cuales representarán el flujo de trabajo.
2. Se crea un tablero con tres columnas, donde la primera columna contendrá las tareas pendientes, en la segunda columna se colocarán las tareas que se están ejecutando y, por último, la tercera columna engloba las tareas culminadas.

Este paso hace referencia a limitar la corriente de trabajo, ya que ningún elemento del equipo puede ejecutar otra tarea sin haber culminado las que se encuentran en la segunda columna del tablero.

3. Las tareas cambian de columna dependiendo de su estado, de esta manera se visualizará el flujo de trabajo.

El tablero permite administrar y mejorar la ejecución del proyecto, ya que destaca el estado de las tareas y muestra si se desarrollaron puntualmente o se acumularon.

Modelo Incremental

El modelo de desarrollo incremental es el ciclo de vida de desarrollo software en el cual un proyecto es descompuesto en una serie de incrementos, cada uno de los cuales suministra una porción de la funcionalidad respecto de la totalidad de los requisitos del proyecto. Los requisitos tienen asignada una prioridad y son entregados según el orden de prioridad en el incremento correspondiente. Se seleccionó como el modelo a seguir para el proceso de elaboración del presente proyecto, ya que es un modelo adaptable a cambios, facilita el manejo de riesgos, es conveniente cuando el personal no es suficiente para llevar a cabo la elaboración del software y permite visualizar de manera continua el crecimiento del sistema, gracias a las cortas iteraciones en las que se divide la implementación (Pressman, 2010).

Acorde con (Maida y Pacienza, 2015) el modelo por incremento consta de las siguientes fases (Figura 7):

1. Análisis: fase en la que se lleva a cabo la obtención y el análisis de los requerimientos, así como un estudio de viabilidad del proyecto.
2. Diseño: fase en la que se elabora el diseño de la arquitectura del sistema, centrándose en sus componentes como interfaces, base de datos etc.
3. Codificación: fase en la que se codifica la funcionalidad del software y se realizan pruebas unitarias.
4. Pruebas: fase en la que se realizan pruebas para detectar errores y asegurar el correcto funcionamiento del sistema implementado, para garantizar la calidad del software.



Figura 7 Modelo del ciclo de vida Incremental.

Capítulo 4

Implementación

En el presente capítulo se describe cómo se realizó cada etapa del modelo incremental para el diseño y desarrollo del sistema web.

Incremento 1

Análisis

Técnica para la Obtención de Requerimientos del Proyecto

El levantamiento de requerimientos del presente proyecto se realizó utilizando la técnica de entrevista con el cliente, ya que permite hacer preguntas abiertas con las que se puede encontrar información detallada del sistema y preguntas cerradas con las que se puede confirmar las características y funciones del proyecto. La mencionada técnica favorece la comunicación entre cliente e implementador, logrando reducir errores o cambios en el desarrollo del sistema.

En la Tabla se describen los requerimientos funcionales identificados para el desarrollo del sistema web.

Tabla 2 Requerimientos Funcionales

Código	Nombre	Descripción
RF01	Iniciar sesión	El usuario coordinador ingresará al sistema por medio de nombre de usuario y contraseña.
RF02	Registrar alumno	El sistema debe permitir el registro del alumno de acuerdo a su generación, así como almacenar su información personal.
RF03	Registrar docente	El sistema debe permitir el registro del docente como miembro de la planta básica y como colaborador, así como, almacenar su información personal.
RF04	Crear expediente de alumno	El sistema debe permitir almacenar documentos del alumno en formato PDF.
RF05	Modificar información de alumnos	El sistema debe permitir modificar la información contenida en la BD de los alumnos.
RF06	Modificar información de docentes	El sistema debe permitir modificar la información contenida en la BD de los docentes.
RF07	Modificar información de coloquios	El sistema debe permitir modificar la información contenida en la BD de los docentes.
RF08	Eliminar alumno	El sistema debe permitir eliminar toda la información relacionada a un alumno.
RF09	Eliminar docente	El sistema debe permitir eliminar toda la información relacionada a un docente.
RF10	Asignar cursos a alumnos	El sistema debe permitir asignar clases al alumno de acuerdo al semestre inscrito.
RF11	Asignar cursos a docentes	El sistema debe permitir asignar clases al docente cada periodo escolar.
RF12	Agendar coloquio	El sistema debe permitir agendar la información correspondiente a los coloquios que se llevarán a cabo en el semestre (fecha, lugar, horarios, orden de presentaciones, rúbricas de evaluación, asignación de evaluadores externos).
RF13	Exportar información	El sistema debe permitir exportar en formato .xlsx la información almacenada en la base de datos
RF14	Imprimir información	El sistema debe permitir imprimir la información almacenada en la base de datos.
RF15	Consultar información de alumnos	El sistema debe permitir consultar la información de los alumnos contenida en la base de datos .
RF16	Consultar información de docentes	El sistema debe permitir consultar la información de los alumnos contenida en la base de datos .
RF17	Consultar coloquios	El sistema debe permitir consultar la información de los coloquios agendados en el sistema.
RF18	Consultar información de cursos	El sistema debe permitir consultar la información de los cursos.
RF19	Consultar información de asignación	El sistema debe permitir consultar la información de alumnos y docentes asignada a un curso.

Diseño

Diagrama de Casos de Uso del Sistema Web

Diagrama que permite visualizar el comportamiento esperado del sistema web para gestión académica de un posgrado ante las posibles acciones que el usuario coordinador ejecute. Las funciones que conforman al sistema son: iniciar sesión, registrar alumnos y docentes, modificar información de alumnos y docentes, crear expediente de alumno, (almacenar documentos), eliminar alumnos y docentes, asignar cursos a alumnos y a docentes, agendar coloquios, consultar información de alumnos, docentes, coloquios, cursos y asignaciones, y exportar e imprimir información contenida en el sistema (Figura 8).

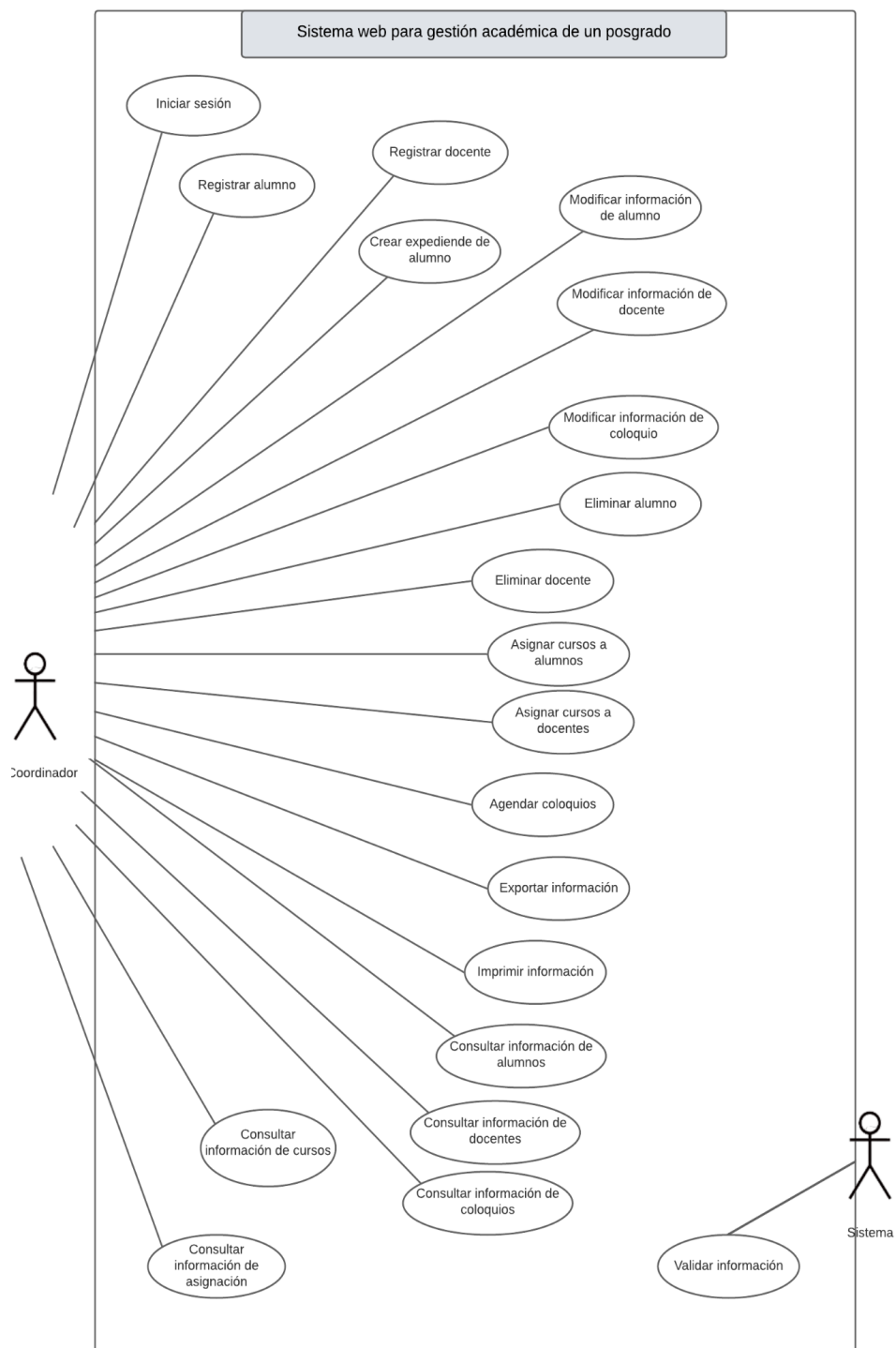


Figura 8 Diagrama de casos de uso del sistema web.

Descripción de Casos de Uso

En las tablas 4 a 22 se describe cada caso de uso. Un caso de uso contiene una descripción textual de todas las maneras que los actores previstos podrían trabajar con el software o el sistema.

Iniciar sesión

Caso de uso que describe la serie de pasos que debe realizar el usuario final para iniciar sesión en el sistema web, de igual forma especifica las precondiciones necesarias para realizar la mencionada acción.

Asimismo, muestra el flujo alternativo que el sistema sigue si los datos de autenticación ingresados no se encuentran registrados en el sistema.

Tabla 3 Descripción del caso de uso Iniciar Sesión

Sistema para gestión académica de un posgrado Componente: Iniciar sesión Fecha: 29 de noviembre del 2022	
Caso de uso	Iniciar sesión
Objetivo	Permitir al coordinador de posgrados ingresar al sistema web
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema
Flujo Normal	1. El sistema muestra formulario de inicio de sesión 2.El coordinador ingresa nombre de usuario y contraseña 3. El sistema valida la existencia del usuario 4. El sistema inicia sesión del usuario 5.El sistema muestra menú de opciones
Flujo Alternativo	3. El sistema muestra en pantalla un mensaje de error en la autenticación
Post-condiciones	

Registrar alumno

Caso de uso que indica el flujo que debe realizar el usuario final para registrar la información correspondiente a un estudiante en la plataforma web. También precisa los actores con quien interactúa y las precondiciones necesarias para realizar la mencionada acción

De igual forma presenta el comportamiento alternativo que el sistema sigue si la información obtenida no se guarda en el sistema.

Tabla 4 Descripción del caso de uso Registrar Alumno

Sistema para gestión académica de un posgrado	
Componente: Registrar alumno	
Fecha: 29 de noviembre del 2022	
Caso de uso	Registrar alumno
Objetivo	Permitir al coordinador de posgrados registrar la información personal de un alumno
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de alumnos 2. El sistema muestra lista de opciones 3. El coordinador selecciona la opción "Registrar alumno" 4. El sistema muestra el formulario de registro 5. El coordinador ingresa la información requerida 6. El sistema verifica la información obtenida 7. El sistema guarda la información registrada
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Registrar docente

Caso de uso que especifica las acciones que el usuario final debe efectuar para registrar información relacionada con un docente en el sistema web. Asimismo, especifica los actores con los que interactúa y las precondiciones y postcondiciones del proceso. Igualmente muestra el flujo alternativo que el sistema sigue si la información ingresada no se guarda en el sistema.

Tabla 5 Descripción del caso de uso Registrar Docente

Sistema para gestión académica de un posgrado	
Componente: Registrar docente	
Fecha: 29 de noviembre del 2022	
Caso de uso	Registrar docente
Objetivo	Permitir al coordinador de posgrados registrar la información personal de un docente
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2. El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de docentes 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Registrar docente" 4. El sistema muestra el formulario de registro 5. El coordinador ingresa la información requerida 6. El sistema verifica la información obtenida 7. El sistema guarda la información registrada
Flujo Alternativo	7. El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Crear expediente de alumno

Caso de uso que describe las actividades que el usuario final debe llevar a cabo para almacenar documentos correspondientes a un estudiante en el sistema web. Muestra el flujo alternativo que el sistema sigue si los documentos ingresados no se almacenan correctamente en el sistema.

Tabla 6 Descripción del caso de uso Crear Expediente de Alumno

Sistema para gestión académica de un posgrado	
Componente: Crear expediente de alumno	
Fecha: 29 de noviembre del 2022	
Caso de uso	Crear expediente de alumno
Objetivo	Permite almacenar documentos en formato PDF de un alumno.
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Precondiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de alumnos 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Crear expediente " 4.El sistema despliega el formulario correspondiente para subir los documentos PDF 5.El coordinador ingresa los archivos PDF 6.El sistema verifica la existencia del usuario estudiante 7. El sistema guarda los archivos
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Precondiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Modificar información de alumno

Caso de uso que indica la serie de pasos que debe hacer el usuario final para modificar información relacionada con un estudiante registrado en la plataforma web. Plantea las precondiciones y postcondiciones de la mencionada operación y precisa los actores con quien interactúa. De igual forma presenta el flujo alternativo que el sistema sigue si la información recibida no se efectúa correctamente en el sistema.

Tabla 7 Descripción del caso de uso Modificar Información de Alumno

Sistema para gestión académica de un posgrado	
Componente: Modificar información de alumno	
Fecha: 29 de noviembre del 2022	
Caso de uso	Modificar información de alumno
Objetivo	Permitir al coordinador de posgrados modificar la información relacionada con un estudiante.
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de alumnos 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Modificar información de alumno" 4. El sistema despliega un formulario para la modificación de información del alumno 5. El coordinador ingresa la información requerida 6. El sistema verifica los datos ingresados 7. El sistema modifica la información del alumno especificado
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Modificar información de docente

Caso de uso que expone la serie de pasos que debe hacer el usuario final para modificar información relacionada con un estudiante registrado en la plataforma web. Plantea las precondiciones y postcondiciones de la mencionada operación y precisa los actores con quien interactúa. De igual forma presenta el flujo alternativo que el sistema sigue si la modificación indicada no se efectúa correctamente en el sistema.

Tabla 8 Descripción del caso de uso Modificar Información de Docente

Sistema para gestión académica de un posgrado	
Componente: Modificar información de docente	
Fecha: 29 de noviembre del 2022	
Caso de uso	Modificar información de docente
Objetivo	Permitir al coordinador de posgrados modificar la información relacionada con un docente
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2. El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de docentes 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Modificar información de docente" 4. El sistema despliega un formulario para la modificación de información del docente 5. El coordinador ingresa la información requerida 6. El sistema verifica los datos ingresados 7. El sistema modifica la información del docente especificado
Flujo Alternativo	7. El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Modificar información de coloquio

Caso de uso que muestra las acciones que debe efectuar el usuario final para modificar información relacionada con un coloquio registrado en la plataforma web. Describe los actores con los que interactúa y las precondiciones y postcondiciones de la mencionada acción. De igual forma presenta la serie de pasos alternativos que el sistema sigue si la modificación indicada no se efectúa correctamente en el sistema.

Tabla 9 Descripción del caso de uso Modificar Información de Coloquio

Sistema para gestión académica de un posgrado	
Componente: Modificar información de coloquio	
Fecha: 29 de noviembre del 2022	
Caso de uso	Modificar información de coloquio
Objetivo	Permitir al coordinador de posgrados modificar la información relacionada con un coloquio registrado
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de coloquios 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Modificar coloquio" 4. El sistema despliega un formulario para la modificación de información del coloquio 5. El coordinador ingresa la información requerida 6. El sistema verifica los datos ingresados 7. El sistema modifica la información del coloquio especificado
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Eliminar alumno

Caso de uso que describe las actividades que el usuario final debe efectuar para eliminar información relacionada con un estudiante registrado en la plataforma web. Plantea las precondiciones y postcondiciones de la mencionada operación y precisa los actores con quien interactúa. De igual forma presenta el flujo alternativo que el sistema sigue si la eliminación indicada no se ejecuta correctamente en el sistema.

Tabla 10 Descripción del caso de uso Eliminar Alumno

Sistema para gestión académica de un posgrado	
Componente: Eliminar alumno	
Fecha: 29 de noviembre del 2022	
Caso de uso	Eliminar alumno
Objetivo	Permitir al coordinador de posgrados eliminar toda la información relacionada con un estudiante.
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de alumnos 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Eliminar alumno" 4. El sistema despliega una caja de texto 5. El coordinador ingresa la matrícula universitaria del estudiante a eliminar 6. El sistema verifica la existencia del usuario a eliminar 7.El sistema elimina toda la información asociada a la matrícula ingresada
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Eliminar docente

Caso de uso que expone la serie de pasos que debe hacer el usuario final para eliminar información relacionada con un docente registrado en la plataforma web. Plantea las precondiciones y postcondiciones de la mencionada operación y precisa los actores con quien interactúa. De igual forma presenta el flujo alternativo que el sistema sigue si la eliminación indicada no se lleva a cabo correctamente en el sistema.

Tabla 11 Descripción del caso de uso Eliminar Docente

Sistema para gestión académica de un posgrado	
Componente: Eliminar docente	
Fecha: 29 de noviembre del 2022	
Caso de uso	Eliminar docente
Objetivo	Permitir al coordinador de posgrados eliminar toda la información relacionada con un docente
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de docentes 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Eliminar docente" 4. El sistema despliega una caja de texto 5. El coordinador ingresa el código de trabajador del docente a eliminar 6. El sistema verifica la existencia del usuario a eliminar 7.El sistema elimina toda la información asociada a la matrícula ingresada
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Asignar cursos a alumnos

Caso de uso que describe la serie de pasos que el usuario final debe realizar para inscribir alumnos a un curso utilizando la plataforma web. Plantea las precondiciones y postcondiciones de la mencionada operación y precisa los actores con quien interactúa. De igual forma presenta el flujo alternativo que el sistema sigue si el registro de información no se ejecuta correctamente en el sistema.

Tabla 12 Descripción del caso de uso Asignar Cursos a Alumnos

Sistema para gestión académica de un posgrado	
Componente: Asignar cursos a alumnos	
Fecha: 29 de noviembre del 2022	
Caso de uso	Asignar cursos a alumnos
Objetivo	Permitir al coordinador de posgrados asignar clases al alumno de acuerdo con el semestre correspondiente a su inscripción
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de cursos 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Asignar curso a alumno" 4. El sistema muestra el formulario de asignación 5. El coordinador ingresa la información requerida 6. El sistema verifica la matrícula del alumno 7. El sistema guarda la información registrada
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Asignar cursos a docentes

Caso de uso que presenta el flujo que el usuario final debe realizar para asignar docentes a un curso por medio de la plataforma web. Describe las precondiciones y postcondiciones de la mencionada acción y presenta el comportamiento alternativo que el sistema sigue si el registro de información no se ejecuta correctamente en el sistema.

Tabla 13 Descripción del caso de uso Asignar Cursos a Docentes

Sistema para gestión académica de un posgrado	
Componente: Asignar cursos a docentes	
Fecha: 29 de noviembre del 2022	
Caso de uso	Asignar cursos a docentes
Objetivo	Permitir al coordinador de posgrados asignar clases al docente de acuerdo con el semestre correspondiente a su inscripción
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2. El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de cursos 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Asignar curso a docente" 4. El sistema muestra el buscador de cursos 5. El coordinador ingresa el nombre del curso a asignar 6. El sistema verifica la información obtenida 7. El sistema despliega los requisitos y características del curso ingresado 8. El coordinador selecciona el nombre del curso 9. El sistema redirige al coordinador a una página adyacente 10. El sistema muestra el formulario de asignación 11. El coordinador ingresa la información requerida 12. El sistema verifica la información obtenida 13. El sistema guarda la información registrada
Flujo Alternativo	7. El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Agendar coloquio

Caso de uso que muestra las acciones que debe efectuar el usuario final para registrar información relacionada con un coloquio en la plataforma web. Describe los actores con los que interactúa y las precondiciones y postcondiciones de la mencionada acción. De igual forma presenta la serie de pasos alternativos que el sistema sigue si la información obtenida no se guarda en el sistema.

Tabla 14 Descripción del caso de uso Agendar Coloquio

Sistema para gestión académica de un posgrado	
Componente: Agendar coloquio	
Fecha: 29 de noviembre del 2022	
Caso de uso	Agendar coloquio
Objetivo	Permitir al coordinador de posgrados agendar la información correspondiente a los coloquios que se llevarán a cabo en el semestre
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de coloquios 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Agendar coloquio" 4. El sistema muestra un formulario para registrar la información del coloquio 5. El coordinador ingresa la información requerida 6. El sistema verifica la información ingresada 7. El sistema guarda la información registrada
Flujo Alternativo	7.El sistema muestra en pantalla un mensaje de error en la operación
Post-condiciones	El sistema despliega en pantalla un mensaje de confirmación de la operación

Exportar información

Caso de uso que muestra las actividades que debe ejecutar el usuario final para exportar información contenida en la plataforma web. Describe los actores con los que interactúa y las precondiciones y postcondiciones de la mencionada acción. De igual forma presenta la serie de pasos alternativos que el sistema sigue si el proceso de exportación es cancelado.

Tabla 15 Descripción del caso de uso Exportar Información

Sistema para gestión académica de un posgrado	
Componente: Exportar información	
Fecha: 29 de noviembre del 2022	
Caso de uso	Exportar información
Objetivo	Permitir al coordinador de posgrados exportar información contenida en el sistema en formato .xlsx
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2. El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la tabla de datos a exportar 2. El sistema muestra la información y la sección de botones 3. El coordinador selecciona la opción "Exportar información" 4. El sistema despliega el formulario para seleccionar la información a exportar 5. El coordinador selecciona información 6. El sistema exporta el archivo en formato .xlsx con toda la información seleccionada
Flujo Alternativo	7. El coordinador rechaza la descarga del documento exportado
Post-condiciones	El sistema exporta la información seleccionada

Imprimir Información

Caso de uso que describe las acciones que el usuario final debe hacer para imprimir información contenida en la plataforma web. Describe los actores con los que interactúa y las precondiciones y postcondiciones de la mencionada acción. También presenta el flujo alternativo que el sistema sigue si el proceso de impresión es cancelado.

Tabla 16 Descripción del caso de uso Imprimir Información

Sistema para gestión académica de un posgrado	
Componente: Imprimir información	
Fecha: 29 de noviembre del 2022	
Caso de uso	Imprimir información
Objetivo	Permitir al coordinador de posgrados imprimir información contenida en el sistema
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la tabla de datos a imprimir 2. El sistema muestra la información y la sección de botones 3. El coordinador selecciona la opción "Imprimir" 4. El sistema muestra el formulario para seleccionar la información a imprimir 5. El coordinador selecciona información 6. El sistema muestra la visualización del documento a imprimir 7.El coordinador acepta la visualización de la información a imprimir
Flujo Alternativo	7. El coordinador rechaza la visualización de la información a imprimir
Post-condiciones	El sistema imprime la información seleccionada

Consultar información de alumnos

Caso de uso que describe la serie de pasos que debe realizar el usuario final para consultar información correspondiente a los alumnos registrados en el sistema web. De igual forma especifica las precondiciones necesarias para realizar la mencionada acción y los actores con lo que interactúa directamente.

Tabla 17 Descripción del caso de uso Consultar Información de Alumno

Sistema para gestión académica de un posgrado	
Componente: Consultar información de alumno	
Fecha: 29 de noviembre del 2022	
Caso de uso	Consultar información de alumno
Objetivo	Permitir al coordinador de posgrados consultar la información almacenada de todos los alumnos registrados en el sistema
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de alumnos 2. El sistema muestra lista de opciones 3. El coordinador selecciona la opción "Alumnos registrados" 4. El sistema despliega toda la información de los alumnos registrados 5. El coordinador visualiza la información
Flujo Alternativo	
Post-condiciones	
Extensiones	1. Imprimir información 2. Exportar información

Consultar información de docentes

Caso de uso que describe la serie de pasos que debe realizar el usuario final para consultar información correspondiente a los docentes registrados en el sistema web. Asimismo, expone los actores con quienes interactúa y las precondiciones necesarias para realizar la mencionada operación.

Tabla 18 Descripción del caso de uso Consultar Información de Docentes

Sistema para gestión académica de un posgrado	
Componente: Consultar información de docentes	
Fecha: 29 de noviembre del 2022	
Caso de uso	Consultar información de docentes
Objetivo	Permitir al coordinador de posgrados consultar la información almacenada de todos los docentes registrados en el sistema
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2. El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de docentes 2. El sistema muestra lista de opciones 3. El coordinador selecciona la opción "Docentes registrados" 4. El sistema despliega toda la información de los docentes registrados 5. El coordinador visualiza la información
Flujo Alternativo	
Post-condiciones	
Extensiones	1. Imprimir información 2. Exportar información

Consultar información de coloquios

Caso de uso que muestra la serie de pasos que debe realizar el usuario final para consultar información correspondiente a los coloquios registrados en el sistema web. También expone las precondiciones necesarias para realizar la mencionada acción y los actores con lo que interactúa directamente.

Tabla 19 Descripción del caso de uso Consultar Información de Coloquios

Sistema para gestión académica de un posgrado	
Componente: Consultar información de coloquios	
Fecha: 29 de noviembre del 2022	
Caso de uso	Consultar información de coloquios
Objetivo	Permitir al coordinador de posgrados consultar la información almacenada de todos los coloquios registrados en el sistema
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de docentes 2. El sistema muestra lista de opciones 3. El coordinador selecciona la opción "Coloquios registrados" 4. El sistema despliega toda la información de los coloquios registrados 5. El coordinador visualiza la información
Flujo Alternativo	
Post-condiciones	
Extensiones	1. Imprimir información 2. Exportar información

Consultar información de cursos

Caso de uso que describe la serie de pasos que debe realizar el usuario final para consultar información correspondiente a los cursos almacenados en el sistema web. De igual forma especifica las precondiciones necesarias para realizar la mencionada acción y los actores con lo que interactúa directamente.

Tabla 20 Descripción del caso de uso Consultar Información de Cursos

Sistema para gestión académica de un posgrado	
Componente: Consultar información de cursos	
Fecha: 29 de noviembre del 2022	
Caso de uso	Consultar información de cursos
Objetivo	Permitir al coordinador de posgrados consultar la información de todos los cursos contenidos en el sistema
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de cursos 2. El sistema muestra lista de opciones 3. El coordinador selecciona la opción "Lista de cursos " 4. El sistema despliega toda la información de los cursos almacenados. 5. El coordinador visualiza la información
Flujo Alternativo	
Post-condiciones	
Extensiones	1. Imprimir información 2. Exportar información

Consultar información de asignación

Caso de uso que indica el flujo que debe realizar el usuario final para consultar la información de docentes y alumnos relacionada con un curso. También precisa los actores con quien interactúa y las precondiciones necesarias para realizar la mencionada acción

Tabla 21 Descripción del caso de uso Consultar Información de Asignación

Sistema para gestión académica de un posgrado	
Componente: Consultar información de asignación	
Fecha: 29 de noviembre del 2022	
Caso de uso	Consultar información de coloquios
Objetivo	Permitir al coordinador de posgrados consultar la información relacionada con la asignación de un curso
Autor	Alexandra Bautista Otero
Actor principal	Coordinador
Actor secundario	Sistema web
Pre-condiciones	1. El coordinador debe estar registrado en el sistema 2.El coordinador debe iniciar sesión en el sistema
Flujo Normal	1. El coordinador ingresa a la sección de cursos 2. El sistema muestra la lista de opciones 3. El coordinador selecciona la opción "Consultar información de asignación" 4. El sistema despliega toda la información de la asignación 5. El coordinador visualiza la información
Flujo Alternativo	
Post-condiciones	
Extensiones	1. Imprimir información 2. Exportar información

Diagramas de Secuencia

Recordando que un diagrama de secuencia UML muestra cómo interactúa un conjunto de objetos en un proceso a lo largo del tiempo y muestra los mensajes que pasan entre los participantes y los objetos del sistema y el orden en que se producen, a continuación, se presentan los diagramas necesarios para el diseño del sistema web (Figura 9 a 24).

Registrar Alumno

Diagrama de secuencia que muestra a detalle el proceso y la interacción entre el usuario final, el sistema, el formulario de registro y la base de datos para registrar información correspondiente a un alumno en el sistema web. De la misma forma muestra el procedimiento que se lleva a cabo en los diferentes componentes si la información no se registra exitosamente.

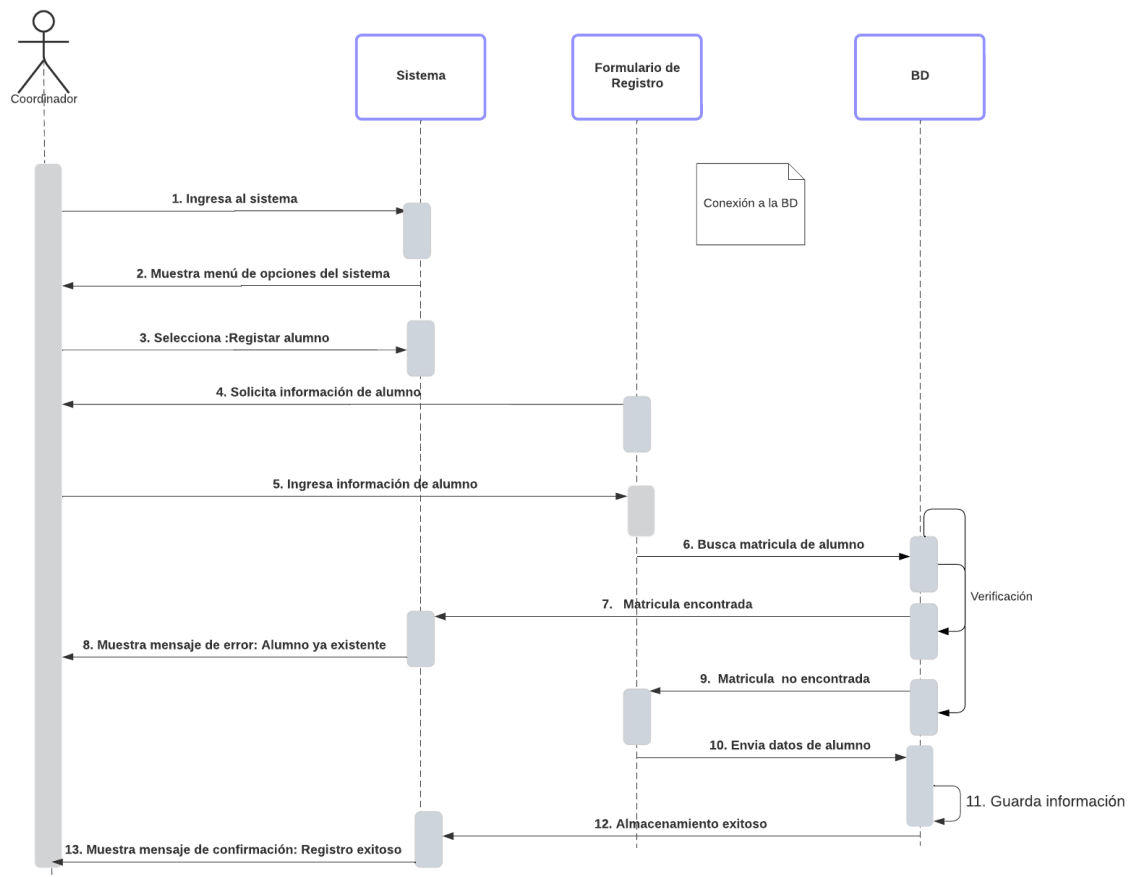


Figura 9 Diagrama de secuencia Registrar Alumno

Eliminar Alumno

Diagrama de secuencia que describe detalladamente la interacción entre el usuario final, el sistema, el formulario de eliminación y la base de datos para eliminar a un alumno del sistema web. También muestra el flujo de desarrollo que realiza cada componente si la acción no se completa exitosamente.

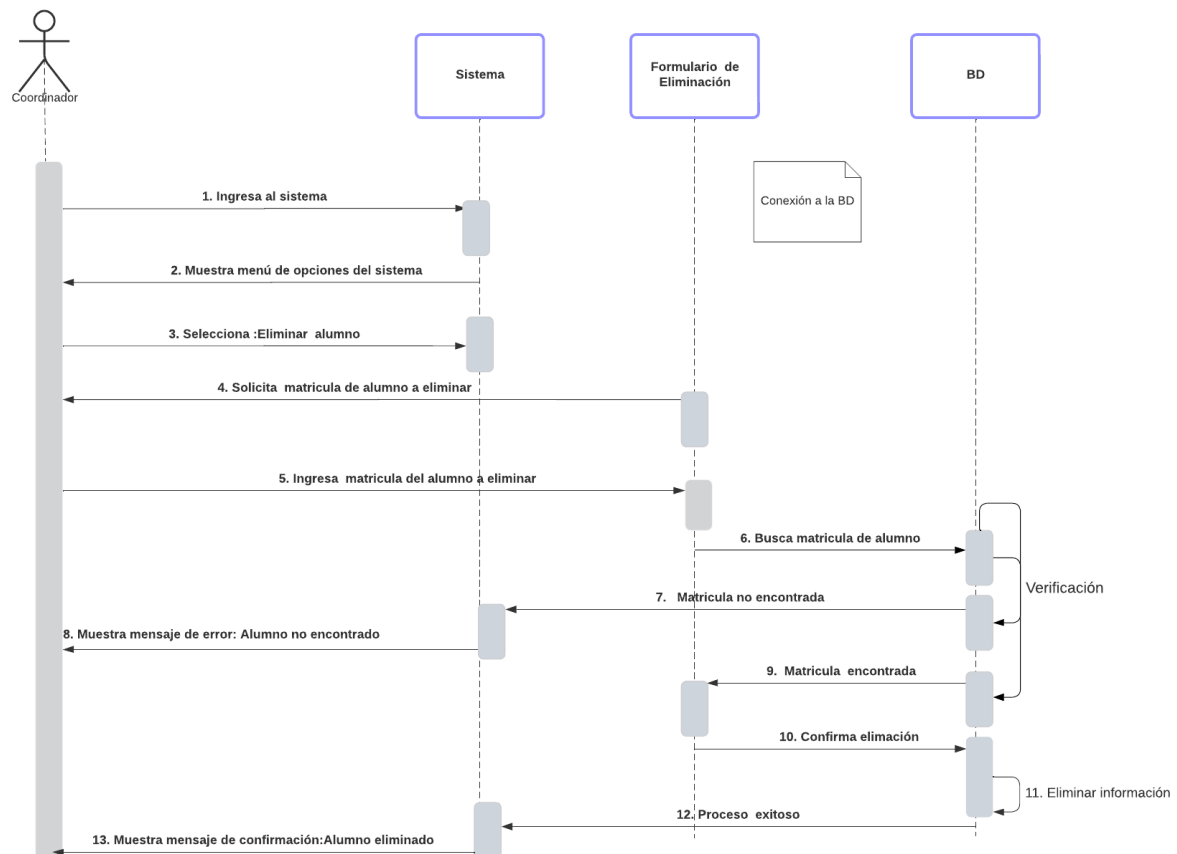


Figura 10 Diagrama de secuencia Eliminar Alumno

Modificar Alumno

Diagrama de secuencia que precisa la interacción entre el usuario final, el sistema, el formulario de modificación y la base de datos para modificar información relacionada con un alumno. Igualmente expone la comunicación que se lleva a cabo entre los diferentes componentes si la información no se modifica exitosamente.

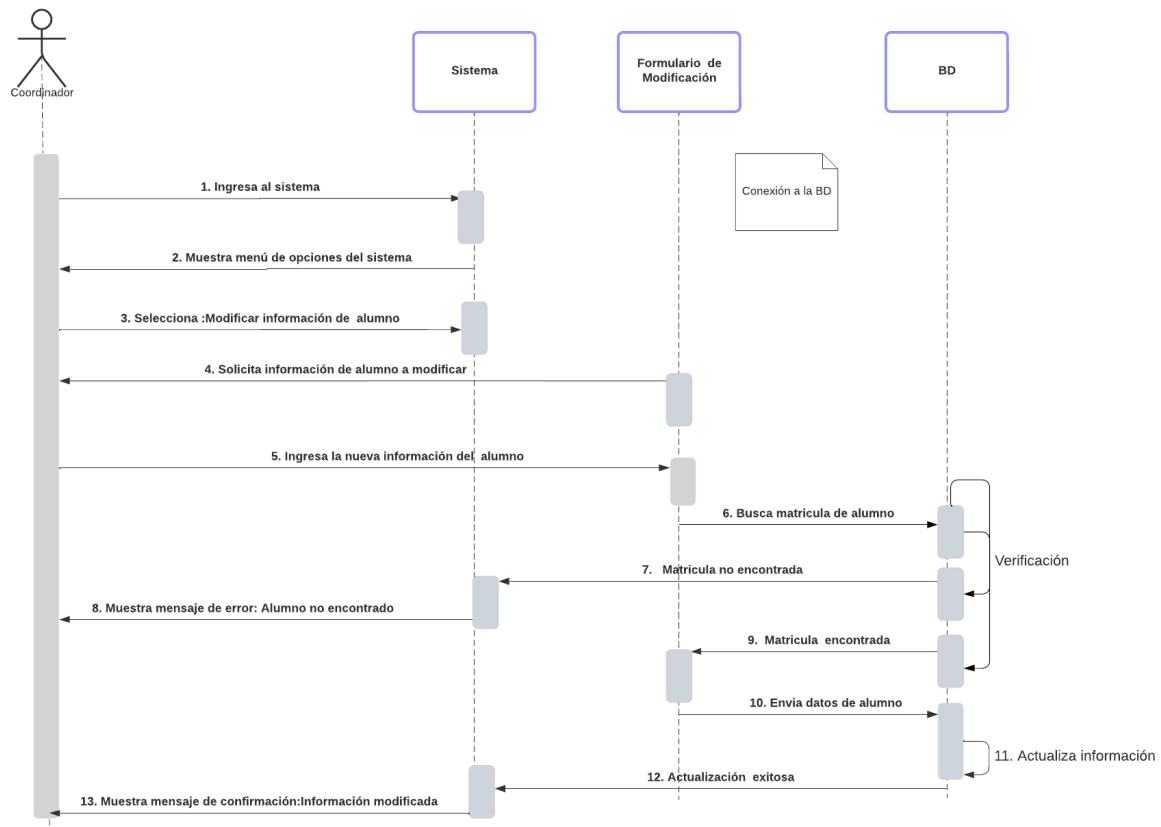


Figura 11 Diagrama de secuencia Modificar Alumno

Crear Expediente de Alumno

Diagrama de secuencia que especifica la comunicación entre el usuario final, el sistema, el formulario de registro y la base de datos para registrar archivos correspondientes a un alumno almacenado en el sistema. De igual forma muestra la interacción entre los diferentes componentes si los archivos no se registran en el sistema.

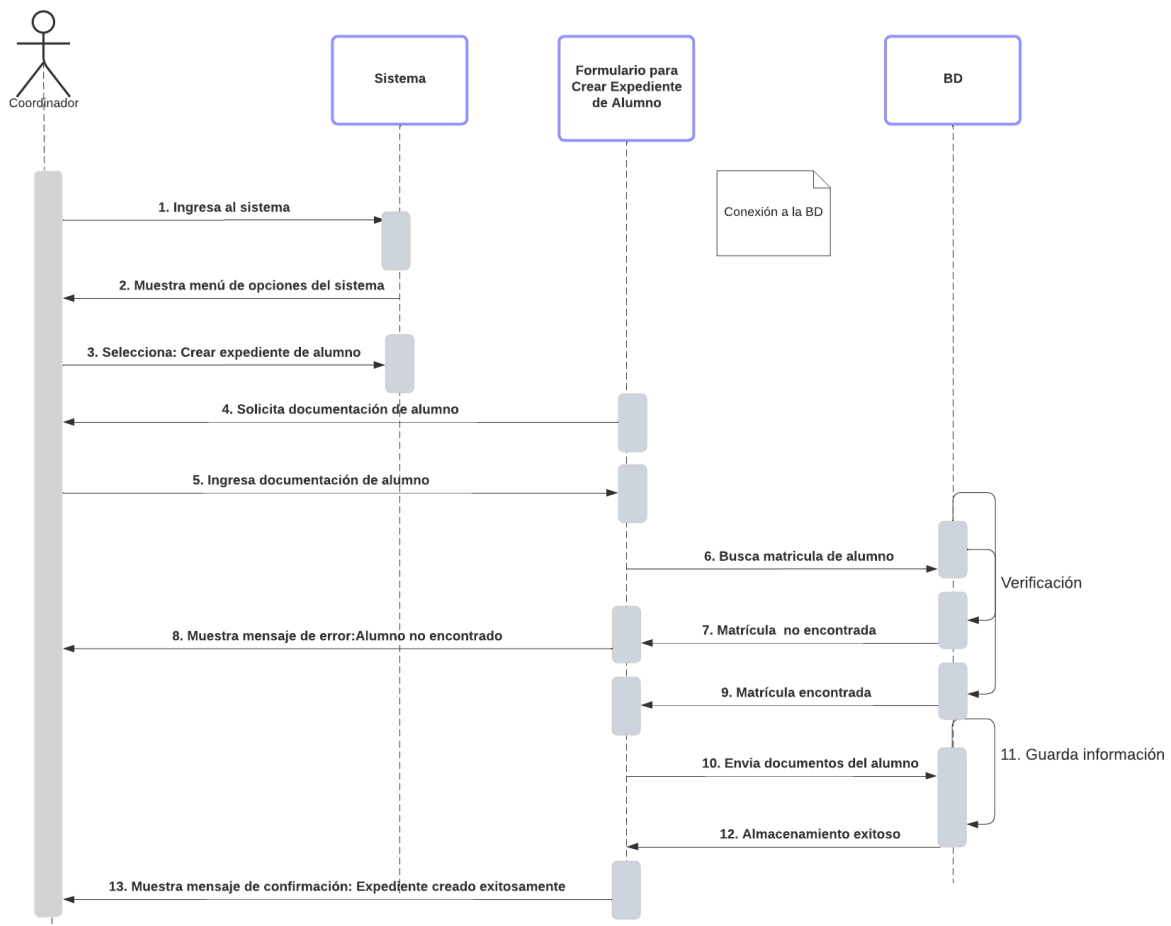


Figura 12 Diagrama de secuencia Crear Expediente de Alumno

Consultar Alumnos Registrados

Diagrama de secuencia que describe la interacción entre el usuario final, el sistema, la interfaz de alumnos registrados y la base de datos para consultar toda la información relacionada con los alumnos registrados en el sistema.

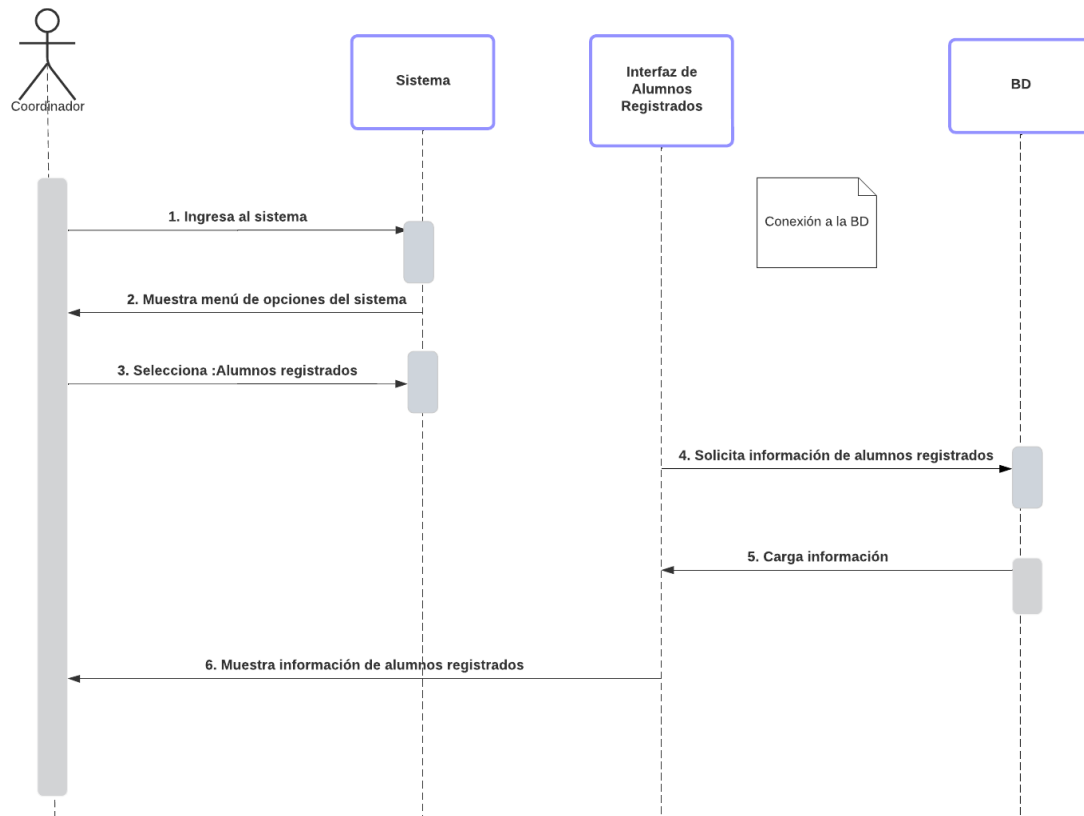


Figura 13 Diagrama de secuencia Consultar Alumnos Registrados

Registrar Docente

Diagrama de secuencia que muestra la comunicación entre el usuario final, el sistema, el formulario de registro y la base de datos para registrar en el sistema web la información pertinente a un docente. Asimismo, presenta el procedimiento que se lleva a cabo en los diferentes componentes si la información no se registra exitosamente.

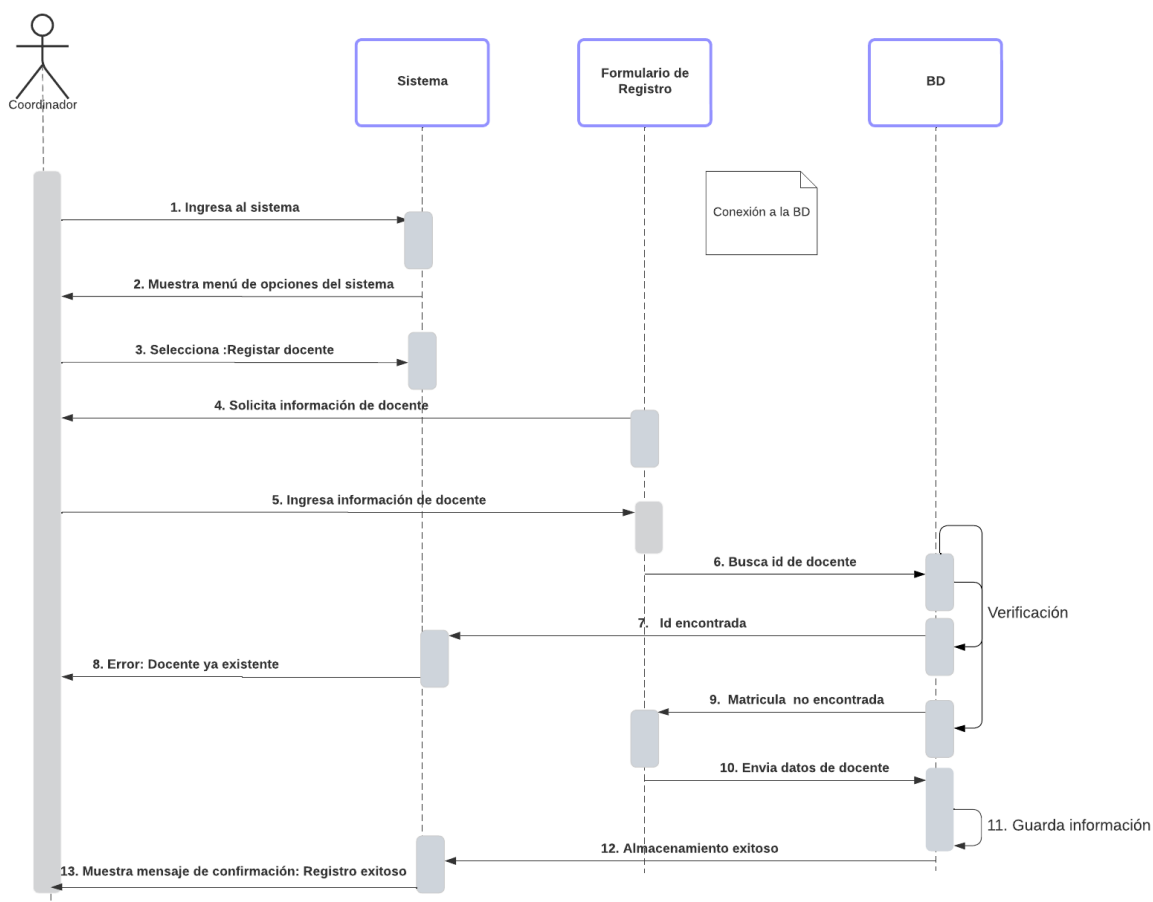


Figura 14 Diagrama de secuencia Registrar Docente

Eliminar Docente

Diagrama de secuencia que describe detalladamente la interacción entre el usuario final, el sistema, el formulario de eliminación y la base de datos para eliminar a un docente del sistema web. También muestra el flujo de desarrollo que realiza cada componente si la acción no se completa exitosamente.

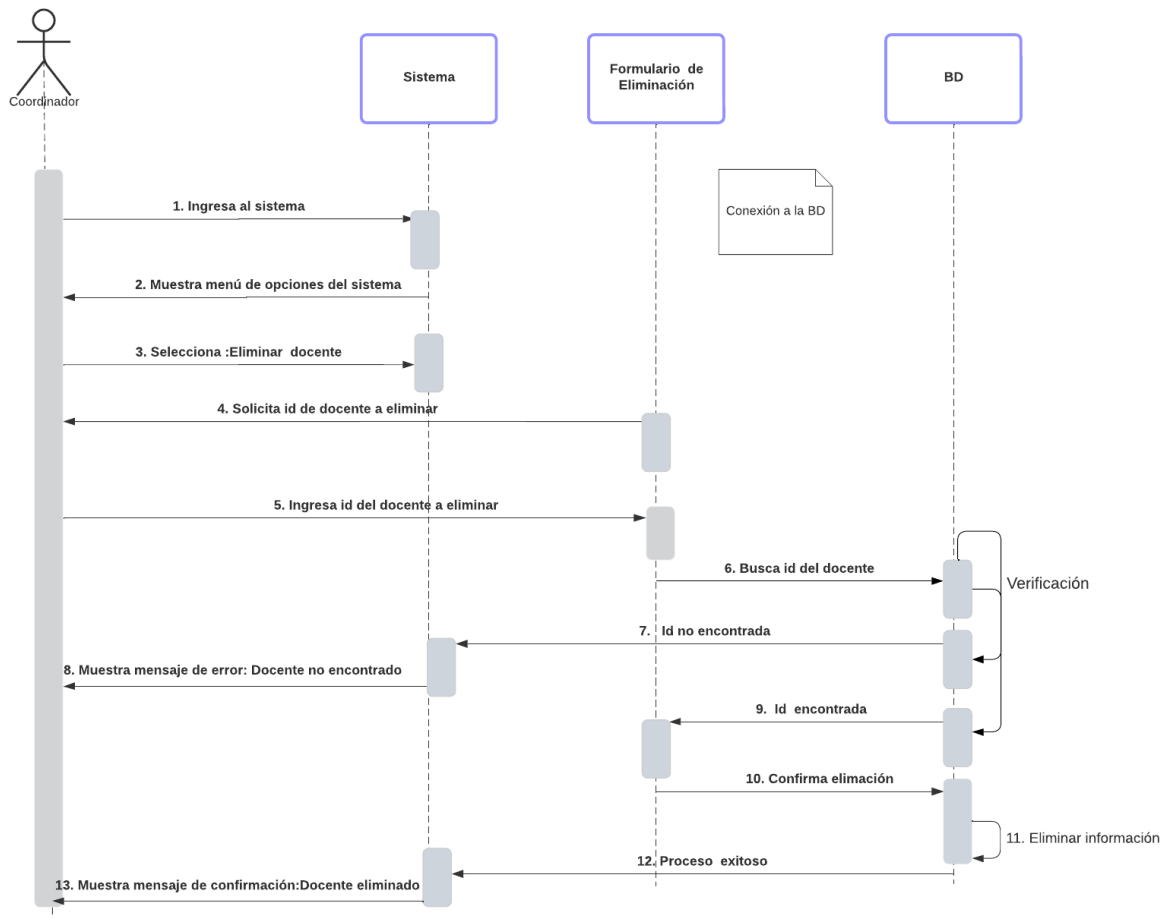


Figura 15 Diagrama de secuencia Eliminar Docente

Modificar Docente

Diagrama de secuencia que precisa la interacción entre el usuario final, el sistema, el formulario de modificación y la base de datos para modificar información relacionada con un docente. Igualmente expone la comunicación que se lleva a cabo entre los diferentes componentes si la información no se modifica exitosamente.

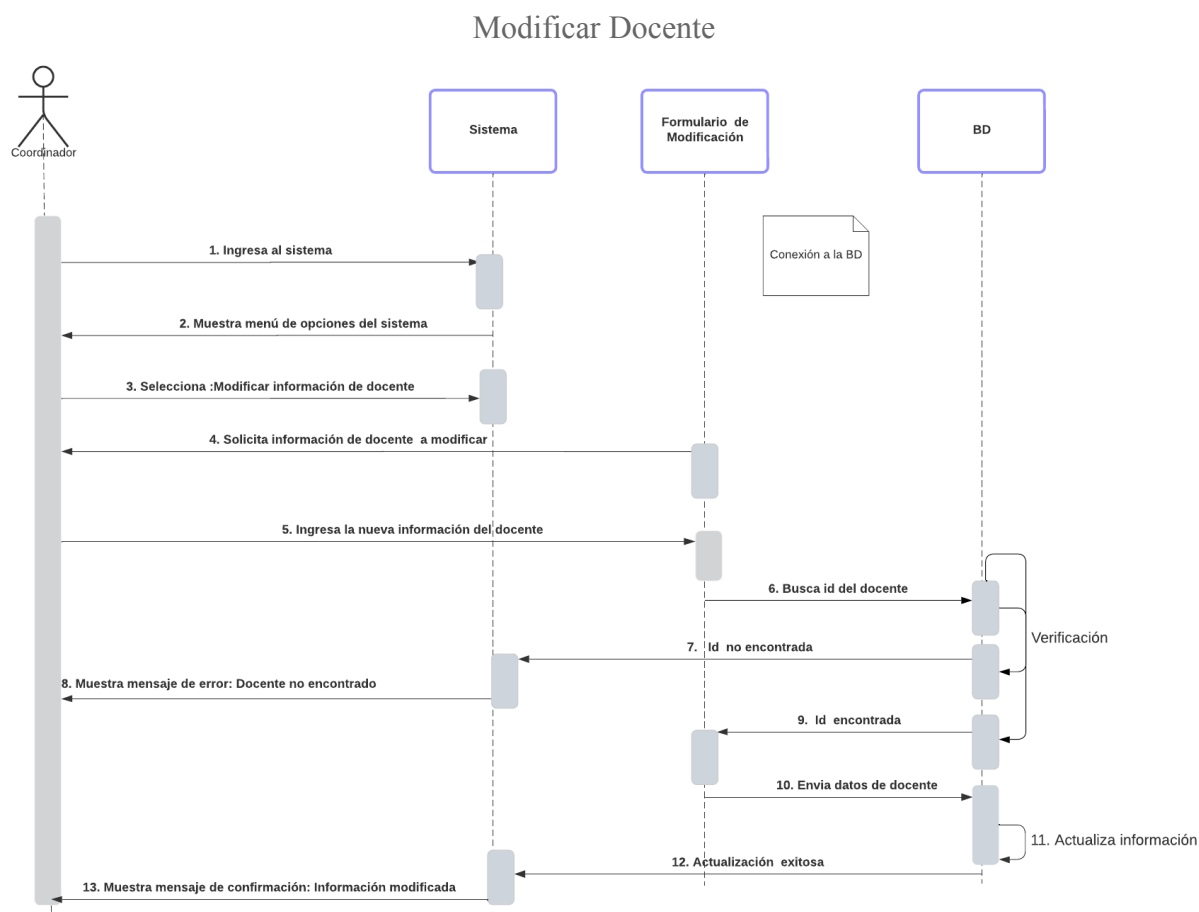


Figura 16 Diagrama de secuencia Modificar Docente

Consultar Docentes Registrados

Diagrama de secuencia que describe la interacción entre el usuario final, el sistema, la interfaz de docentes registrados y la base de datos para consultar toda la información relacionada con los docentes registrados en el sistema.

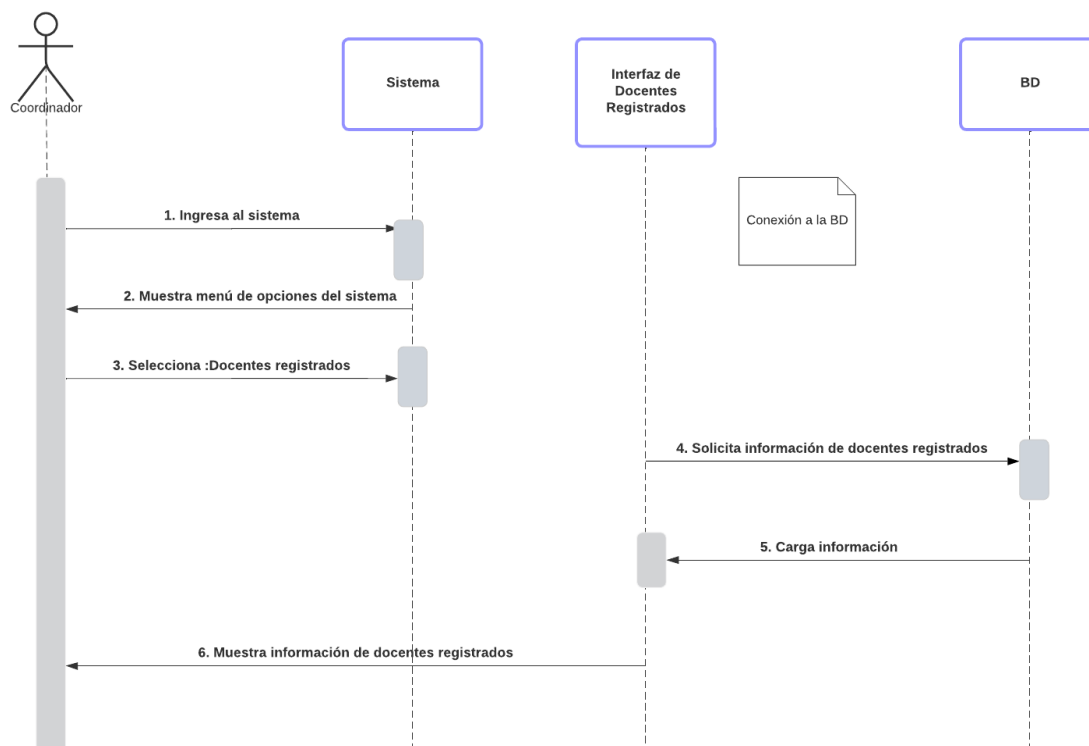


Figura 17 Diagrama de secuencia Consultar Docentes Registrados

Agendar Coloquio

Diagrama de secuencia que expone la comunicación entre el usuario final, el sistema, el formulario de registro y la base de datos para almacenar en el sistema la información pertinente a un coloquio. También muestra el flujo de desarrollo que realiza cada componente si la acción no se completa exitosamente.

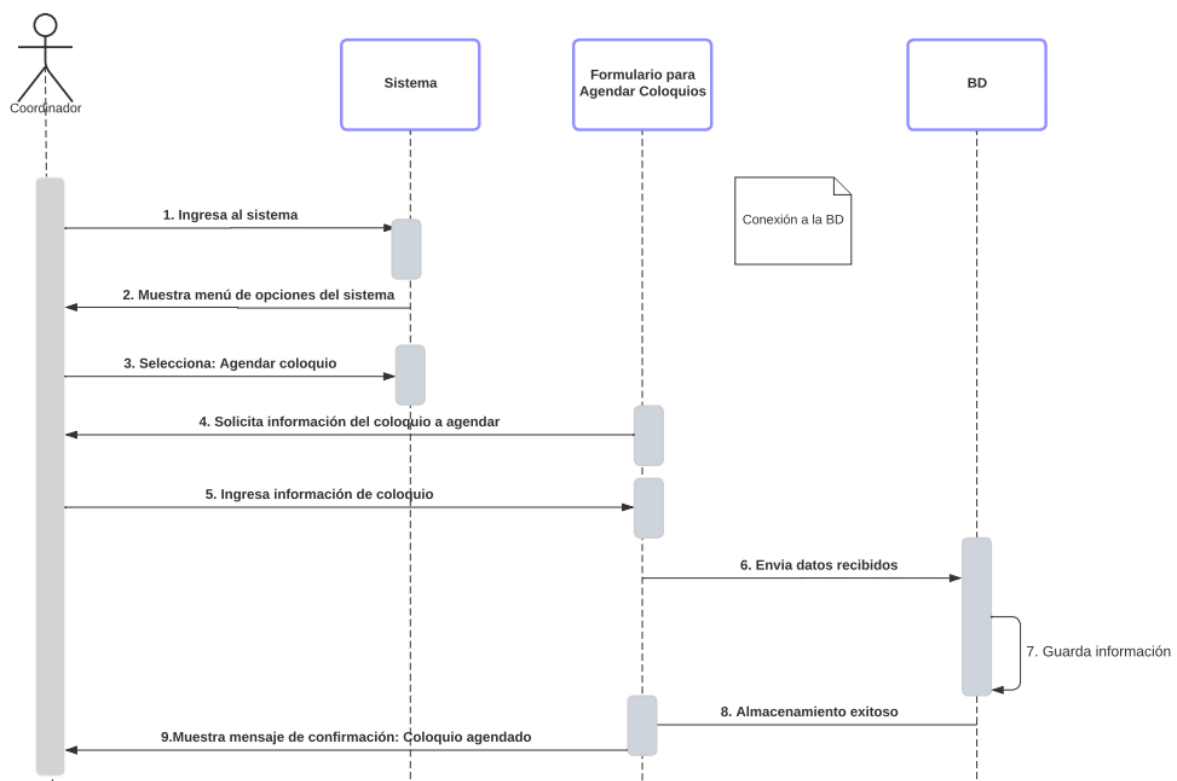


Figura 18 Diagrama de secuencia Agendar Coloquio

Modificar Coloquio

Diagrama de secuencia que precisa la interacción entre el usuario final, el sistema, el formulario de modificación y la base de datos para modificar información relacionada con un coloquio. Igualmente expone la comunicación que se lleva a cabo entre los diferentes componentes si la operación no se efectúa correctamente.

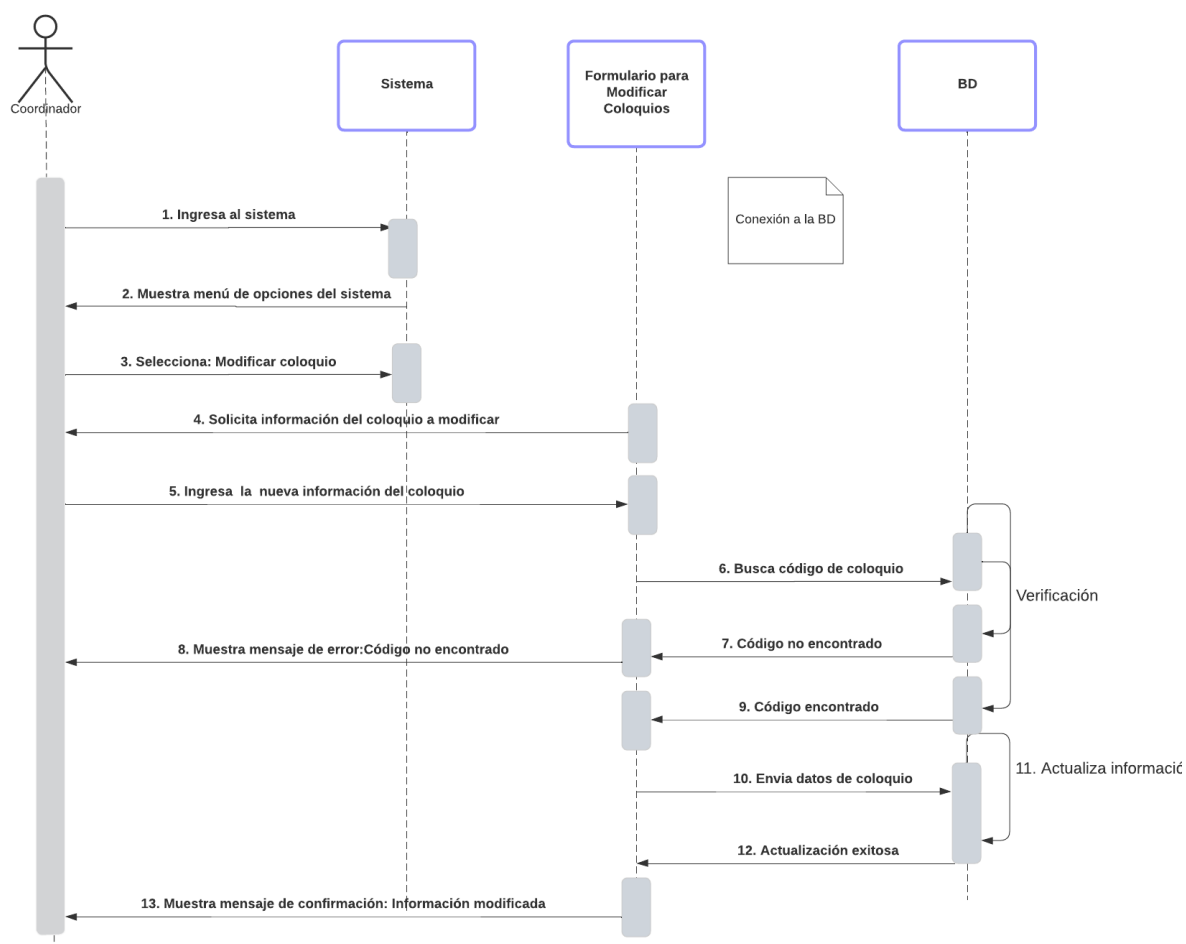


Figura 19 Diagrama de secuencia Modificar Coloquio

Coloquios Registrados

Diagrama de secuencia que describe la interacción entre el usuario final, el sistema, la interfaz de coloquios registrados y la base de datos para consultar toda la información relacionada con los coloquios registrados en el sistema.

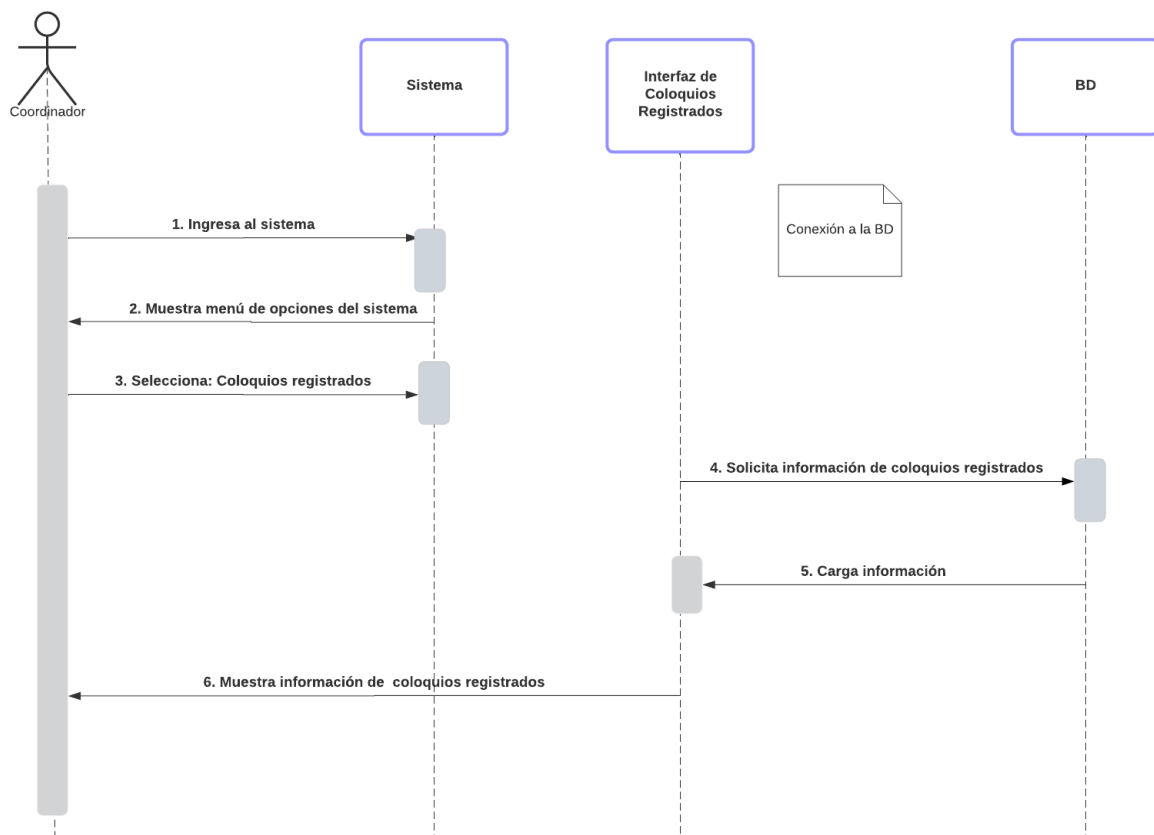


Figura 20 Diagrama de secuencia Consultar Coloquios Registrados

Asignar Cursos a Alumnos

Diagrama de secuencia que muestra la comunicación entre el usuario final, el sistema, el formulario de asignación y la base de datos para registrar información pertinente de los alumnos solicitantes de un curso. También expone la interacción entre los diferentes componentes si la operación no se ejecuta correctamente.

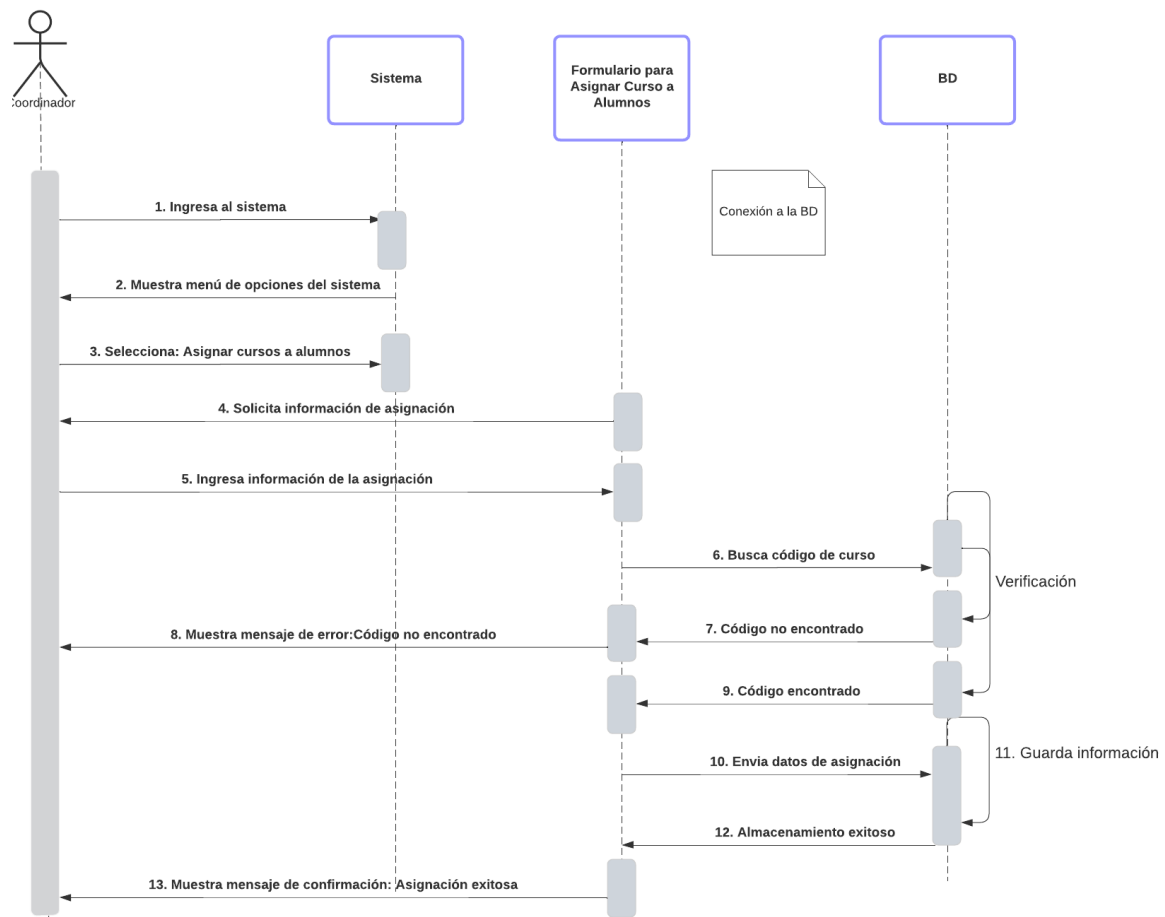


Figura 21 Diagrama de secuencia Asignar Cursos a Alumnos

Asignar Cursos a Docentes

Diagrama de secuencia que precisa la intercomunicación entre el usuario final, el sistema, el formulario de asignación y la base de datos para registrar información de los docentes que impartirán un curso. Asimismo, muestra el proceso de comunicación entre los diferentes componentes si la asignación no se efectúa correctamente.

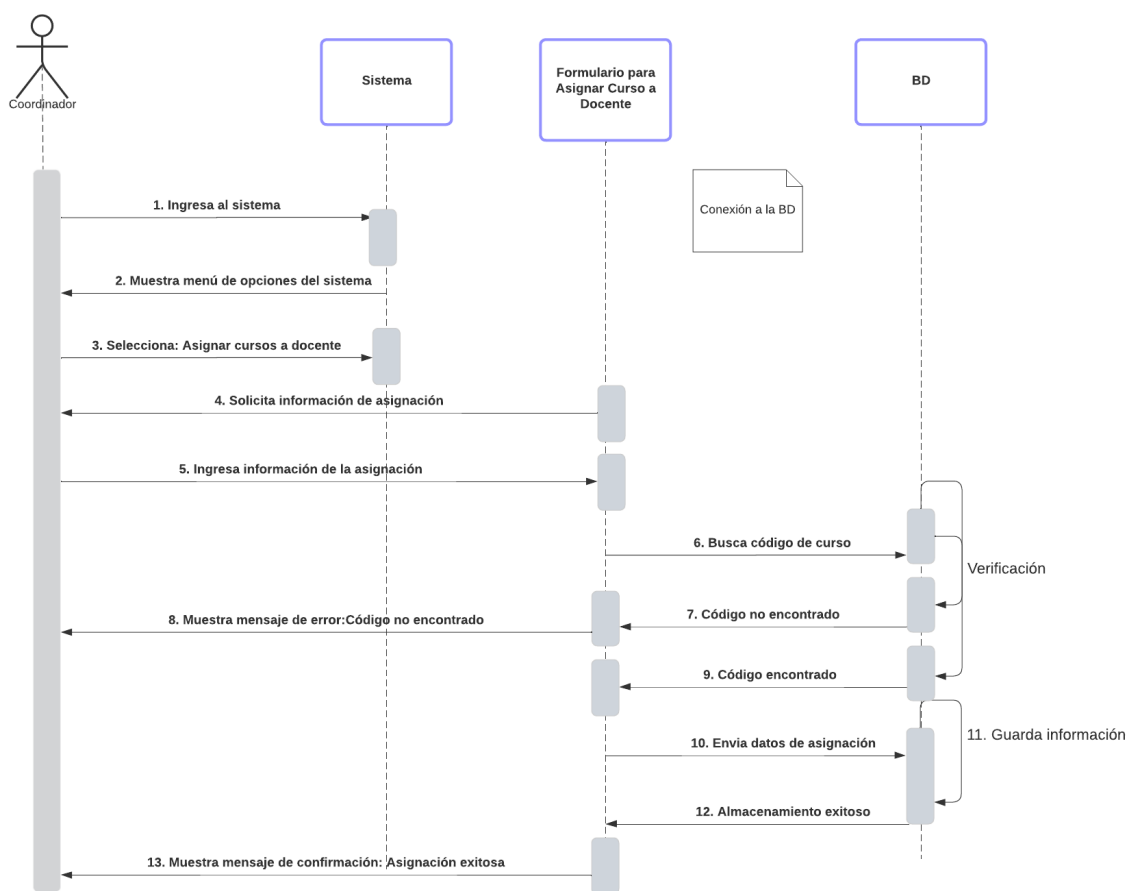


Figura 22 Diagrama de secuencia Asignar Curso a Docentes

Consultar Lista de Cursos

Diagrama de secuencia que describe la interacción entre el usuario final, el sistema, la interfaz de lista de cursos y la base de datos para consultar toda la información relacionada con los cursos almacenados en el sistema.

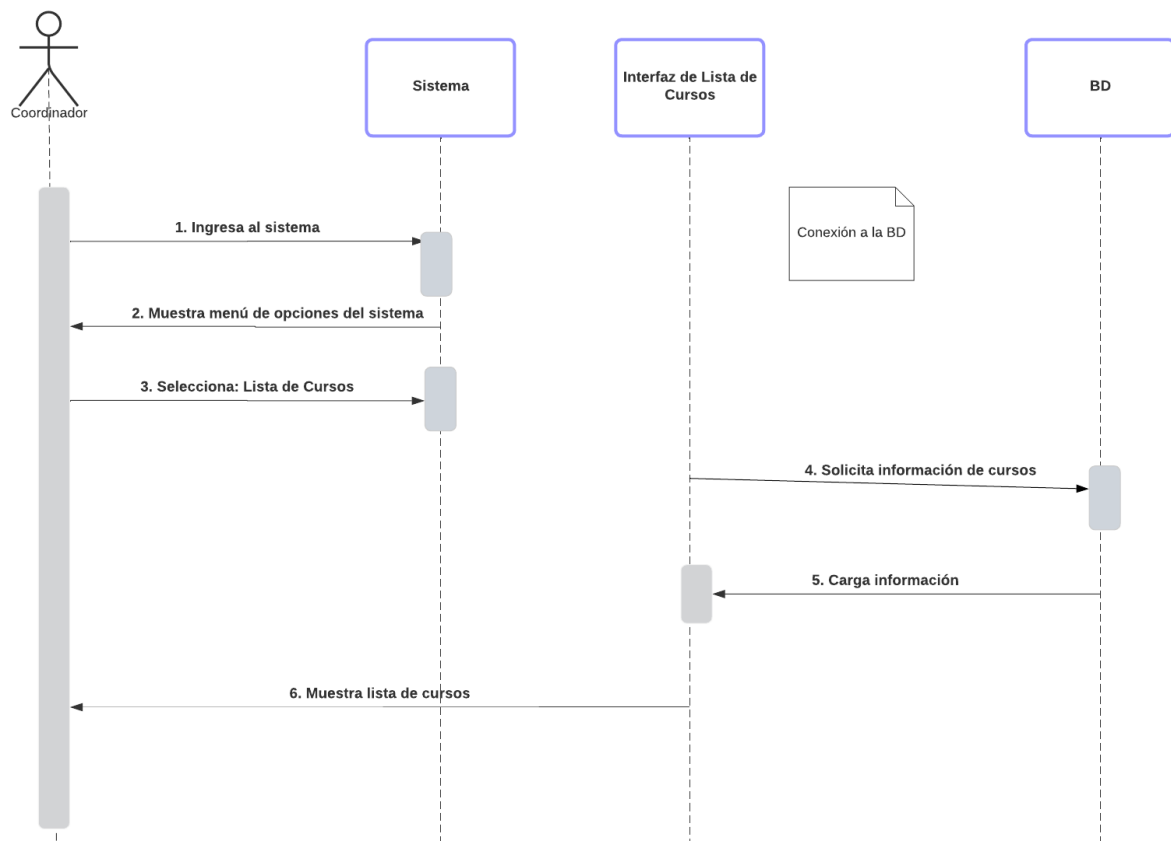


Figura 23 Diagrama de secuencia Consultar Lista de Cursos

Consultar Información de Asignación

Diagrama de secuencia que muestra el proceso de comunicación entre el usuario final, el sistema, la interfaz de información de asignación y la base de datos para consultar toda la información de profesores y alumnos relacionada con un curso.

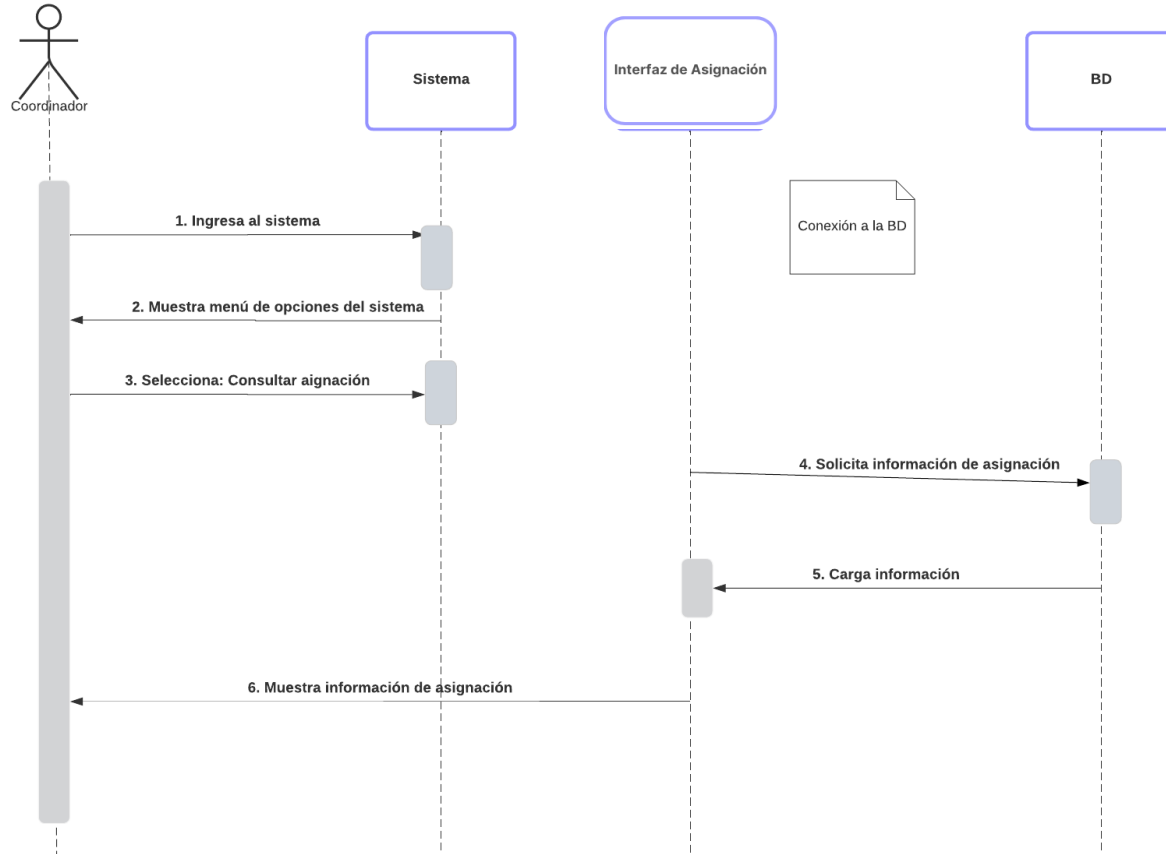


Figura 24 Diagrama de secuencia Consultar Información de Asignación

Mapa de Navegación Jerárquico

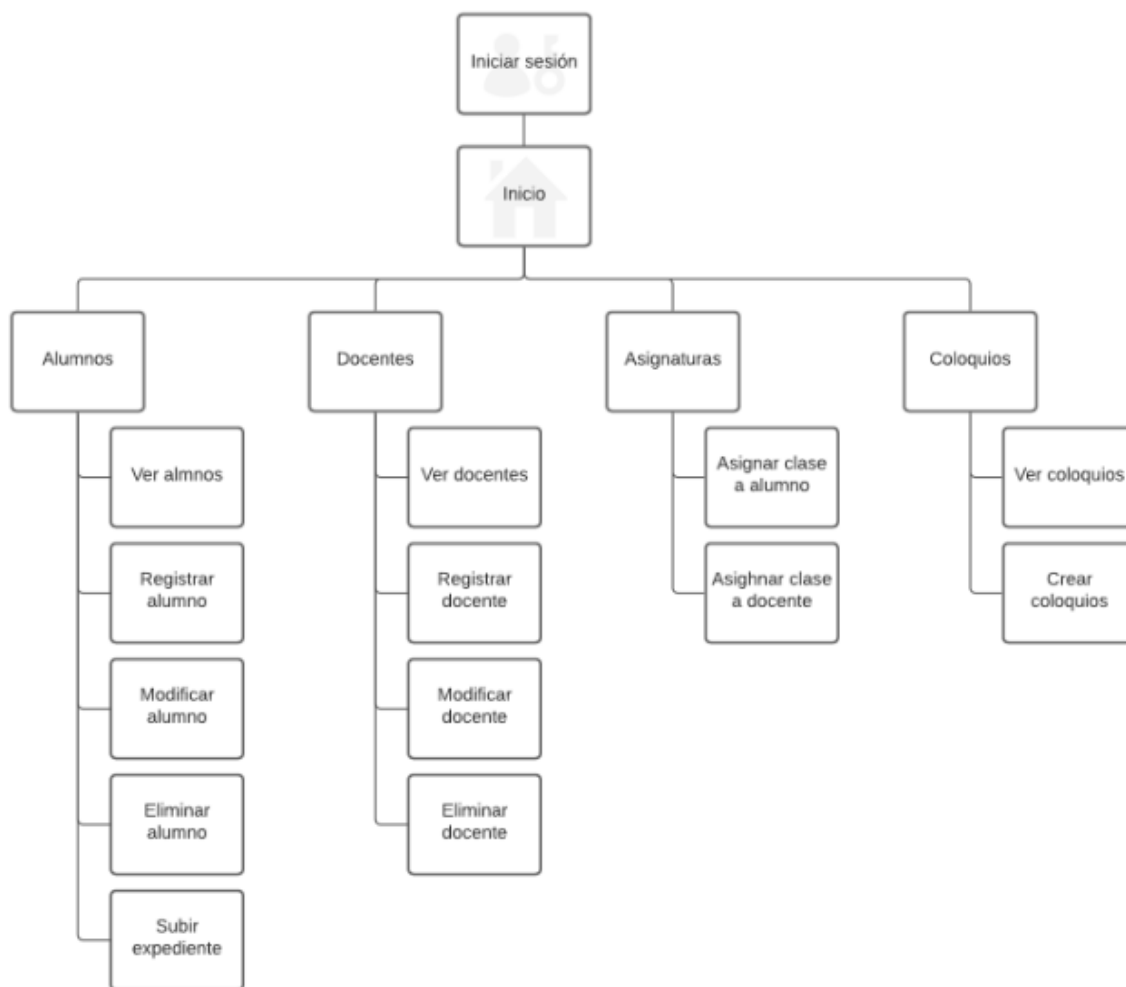


Figura 25 Mapa de navegación del sistema web

Diagrama Entidad Relación

Diagrama que permite definir las tablas que conformarán a la base de datos del sistema web, así como las relaciones entre ellas, las cuales permiten desarrollar la estructura lógica del proyecto.

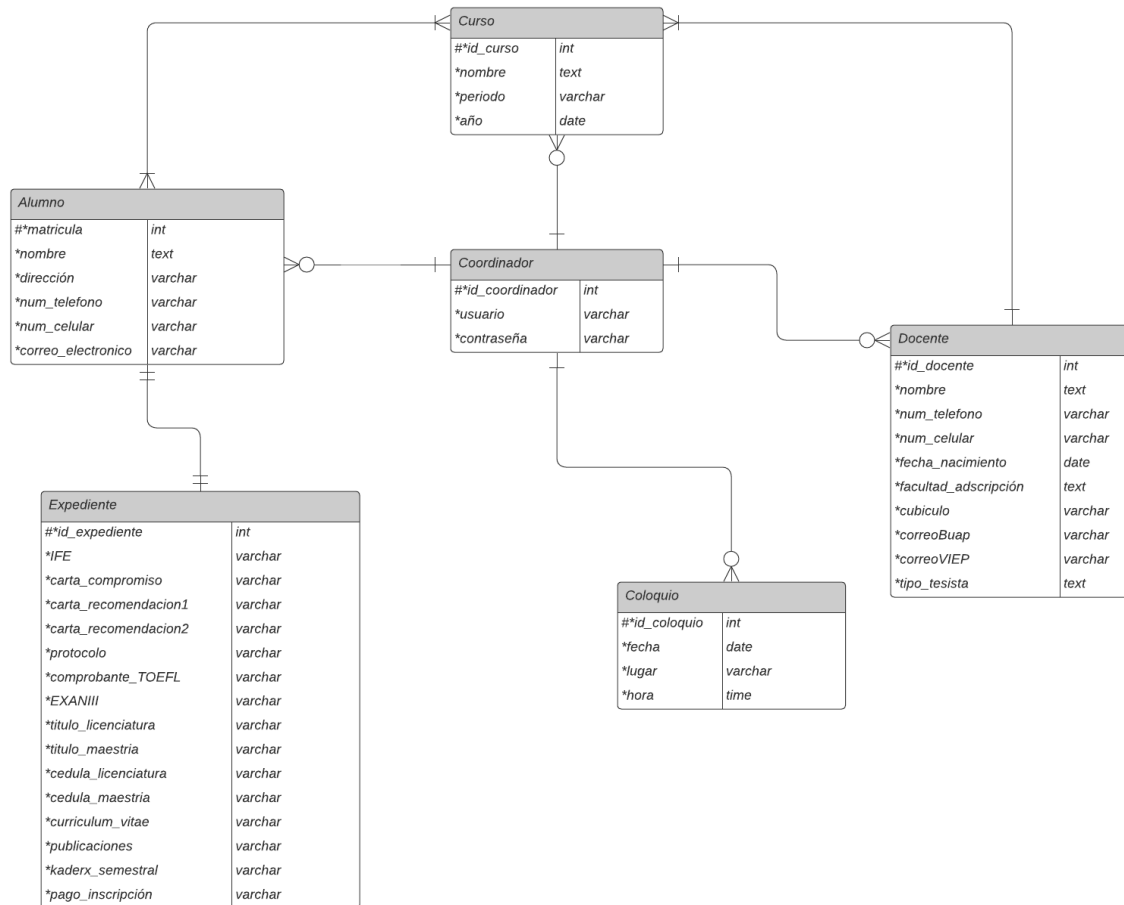


Figura 26 Diagrama entidad relación del sistema web

Estructura de archivos del proyecto web

MVC

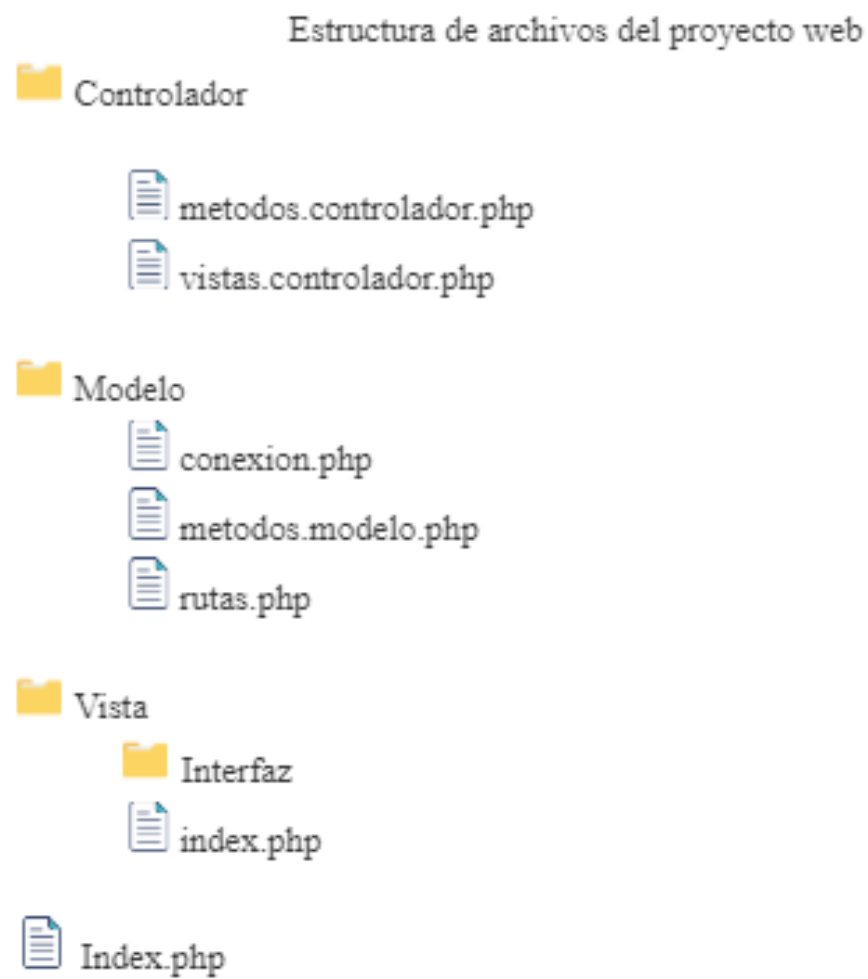


Figura 27 Estructura de archivos del proyecto web

Descripción de archivos y carpetas del proyecto web

En las tablas 22 y 23 se describen las carpetas y archivos de proyecto respectivamente.

Tabla 22 Descripción de carpetas del proyecto web

CARPETA	DESCRIPCIÓN
Controlador	Carpeta que contiene los archivos que realizan la comunicación entre el modelo y la vista.
Modelo	Carpeta que contiene los archivos que realizan operaciones en la base de datos.
Vista	Carpeta que contiene los archivos referentes a la gestión de las diferentes vistas que conforman el sistema.
Vista/CSS	Carpeta que contiene las hojas de estilo de las interfaces del proyecto.
Vista/Interfaz	Carpeta que contiene las interfaces del proyecto.

Tabla 23 Descripción de archivos del proyecto web

ARCHIVO	DESCRIPCIÓN
metodos.controlador.php	Archivo que recibe la información proporcionada en la vista por el usuario.
vistas.controlador.php	Archivo que gestiona las rutas de las interfaces del proyecto,
conexion.php	Archivo que realiza la conexión entre el sistema y la base de datos.
metodos.modelo.php	Archivo que recibe las instrucciones obtenidas en el controlador y las ejecuta en la base de datos.
rutas.php	Archivo que contiene la ruta raíz del proyecto.
Vista/index.php	Archivo que carga la ruta de la interfaz seleccionada por el usuario.
index.php	Archivo encargado de efectuar los ficheros necesarios para la correcta ejecución y funcionamiento del sistema.

Prototipos

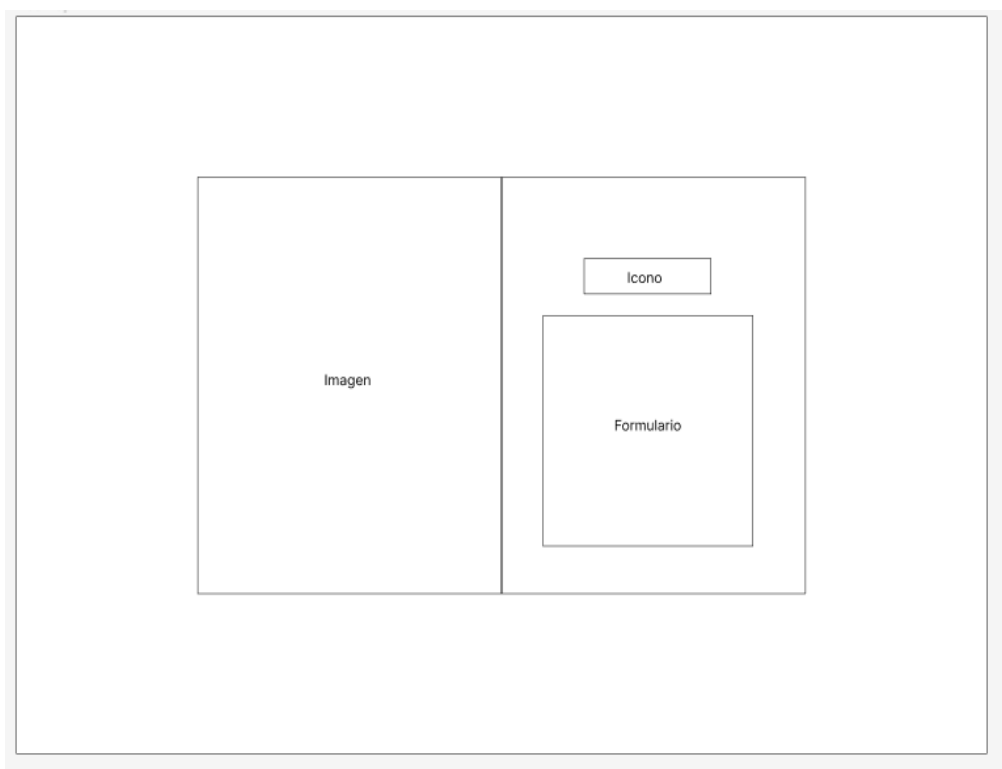


Figura 28 Estructura general del formulario de Inicio de Sesión

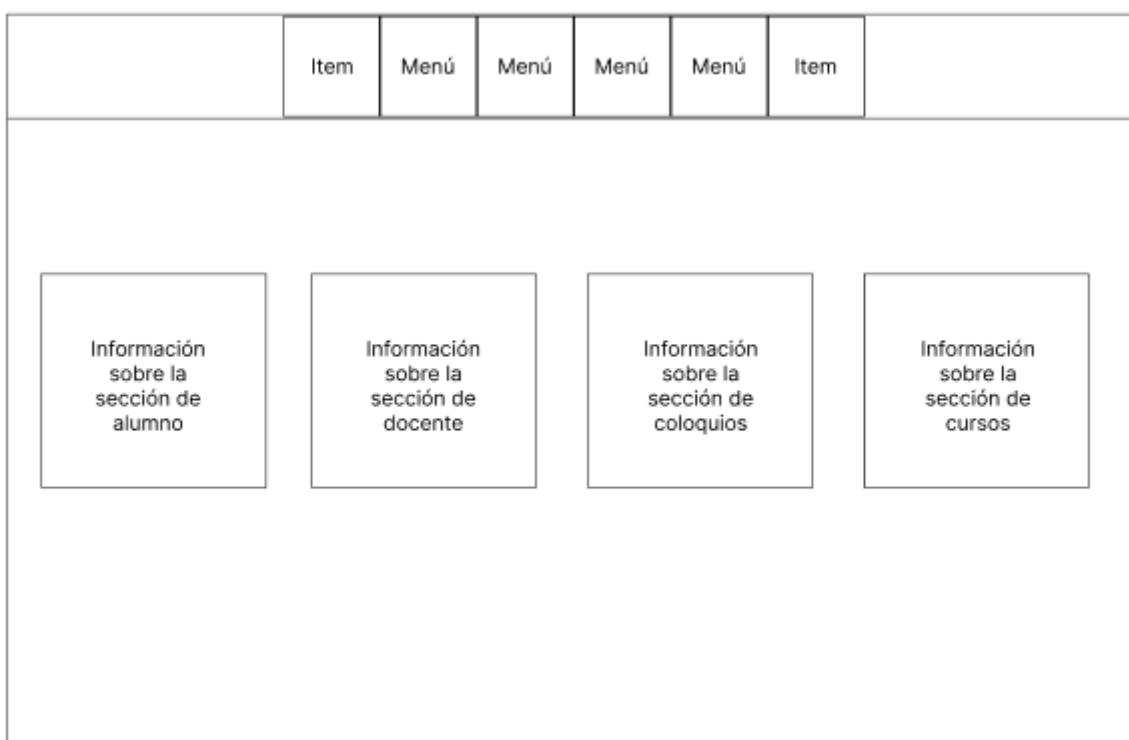


Figura 29 Estructura general de la página principal.

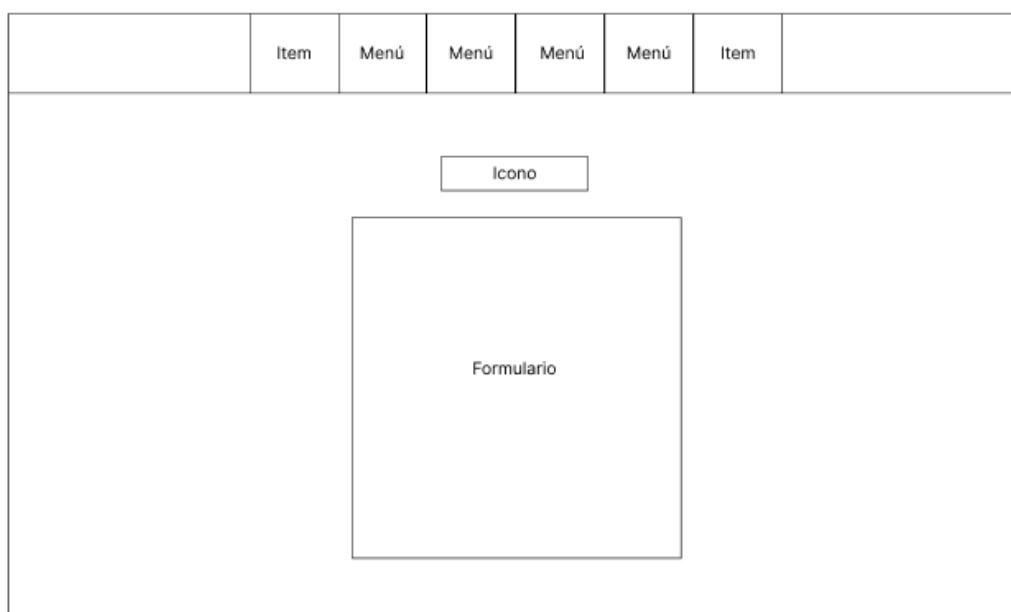


Figura 30 Estructura general de formularios

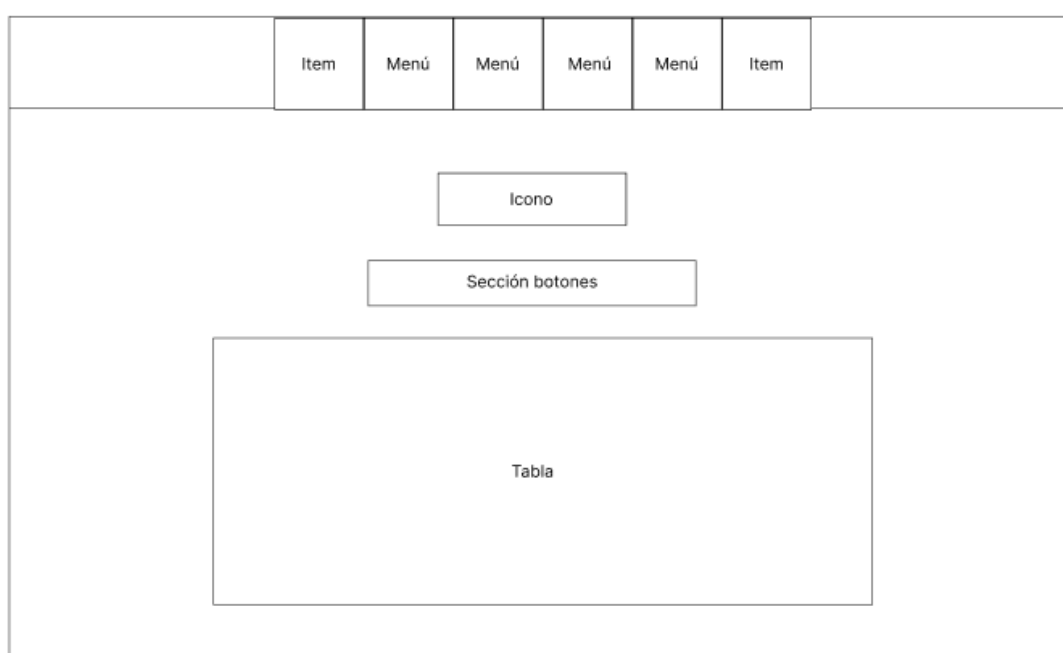


Figura 31 Estructura general de tablas

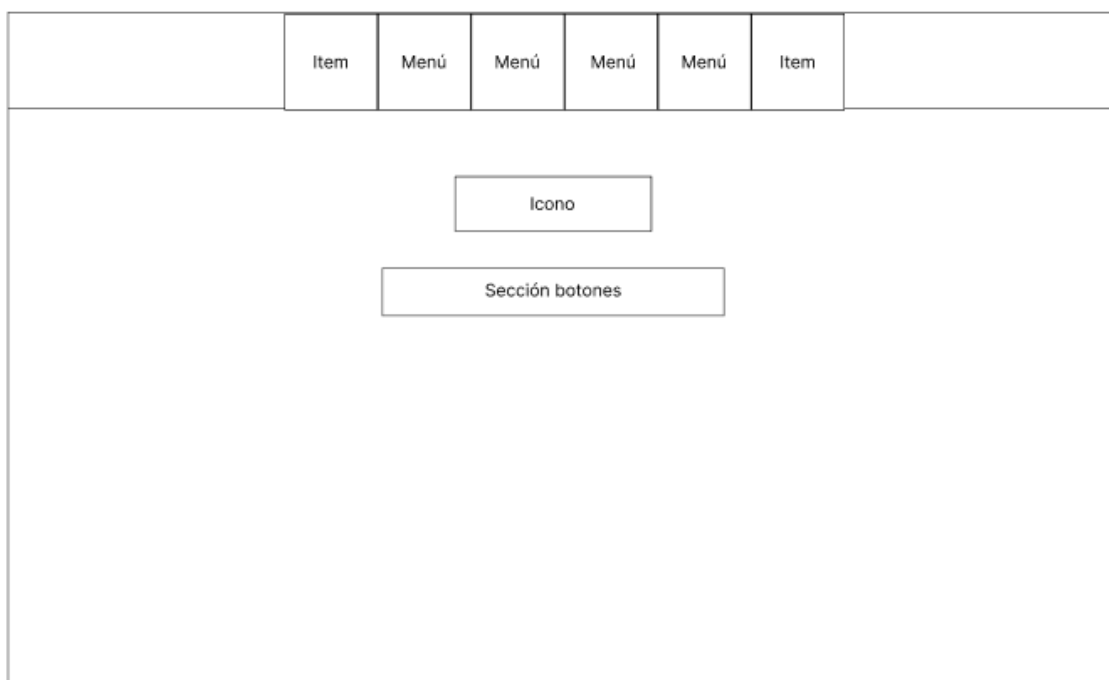


Figura 32 Estructura general de la página Asignar Cursos a Docentes



Figura 33 Estructura general de la página Consultar Información de Asignación.

Programación

Clase Controlador

Función: Iniciar sesión

Función que tiene como finalidad de recibir los valores de usuario y contraseña obtenidos en la interfaz gráfica. Remite los datos a la función Iniciar Sesión perteneciente a la clase Métodos Modelo. Si el usuario existe el método redirige a la página de inicio al usuario, en caso contrario muestra un mensaje de error y niega el ingreso a la plataforma.

```
public static function C_iniciarSesion(){
    if(isset($_POST["usuario"])){
        $usuario=$_POST["usuario"];
        $password=$_POST["password"];
        $tabla = "coordinador";
        $filas=MetodosModelo::M_iniciarSesion($usuario,$password,$tabla);

        if($filas <=0 ){
            echo
            '<script>
            Swal.fire({
                icon: "error",
                title: "Oops...",
                text: "Error en Autenticación de Usuario"
            })
            </script>';
        }else{
            echo '<script>>window.location="inicio" </script>';
        }
    }
}#end
```

Figura 34 Código del método Iniciar Sesión (clase controlador)

Función: Registrar alumno

Función que tiene como objetivo recibir la información ingresada en el formulario de registro y posteriormente enviarla al método de la clase Modelo para su almacenamiento en la base de datos. Como primer paso recibe los datos del alumno, posteriormente envía como parámetro la matrícula del estudiante al método Validar de la clase Métodos Modelo para verificar la existencia del registro en el sistema. Si en la base de datos ya existe información relacionada con esa matrícula, muestra un mensaje de error y no permite el registro de información. Si los datos no existen, convoca a la función Registrar Alumno de la clase Modelo para llevar a cabo el registro de información en la base de datos. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```

public static function C_registrarAlumno(){

    if(isset($_POST["id"])){

        $id=$_POST["id"];
        $tabla = "alumno";

        $filas=MetodosModelo::M_validar($id,$tabla);

        if ($filas <= 0) {

            $datos = array("id"=>$_POST['id'], "nombre"=>$_POST['nombre'],
                "direccion"=>$_POST['direccion'], "num_telefono"=>$_POST['telefono'],
                "num_celular"=>$_POST['celular'], "correop"=>$_POST['correop'],
                "correob"=>$_POST['correob'], "correod"=>$_POST['correod']);

            $respuesta = MetodosModelo::M_registrarAlumno($datos, $tabla);

            if($respuesta == "true"){...
            }
        } else {...
        }
    }
} //end de if isset
}

```

Figura 35 Código del método Registrar Alumno (clase controlador)

Función: Registrar docente

Función que tiene como finalidad recibir los datos ingresados en el formulario de registro y posteriormente enviarlos para su almacenamiento en la base de datos. Como primer paso recibe los datos del docente, posteriormente envía como parámetro el id del docente al método Validar de la clase Metodos Modelo para verificar la existencia del registro en el sistema. Si en la base de datos ya existe información relacionada con ese id, muestra un mensaje de error y no permite el registro de información. Si los datos no existen, convoca a la función Registrar Docente de la clase Modelo para llevar a cabo el registro de información en la base de datos. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```

public static function C_registrarDocente(){
    if(isset($_POST["id"])){
        $id=$_POST["id"];

        $tabla = "docente";
        $filas=MetodosModelo::M_validar($id,$tabla);

        if($filas <=0 ){

            $datos = array("id_docente"=>$_POST['id'], "nombre"=>$_POST['nombre'], "num_telefono"=>$_POST['telefono'],
                "num_celular"=>$_POST['celular'], "fecha_nacimiento"=>$_POST['fecha'], "facultad"=>$_POST['facultad'],
                "cubiculo"=>$_POST['cubiculo'], "correoBuap"=>$_POST['correob'], "correoVIEP"=>$_POST['correov'],
                "tipo_tesista"=>$_POST['t_tesista']);

            $respuesta = MetodosModelo::M_registrarDocente($datos, $tabla);

            if($respuesta == "true"){...
            }
        }else{ ...
        }
    }
}#end

```

Figura 36 Código del método Registrar Docente (clase controlador)

Función: Registrar coloquio

Función que tiene como objetivo recibir la información ingresada en el formulario de registro y posteriormente enviarla a la función Registrar Coloquio en la clase Métodos Modelo para su almacenamiento en la base de datos. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```

#Registrar coloquio
public static function C_registrarColoquio(){
    if(isset($_POST["fecha"])){
        $datos = array("titulo"=>$_POST['titulo'], "fecha"=>$_POST['fecha'], "lugar"=>$_POST['lugar'], "hora"=>$_POST['hora']);

        $tabla = "coloquio";

        $respuesta = MetodosModelo::M_registrarColoquio($datos, $tabla);

        if($respuesta == "true"){
            echo '<script>
                Swal.fire({...
            </script>';
        }else{ ...
        }
    }
}#end

```

Figura 37 Código del método Registrar Coloquio (clase controlador)

Función: Expediente de alumno

Función que recibe la información ingresada en el formulario de registro. Posteriormente, verifica la existencia del alumno en el sistema mediante al método Validar de la clase Métodos Modelo. Si no se encuentra ningún registro relacionado con la matrícula ingresada muestra un mensaje de error y no permite el registro de la información. Si en la base de datos existe información relacionada con esa matrícula, crea la carpeta en donde se almacenarán los archivos PDF ingresados en el formulario, después invoca al método Expediente Alumno de la clase Métodos Modelo y envía la ruta donde se encuentra cada documento. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```
public static function C_expedienteAlumno(){
    if(isset($_POST["id"])){
        $num_archivos=count($_FILES['archivo']['name']); //recupera el numero de archivos recibidos
        $matricula = $_POST["id"];
        $direccion='Documentos/'.$matricula;
        $bandera = true;
        $datos = ['', '', '', '', '', '', '', '', '', '', '', '', '', '', '', ''];
        $tabla="alumno";
        $filas=MetodosModelo::M_validar($matricula,$tabla);

        if($filas<=0){ ...
        }else{
            /*Crea la carpeta con la matricula si no existe*/
            if (!file_exists($direccion)) { ...
            }

            for ($i=0; $i<=$num_archivos; $i++){ ...
            }

            if($bandera == true){
                $tabla = "expediente";

                $respuesta = MetodosModelo::M_expedienteAlumno($matricula,$datos,$tabla);

                if($respuesta == "true"){ ...
                }else{ ...
                }
            }
        }
    }
}
```

Figura 38 Código del método Expediente Alumno (clase controlador)

Función: Eliminar

Función que recibe la información ingresada en el formulario de eliminación, después, envía como parámetro el identificador recibido al método Validar de la clase Métodos Modelo para comprobar que exista información relacionada con el identificador. Si no existe la información, manda a pantalla un mensaje de error, en caso contrario, envía la información a la función Eliminar de la clase Métodos Modelo para eliminar todos los datos relacionados con él. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```

public static function C_eliminar($tabla){
    if(isset($_POST["id"])){

        $id=$_POST['id'];

        $filas=MetodosModelo::M_validar($id,$tabla);

        if($filas <=0 ){ ...
    }else{

        $respuesta = MetodosModelo::M_eliminar($id, $tabla);
        if($respuesta=="true"){ ...
        </script>';

    }

}

}
}#end

```

Figura 39 Código del método Eliminar (clase controlador)

Función: Modificar

Función que recibe la información ingresada en el formulario de modificación, para posteriormente, verificar la existencia del identificador en el sistema mediante el método Validar de la clase Métodos Modelo. Si no existe la información, manda a pantalla un mensaje de error, en caso contrario, envía la información obtenida en el formulario a la función Modificar de la clase Métodos Modelo para realizar la modificación. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```

public static function C_modificar($tabla){

    if(isset($_POST["id"])){

        $id=$_POST['id'];

        $filas=MetodosModelo::M_validar($id,$tabla);

        if($filas <=0 ){ ...
    }else{

        $datos=array("id"=>$_POST['id'], "rubro"=>$_POST['rubro'], "modificacion"=>$_POST['modificacion']);

        $respuesta = MetodosModelo::M_modificar($datos, $tabla);

        if($respuesta == "true"){ ...
        </script>';
    }

}

}

```

Figura 40 Código del método Modificar (clase controlador)

Función: Asignar alumno

Función que tiene como objetivo recibir la información ingresada en el formulario de registro y posteriormente enviarla para su almacenamiento en la base de datos. Como primer paso recibe la información, posteriormente envía como parámetro las matrículas de los estudiantes al método Validar de la clase Métodos Modelo para verificar su existencia en el sistema. Si los datos no existen, muestra un mensaje de error y no permite el registro de información. Si en la base de datos ya existe información relacionada con esas matrículas, convoca a la función Asignar Alumno de la clase Métodos Modelo para llevar a cabo el registro de información en la base de datos. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```
public static function C_asignarAlumno(){
    if(isset($_POST["curso"])){
        $longitud =count($_POST['matricula']);//recupera el numero de archivos recibidos
        $curso=$_POST["curso"];
        $datos= array();
        $faltantes= array();
        $bandera=0;

        for($i=0;$i<$longitud;$i++)
        { ...
        }

        $tabla_validar="alumno";
        for($i=0;$i<$longitud;$i++)
        {
            $id=$datos[$i];
            $filas = MetodosModelo::M_validar($id,$tabla_validar);
            if($filas<=0){
                $bandera=1;
                $faltantes[$i]=$id;
            }
        }
        $tamaño=count($faltantes);

        if($bandera==1){ ...
        }else{
            $tabla = "alumno_asignacion";
            $respuesta = MetodosModelo::M_asignarAlumno($longitud,$datos,$curso,$tabla);
            if($respuesta == "true"){ ...
            </script>';
        }
    }
}
```

Figura 41 Código del método Asignar Alumno (clase controlador)

Función: Asignar tutoría

Función que tiene como objetivo recibir la información ingresada en el formulario de registro y posteriormente enviarla para su almacenamiento en la base de datos. Como primer paso recibe la información, posteriormente envía como parámetro el id del docente al método Validar de la clase Métodos Modelo para verificar su existencia en el sistema. Si los datos no existen, muestra un mensaje de error y no permite el registro de información. Si en la base de datos ya existe información relacionada con el id, convoca a la función Asignar Tutoría de la clase Metodos Modelo para llevar a cabo el registro de información en la base de datos.

```
public static function C_asignarTutoria(){

    if(isset($_POST["id"])){

        echo "Entra a tutoria docente";
        $id=$_POST["id"];
        $tabla_validar="docente";

        $filas = MetodosModelo::M_validar($id,$tabla_validar);
        echo $filas;

        if($filas<=0){ ...
        }else{

            $datos = array("curso"=>$_POST['curso'], "id_docente"=>$_POST['id'],
            "docente"=>$_POST['nombre'], "solicitantes"=>$_POST['solicitantes']);
            $tabla = "docente_asignacion";

            $respuesta = MetodosModelo::M_asignarTutoria($datos, $tabla);

            if($respuesta == "true"){ ...
            </script>';

            }else{ ...
            }
        }
    }
}
```

Figura 42 Código del método Asignar Tutoría (clase controlador)

Función: Asignar materia

Función que tiene como objetivo recibir la información ingresada en el formulario de registro y posteriormente enviarla para su almacenamiento en la base de datos. Como primer paso recibe la información, posteriormente envía como parámetro los identificadores de los docentes al método Validar de la clase Métodos Modelo para verificar su existencia en el sistema. Si los datos no existen, muestra un mensaje de error y no permite el registro de información. Si en la base de datos ya existe información relacionada con los id's, convoca a la función Asignar Materia de la clase Metodos Modelo para llevar a cabo el registro de información en la base de datos.

```
public static function C_otraMateria(){
    if(isset($_POST["id"])){
        $id=$_POST["id"];
        $id_aux=$_POST["id_aux"];
        $tabla_validar="docente";

        $filas = MetodosModelo::M_validar($id,$tabla_validar);
        echo $filas;
        $filas2 = MetodosModelo::M_validar($id_aux,$tabla_validar);
        echo $filas2;

        if($filas <=0 || $filas2 <=0 ){
            echo '<script>
            Swal.fire({...
        }else{

            $datos=array("curso"=>$_POST['curso'], "solicitanter"=>$_POST['solicitanter'], "id"=>$_POST['id'], "docente"=>$_POST['docente'],
            "id_aux"=>$_POST['id_aux'], "docente_aux"=>$_POST['docente_aux']);

            $tabla = "docente_asignacion";

            $respuesta = MetodosModelo::M_asignarMateria($datos, $tabla);
            if($respuesta == "true"){ ...
            </script>';
        }else{ ...
        }
    }
}
```

Figura 43 Código del método Asignar Materia (clase controlador)

Función: Mostrar Tabla

Función que invoca al método Mostrar Tabla de la clase Métodos Modelo para la recuperación de información, una vez obtenidos los datos los retorna a la interfaz gráfica.

```
public static function C_mostrarTabla($tabla){
    $respuesta = MetodosModelo::M_mostrarTabla($tabla);
    return $respuesta;
}#end
```

Figura 44 Código del método Mostrar Tabla (clase controlador)

Clase Modelo

Función: Registrar Alumno

Función que recibe los metadatos del alumno y efectúa su almacenamiento en la base de datos mediante la sentencia INSERT INTO utilizando PDO.

```
public static function M_registrarAlumno($datos, $tabla){
    //conexión a la bd
    $con = Conexion::conectar()->prepare("INSERT INTO $tabla(id,nombre,direccion,num_telefono,num_celular,correo_personal,correo_buap,
    correo_doctorado) VALUES (:matricula,:nombre,:direccion,:num_telefono,:num_celular,:correo_personal,:correo_buap,:correo_doctorado)");

    $con -> bindParam(":matricula", $datos["id"], PDO::PARAM_STR);
    $con -> bindParam(":nombre", $datos["nombre"], PDO::PARAM_STR);
    $con -> bindParam(":direccion", $datos["direccion"], PDO::PARAM_STR);
    $con -> bindParam(":num_telefono", $datos["num_telefono"], PDO::PARAM_STR);
    $con -> bindParam(":num_celular", $datos["num_celular"], PDO::PARAM_STR);
    $con -> bindParam(":correo_personal", $datos["correop"], PDO::PARAM_STR);
    $con -> bindParam(":correo_buap", $datos["correob"], PDO::PARAM_STR);
    $con -> bindParam(":correo_doctorado", $datos["corneod"], PDO::PARAM_STR);

    if($con -> execute()){
        | return "true";
    }else{
        | return "false";
    }
}
}#end
```

Figura 45 Código del método Registrar Alumno (clase modelo)

Función: Registrar Docente

Función que recibe los metadatos del docente a ingresa en el sistema y efectúa su almacenamiento en la base de datos mediante la sentencia INSERT INTO utilizando PDO.

```
public static function M_registrarDocente($datos, $tabla){
    $stmt = Conexion::conectar()->prepare("INSERT INTO $tabla(id,nombre,num_telefono,num_celular,fecha_nacimiento,facultad_adscripcion,
    cubiculo,correoBuap,correoVIEP,tipo_tesista) VALUES(:id_docente,:nombre,:num_telefono,:num_celular,:fecha_nacimiento,:facultad_adscripcion,
    :cubiculo,:correoBuap,:correoVIEP,:tipo_tesista)");

    $stmt -> bindParam(":id_docente", $datos["id_docente"], PDO::PARAM_STR);
    $stmt -> bindParam(":nombre", $datos["nombre"], PDO::PARAM_STR);
    $stmt -> bindParam(":num_telefono", $datos["num_telefono"], PDO::PARAM_STR);
    $stmt -> bindParam(":num_celular", $datos["num_celular"], PDO::PARAM_STR);
    $stmt -> bindParam(":fecha_nacimiento", $datos["fecha_nacimiento"], PDO::PARAM_STR);
    $stmt -> bindParam(":facultad_adscripcion", $datos["facultad"], PDO::PARAM_STR);
    $stmt -> bindParam(":cubiculo", $datos["cubiculo"], PDO::PARAM_STR);
    $stmt -> bindParam(":correoBuap", $datos["correoBuap"], PDO::PARAM_STR);
    $stmt -> bindParam(":correoVIEP", $datos["correoVIEP"], PDO::PARAM_STR);
    $stmt -> bindParam(":tipo_tesista", $datos["tipo_tesista"], PDO::PARAM_STR);

    if($stmt -> execute()){
        | return "true";
    }else{
        | return "false";
    }
}
```

Figura 46 Código del método Registrar Docente (clase modelo)

Función: Registrar Coloquio

Función que recibe los metadatos del coloquio a registrar y efectúa su almacenamiento en la base de datos mediante la sentencia INSERT INTO utilizando PDO.

```
public static function M_registrarColoquio($datos, $tabla){

    //conexión a la bd
    $con = Conexion::conectar()->prepare("INSERT INTO $tabla(titulo,fecha,lugar,hora) VALUES (:titulo,:fecha,:lugar,:hora)");

    $con -> bindParam(":titulo", $datos["titulo"], PDO::PARAM_STR);
    $con -> bindParam(":fecha", $datos["fecha"], PDO::PARAM_STR);
    $con -> bindParam(":lugar", $datos["lugar"], PDO::PARAM_STR);
    $con -> bindParam(":hora", $datos["hora"], PDO::PARAM_STR);

    if($con -> execute()){

        | return "true";

    }else{

        | return "false";

    }
}
```

Figura 47 Código del método Registrar Coloquio (clase modelo)

Función: Expediente Alumno

Función que recibe la ubicación de los archivos PDF y efectúa su almacenamiento en la base de datos mediante la sentencia INSERT INTO utilizando PDO.

```
public static function M_expedienteAlumno($matricula,$datos,$tabla){

    $con = Conexion::conectar()->prepare("INSERT INTO $tabla(matricula,IFE,
    carta_compromiso,carta_recomendacion1,carta_recomendacion2,protocolo,
    comprobante_TOEFL,EXANIII,titulo_licenciatura,titulo_maestria,cedula_licenciatura,
    cedula_maestria,curriculum_vitae,publicaciones,kardex_semestral,pago_inscripcion) VALUES(:matricula,:IFE,:carta_compromiso,
    :carta_recomendacion1,:carta_recomendacion2,:protocolo,
    :comprobante_TOEFL,:EXANIII,:titulo_licenciatura,
    :titulo_maestria,:cedula_licenciatura,:cedula_maestria,:curriculum_vitae,:publicaciones,:kardex_semestral,:pago_inscripcion)");

    $con -> bindParam(":matricula", $matricula, PDO::PARAM_STR);
    $con -> bindParam(":IFE", $datos[0], PDO::PARAM_STR);
    $con -> bindParam(":carta_compromiso", $datos[1], PDO::PARAM_STR);
    $con -> bindParam(":carta_recomendacion1", $datos[2], PDO::PARAM_STR);
    $con -> bindParam(":carta_recomendacion2", $datos[3], PDO::PARAM_STR);
    $con -> bindParam(":protocolo", $datos[4], PDO::PARAM_STR);
    $con -> bindParam(":comprobante_TOEFL", $datos[5], PDO::PARAM_STR);
    $con -> bindParam(":EXANIII", $datos[6], PDO::PARAM_STR);
    $con -> bindParam(":titulo_licenciatura", $datos[7], PDO::PARAM_STR);
    $con -> bindParam(":titulo_maestria", $datos[8], PDO::PARAM_STR);
    $con -> bindParam(":cedula_licenciatura", $datos[9], PDO::PARAM_STR);
    $con -> bindParam(":cedula_maestria", $datos[10], PDO::PARAM_STR);
    $con -> bindParam(":curriculum_vitae", $datos[11], PDO::PARAM_STR);
    $con -> bindParam(":publicaciones", $datos[12], PDO::PARAM_STR);
    $con -> bindParam(":kardex_semestral", $datos[13], PDO::PARAM_STR);
    $con -> bindParam(":pago_inscripcion", $datos[14], PDO::PARAM_STR);

}
```

Figura 48 Código del método Expediente Alumno (clase modelo)

Función: Eliminar

Función que elimina toda la información relacionada con un identificador mediante la sentencia DELETE utilizando PDO. Recibe como parámetros el nombre de la tabla y el identificador.

```
public static function M_eliminar($datos, $tabla){

    $stmt = Conexion::conectar()->prepare("DELETE from $tabla where id =:id");

    $stmt -> bindParam(":id", $datos, PDO::PARAM_STR);

    if($stmt -> execute()){

        | return "true";

    }else{

        | return "false";

    }

}

}#end
```

Figura 49 Código del método Eliminar (clase modelo)

Función: Modificar

Función que modifica una propiedad relacionada con un registro contenido en la base de datos por medio de su identificador. Este método recibe como parámetros el nombre de la tabla, el identificador, la propiedad a modificar y el nuevo valor de la propiedad. La ejecución de la acción se realiza mediante la sentencia UPDATE utilizando PDO.

```
public static function M_modificar($datos, $tabla){

    $var=$datos["rubro"];

    $con = Conexion::conectar()->prepare("UPDATE $tabla set $var=:modificacion where id=:id");

    $con->bindValue(':id', $datos["id"], PDO::PARAM_INT);
    $con->bindValue(':modificacion', $datos["modificacion"], PDO::PARAM_STR);

    // ...
}
```

Figura 50 Código del método Modificar (clase modelo)

Función: Asignar Alumno

Función que recibe como parámetros las matrículas de los alumnos aspirantes, el nombre del curso y la cantidad de solicitantes, para efectuar su almacenamiento en la base de datos mediante la sentencia INSERT INTO utilizando PDO.

```
public static function M_asignarAlumno($solicitantes,$datos,$curso,$tabla){
    $bandera = true;

    for($i=0;$i<$solicitantes;$i++)
    {
        $con= Conexion::conectar()->prepare("INSERT INTO $tabla(curso,matricula) VALUES (:curso,:matricula)");

        $con -> bindParam(":curso", $curso, PDO::PARAM_STR);
        $con -> bindParam(":matricula", $datos[$i], PDO::PARAM_STR);

        if($con -> execute()){...
        }else{...
        }

        $con = null;
    }

    if($bandera == "true"){...
}#end
```

Figura 51 Código del método Asignar Alumno (clase modelo)

Función: Asignar Tutoría

Función que recibe como parámetro el nombre del curso, el nombre e id del docente que impartirá la materia. Efectúa el registro de información en la base de datos empleando la sentencia INSERT INTO utilizando PDO.

```
public static function M_asignarTutoria($datos, $tabla){

    //conexión a la bd
    $con= Conexion::conectar()->prepare("INSERT INTO $tabla(curso,id_docente,docente,num_solicitantes) VALUES (:curso,:id_docente,
:docente,:solicitantes)");

    $con -> bindParam(":curso", $datos["curso"], PDO::PARAM_STR);
    $con -> bindParam(":id_docente", $datos["id_docente"], PDO::PARAM_STR);
    $con -> bindParam(":docente", $datos["docente"], PDO::PARAM_STR);
    $con -> bindParam(":solicitantes", $datos["solicitantes"], PDO::PARAM_STR);
```

Figura 52 Código del método Asignar Tutoría (clase modelo)

Función: Asignar Materia

Función que recibe como parámetro el nombre del curso, el nombre e id de los docentes que impartirán la materia. El registro de información en la base de datos se realiza mediante la sentencia INSERT INTO utilizando PDO.

```
public static function M_asignarMateria($datos, $tabla){

    $con= Conexion::conectar()->prepare("INSERT INTO $tabla(curso,id_docente,docente,id_aux,docente_aux,num_solicitantes) VALUES (:curso,
    :id_docente,:docente,:id_aux,:docente_aux,:solicitantes)");

    $con -> bindParam(":curso", $datos["curso"], PDO::PARAM_STR);
    $con -> bindParam(":id_docente", $datos["id"], PDO::PARAM_STR);
    $con -> bindParam(":docente", $datos["docente"], PDO::PARAM_STR);
    $con -> bindParam(":id_aux", $datos["id_aux"], PDO::PARAM_STR);
    $con -> bindParam(":docente_aux", $datos["docente_aux"], PDO::PARAM_STR);
    $con -> bindParam(":solicitantes", $datos["solicitantes"], PDO::PARAM_STR);

    if($con -> execute()){...
    }else{...
    }

}#end
```

Figura 53 Código del método Asignar Materia (clase modelo)

Función: Mostrar Tabla

Función que recupera información de la base de datos mediante la sentencia SELECT utilizando PDO. Recibe como parámetro el nombre de la tabla.

```
public static function M_mostrarTabla($tabla){

    $con = Conexion::conectar()->prepare("SELECT * FROM $tabla");
    $con -> execute();

    $resultado = $con -> fetchAll();
    return $resultado;

    $resultado = null;
    $con = null;

}#end
```

Figura 54 Código del método Mostrar Tabla (clase modelo)

Función: Validar

Función que utiliza un identificador para verificar la existencia de un registro en la base de datos y retorna el número de filas relacionadas con el id. La acción se realiza mediante la sentencia SELECT con PDO.

```
public static function M_validar($id,$tabla){  
    $con = Conexion::conectar()->prepare("SELECT id FROM $tabla WHERE id=:id");  
    $con -> bindParam(":id", $id, PDO::PARAM_STR);  
    $con->execute();  
  
    $filas = $con->rowCount();#Obtiene el numero de filas  
    return $filas;  
}#end
```

Figura 55 Código del método Validar (clase modelo)

Incremento 2

Análisis

En la Tabla se describen los requerimientos no funcionales correspondientes al sistema web.

Tabla 24 Requerimientos No Funcionales

Requerimientos No Funcionales	
Tipo de Requerimiento	Descripción de aplicación
Integridad en los datos	Diseñar y crear una base de datos que le proporcione integridad y exactitud a la información almacenada.
Seguridad	Asegurar y controlar el acceso al sistema web.
Eficiencia	Programar módulos funcionales que faciliten la alta,baja,modificación y salida de información logrando así una gestión eficaz.
Usabilidad	Diseñar y programar un sistema web que permita gestionar información sin tener que realizar procesos tediosos.
Usabilidad	Diseñar y programar una interfaz intuitiva para el usuario.
Disponibilidad	El usuario debe poder utilizar el sistema en cualquier momento.

Diseño

Mapa de Navegación Jerárquico

Diagrama que muestra la organización de los niveles y subniveles que permite al usuario final recorrer el sistema web de forma eficiente y eficaz.

La jerarquía de la estructura está dividida en 3 niveles, en el primer nivel se encuentra el inicio de sesión, en el segundo nivel las secciones que conforman al sistema y en el último nivel las opciones que conforman cada sección (Figura 25).

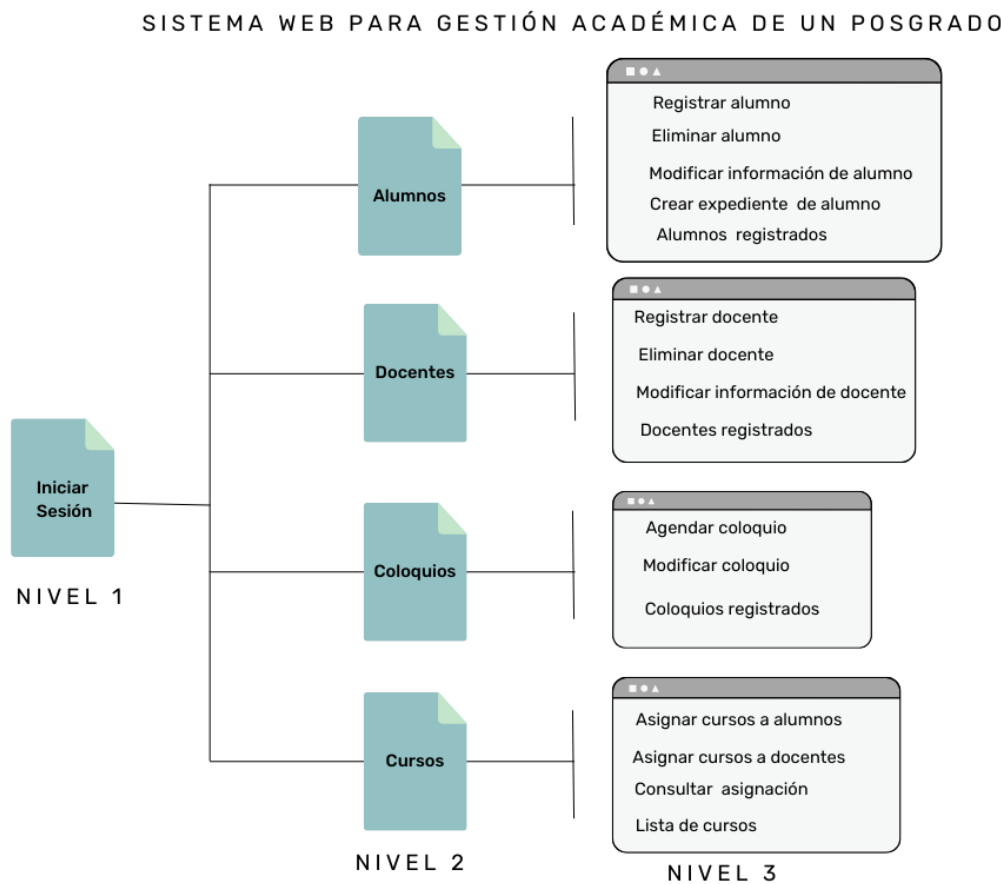


Figura 56 Mapa de navegación actualizado del sistema web

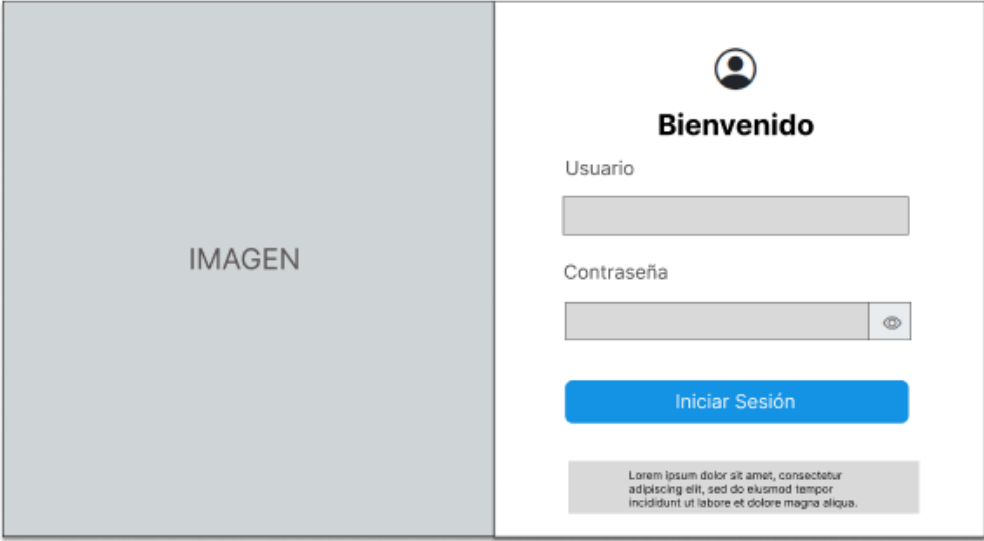
Prototipos

De la figura a se presentan los prototipos actualizados de las interfaces de usuario gráficas.

A continuación se mencionan los recursos utilizados para el diseño de las interfaces de usuario.

Tabla 25 Recursos utilizados en el diseño del sistema web

Tipografía	Gama de colores	Iconos
Helvetica Neue	#000000	bi bi-mortarboard-fill
Helvetica	#FFFFFF	bi bi-person-video3
Arial	#0d6efd	bi bi-chat-dots-fill
sans-serif	#0093E9	bi bi-journal-text
	#80D0C7	bi bi-info-circle-fill



El prototipo muestra una interfaz de usuario para el inicio de sesión. A la izquierda hay un recuadro gris etiquetado como "IMAGEN". A la derecha, el formulario tiene un encabezado con un ícono de usuario y el texto "Bienvenido". Debajo, hay campos para "Usuario" y "Contraseña", un botón azul "Iniciar Sesión" y un pie de página con texto de relleno.

Figura 57 Estructura general actualizada del formulario de Inicio de Sesión

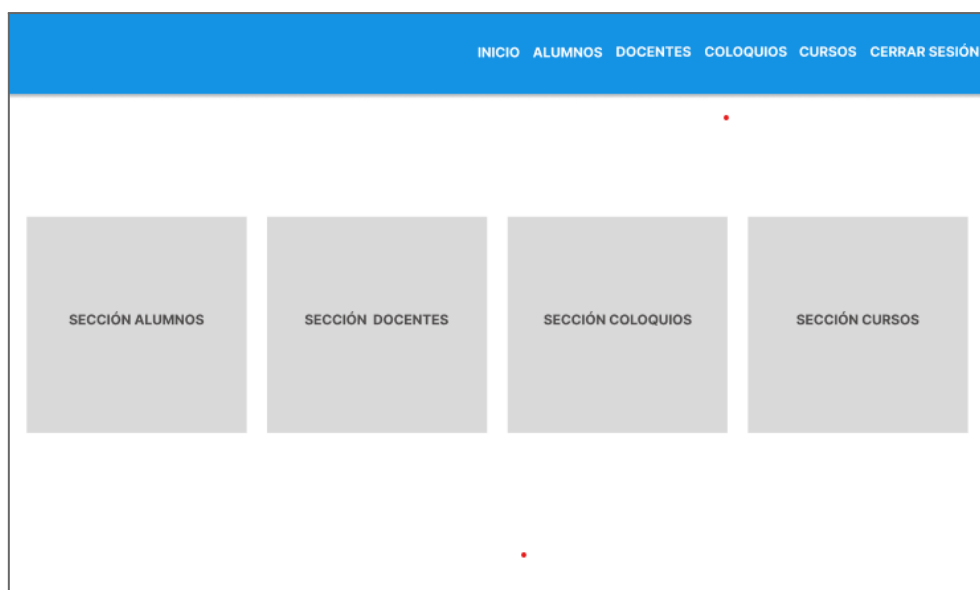


Figura 58 Estructura general actualizada de la página principal



Figura 59 Estructura general actualizada de formularios

INICIO

ALUMNOS

DOCENTES

COLOQUIOS

CURSOS

CERRAR SESIÓN



ENCABEZADO

Imprimir

Exportar

ITEM 1	ITEM 2	ITEM 3	ITEM 4	ITEM 5	ITEM 6	ITEM 7	ITEM 8

Figura 60 Estructura general actualizada de tablas

INICIO

ALUMNOS

DOCENTES

COLOQUIOS

CURSOS

CERRAR SESIÓN



Asignar Curso a Docentes

Seleccione el tipo de curso

Tutoria

Otro

Figura 61 Estructura general actualizada de la página Asignar Cursos a Docentes

[INICIO](#) [ALUMNOS](#) [DOCENTES](#) [COLOQUIOS](#) [CURSOS](#) [CERRAR SESIÓN](#)



Información de Asignación

[IMPRIMIR](#) [EXPORTAR](#)

Buscar Curso

[Buscar](#)

CURSO	XXXXXX	DOCENTE	XXXXXX	N° DE ALUMNOS INSCRITOS	XXXXXX	DOCENTE AUXILIAR	XXXXXX
-------	--------	---------	--------	-------------------------	--------	------------------	--------

Alumnos Inscritos

MATRICULA	NOMBRE
-----------	--------

Figura 62 Estructura general actualizada de la página Consultar Información de Asignación.

Programación

Clase Controlador

Función: Buscar Curso

Función que tiene como objetivo recibir la información ingresada en el formulario de búsqueda y posteriormente verificar si el curso ingresado ya cuenta con información asignada de docentes. Como primer paso recibe la información, posteriormente envía como parámetro el nombre del curso al método Validar Curso de la clase Métodos Modelo para verificar la información. Si en la base de datos ya existe información relacionada con la materia, convoca a la función Buscar Curso de la clase Métodos Modelo para obtener la información de la base de datos.

```
public static function C_buscarCurso(){
    if(isset($_POST["curso"])){
        $curso=$_POST["curso"];
        $tabla="docente_asignacion";

        #validar
        $filas=MetodosModelo::M_validarCurso($tabla,$curso);
        #echo $filas;

        if($filas==1){
            $respuesta=MetodosModelo::M_buscarCurso($tabla,$curso);
            return $respuesta;
        }
    }
}
```

Figura 63 Código del método Buscar Curso (clase controlador)

Función: Buscar Alumno

Función que tiene como objetivo recibir la información ingresada en el formulario de búsqueda y posteriormente verificar si el curso ingresado ya cuenta con información asignada de alumnos. Como primer paso recibe la información, posteriormente envía como parámetro el nombre del curso al método Validar Curso de la clase Métodos Modelo para verificar la información. Si en la base de datos ya existe información relacionada con la materia, convoca a la función Buscar Alumno de la clase Métodos Modelo para obtener la información de la base de datos.

```

public static function C_buscarAlumno(){

    if(isset($_POST["curso"])){

        $curso=$_POST["curso"];

        $tabla="alumno_asignacion";

        #validar
        $filas=MetodosModelo::M_validarCurso($tabla,$curso);

        if($filas>0){
            $respuesta=MetodosModelo::M_buscarAlumno($curso);
            return $respuesta;
        }

    }

}#end

```

Figura 64 Código del método Buscar Alumno (clase controlador)

Función: Selección Alumno

Función que tiene como objetivo buscar y almacenar la información relacionada con los rubros seleccionados de un estudiante. Como primer paso verifica la existencia del alumno mediante su matrícula utilizando el método Validar de la clase Métodos Modelo. Si en la base de datos ya existe información relacionada con el alumno, convoca a la función Selección de la clase Métodos Modelo para obtener los metadatos del estudiante. Por último, llama al método Almacenar Selección de la clase Métodos Modelo para guardar los datos obtenidos. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error.

```

public static function C_seleccionAlumno(){

    if(isset($_POST["id"])){

        $id=$_POST["id"];
        $tabla="alumno";
        $cont=0;

        #Valida la existencia del registro
        $filas=MetodosModelo::M_validar($id,$tabla);

        if($filas<=0){ ...
        }
        else{

            $respuesta=MetodosModelo::M_seleccion($tabla,$id);#Selecciona toda la inf del alumno

            foreach($respuesta as $row) { ...
            }

            if($cont==1){

                $datos = $matricula." ".$nombre." ".$direccion." ".$telefono." ".$celular." ".$correop." ".$correob." ".$correod;
                echo " cadena \n$datos";

                $bandera=MetodosModelo::M_almacenarSeleccion($datos);
            }

        }

    }

}

```

Figura 65 Código del método Selección Alumno (clase controlador)

Función: Selección Docente

Función que tiene como objetivo buscar y almacenar la información relacionada con los rubros seleccionados de un docente. Como primer paso verifica la existencia del docente en la base de datos mediante su id utilizando el método Validar de la clase Métodos Modelo. Si en la base de datos ya existe información relacionada con el docente, convoca a la función Selección de la clase Métodos Modelo para obtener los metadatos del catedrático. Por último, llama al método Almacenar Selección de la clase Métodos Modelo para guardar los datos obtenidos. De acuerdo con el estado resultante del proceso muestra un mensaje de éxito o error

```
public static function C_seleccionDocente(){
    if(isset($_POST["id"])){
        $id=$_POST["id"];
        $tabla="docente";
        $cont=0;

        #Valida la existencia del registro
        $filas=MetodosModelo::M_validar($id,$tabla);

        if($filas<=0){...
        }
        else{
            $respuesta=MetodosModelo::M_seleccion($tabla,$id);#Selecciona toda la inf del docente

            foreach($respuesta as $row) {...
            }

            if($cont==1){

                $datos = $codigo." ".$nombre." ".$telefono." ".$celular." ".$fecha." ".$facultad." ".$cubiculo." ".$;
                echo " cadena \n$datos";

                $bandera=MetodosModelo::M_almacenarSeleccion($datos);
            }
        }
    }
}
```

Figura 66 Código del método Selección Docente (clase controlador)

Función: Mostrar Selección

Función que invoca al método Mostrar Selección de la clase Métodos Modelo para la recuperación de información, una vez obtenidos los datos los retorna la interfaz gráfica.

```
public static function C_mostrarSeleccion(){
    $resultado=MetodosModelo::M_mostrarSeleccion();
    return $resultado;
}
```

Figura 67 Código del método Mostrar Selección (clase controlador)

Función: Limpiar Selección

Función que invoca al método Limpiar Selección de la clase Métodos Modelo para la eliminación de información.

```
public static function C_limpiarSeleccion(){  
    MetodosModelo::M_limpiarSeleccion();  
}  
#end
```

Figura 68 Código del método Limpiar Selección (clase controlador)

Clase Modelo

Función: Buscar Curso

Función que busca información de los docentes que impartirán un curso, como parámetro recibe el nombre de la materia. La selección de información en la base de datos se realiza mediante la sentencia SELECT utilizando PDO.

```
public static function M_buscarCurso($tabla,$curso){
    $con = Conexion::conectar()->prepare("SELECT * FROM $tabla WHERE curso=:curso;");
    $con -> bindParam(":curso", $curso, PDO::PARAM_STR);
    $con -> execute();
}
```

Figura 69 Código del método Buscar Curso (clase modelo)

Función: Buscar Alumno

Función que busca información de los alumnos inscritos en un curso, como parámetro recibe el nombre de la materia. La selección de información en la base de datos se realiza mediante las sentencias SELECT e INNER JOIN utilizando PDO.

```
public static function M_buscarAlumno($curso){
    $con = Conexion::conectar()->prepare("SELECT a.nombre,asig.matricula
    FROM alumno AS a INNER JOIN alumno_asignacion AS asig
    ON a.id=asig.matricula WHERE asig.curso=:curso;");
    $con -> bindParam(":curso", $curso, PDO::PARAM_STR);
    $con -> execute();

    $resultado = $con -> fetchAll();
    return $resultado;

    $resultado = null;
    $con = null;
}
```

Figura 70 Código del método Buscar Alumno (clase modelo)

Función: Almacenar Selección

Función que guarda información en un archivo de texto, si el archivo no existe lo crea con el modo de apertura "a", el cual permite añadir información en el documento en vez de sobrescribir en él.

```

public static function M_almacenarSeleccion($datos){

    $archivo = fopen('Documentos/archivo.txt','a');

    if( $archivo == false ){
        $bandera=0;
    }
    else{

        fwrite($archivo, $datos. PHP_EOL);

        // Fuerza a que se escriban los datos pendie
        fflush($archivo);

        fclose($archivo);
        $bandera=1;
    }

    return $bandera;
}#end

```

Figura 71 Código del método Almacenar Selección (clase modelo)

Función: Mostrar Selección

Función que recupera información de un archivo de texto, recupera línea por línea contenida en el documento y retorna el texto obtenido.

```

public static function M_mostrarSeleccion(){

    #Abre archivo
    $archivo = fopen("Documentos/archivo.txt", "r");

    #Recorremos todo el contenido del archivo
    $resultado="";
    while(!feof($archivo)){

        #Lee linea
        $linea = fgets($archivo,1024);

        $resultado .= $linea."<br>";

    }

    // Cerrando el archivo
    fclose($archivo);
    return $resultado;
}

```

Figura 72 Código del método Mostrar Selección (clase modelo)

Función: Limpiar Selección

Función que guarda información en un archivo de texto, el acceso al archivo es de modo “w”, el cual permite sobrescribir en él.

```
public static function M_limpiarSeleccion(){
    $datos=" ";
    $archivo = fopen('Documentos/archivo.txt','w');

    if( $archivo == false ) {
        echo "Error al crear el archivo";
    }
    else
    {
        fwrite($archivo, $datos. PHP_EOL);
        fwrite($archivo, PHP_EOL);
        // Fuerza a que se escriban los datos pendientes en el buffer:
        fflush($archivo);
    }
    // Cerrar el archivo:
    fclose($archivo);
}
}
```

Figura 73 Código del método Limpiar Selección (clase modelo)

Función: Validar Curso

Función que verificar si un curso tiene información asignada de alumnos o docentes. Retorna el número de filas relacionadas con la materia. La acción se realiza mediante la sentencia SELECT con PDO.

```
public static function M_validarCurso($tabla,$curso){

    $con = Conexion::conectar()->prepare("SELECT * FROM $tabla WHERE curso=:curso");
    $con -> bindParam(":curso", $curso, PDO::PARAM_STR);
    $con->execute();

    $filas = $con->rowCount();#Obtiene el numero de filas
    return $filas;
}#end
```

Figura 74 Código del método Validar Curso (clase modelo)

Interfaces de Usuario

La interfaz de usuario es el medio por el cual una persona controla una aplicación de software o dispositivo de hardware. Es decir, el programa incluye controles gráficos que optimizan la experiencia de usuario al emplear un mouse o teclado, lo que posibilita la interacción con los procesadores para realizar un trabajo; y su diseño juega un papel fundamental en la experiencia y satisfacción del usuario.

Página Inicio de Sesión

Página que permite al usuario iniciar sesión en la plataforma web. La presente ventana está compuesta por una imagen decorativa y por el formulario de login. La aplicación web. El formulario está conformado por el icono que representa la sección, el mensaje de bienvenida, dos componentes que reciben la información necesaria para iniciar sesión y el botón que ejecuta la acción.

El elemento que recibe el valor de la contraseña tiene la facultad de mostrar u ocultar la información recibida.



Figura 75 Interfaz de Inicio de Sesión

Página Principal

Página de bienvenida que se le despliega al usuario después de autenticarse correctamente en el sistema. Esta interfaz muestra la barra de navegación y el área de información sobre las diferentes secciones que conforman la aplicación web. Así como un mensaje de bienvenida al sitio.

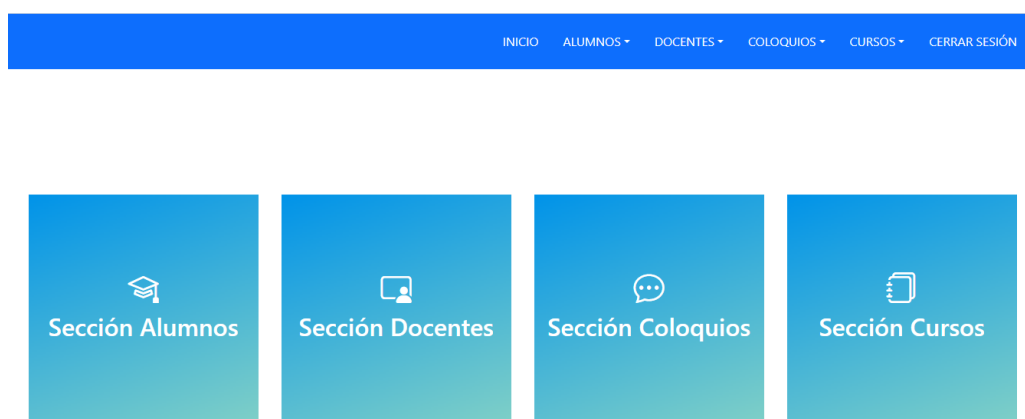


Figura 76 Interfaz página principal

Área de Información

Área compuesta por 4 elementos, donde cada uno tiene como objetivo mostrar una breve descripción de las acciones que se pueden realizar en las diferentes secciones del sistema.

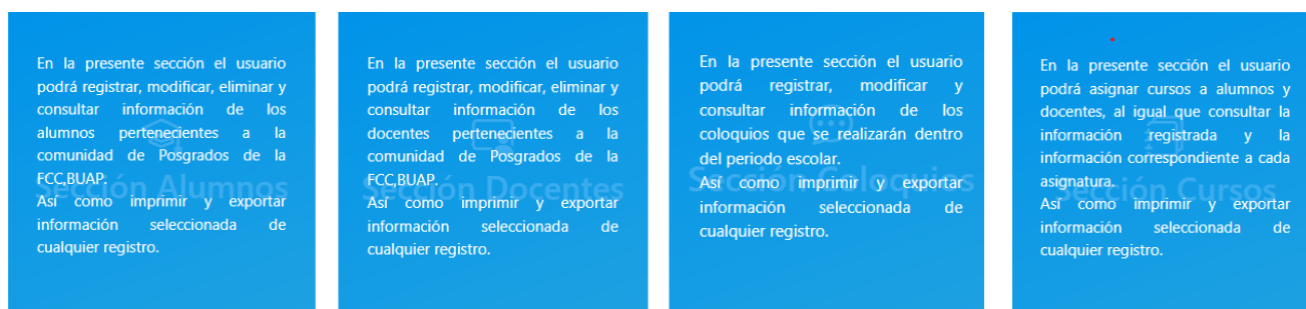


Figura 77 Área de información

Barra de Navegación

Componente formado por 6 elementos, 2 ítems y 4 menús, los cuales le permitirán al usuario moverse entre las diferentes secciones del sistema web.



Figura 78 Interfaz Barra de Navegación

Inicio: elemento que permite regresar a la página de bienvenida del sitio web.

Menú Alumnos: muestra las acciones que el usuario coordinador puede realizar en la sección de alumnos.

Registrar alumno

Eliminar alumno

Modificar información de alumno

Crear expediente de alumno

Alumnos registrados

Menú Docentes: despliega la lista de funciones que el usuario coordinador puede realizar en la sección de docentes.

Registrar docente

Eliminar docente

Modificar información de docente

Docentes registrados

Menú Coloquios: lista las diferentes opciones que el usuario coordinador puede realizar en la sección de coloquios.

Agendar coloquio

Modificar información de coloquio

Coloquios registrados

Menú Cursos: presenta el repertorio de acciones que el usuario coordinador puede realizar en la sección de cursos.

Asignar cursos a alumnos

Asignar cursos a docentes

Consultar información de asignación

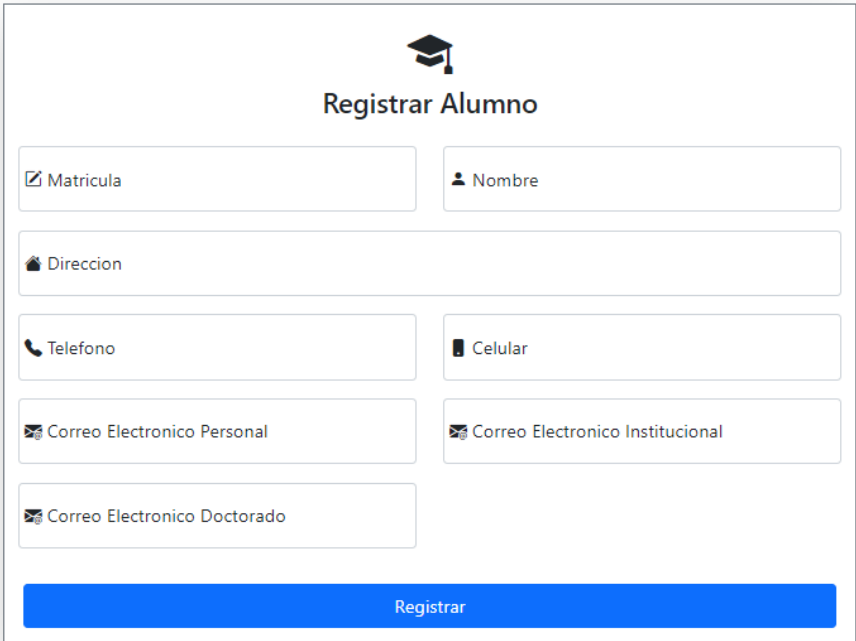
Lista de cursos

Cerrar sesión: elemento que permite finalizar la sesión iniciada por el usuario al momento de autenticarse en el sitio web.

Página Registrar Alumno

Página que permite registrar información correspondiente de un alumno a la base de datos del sistema. Los elementos que componen esta interfaz son el icono que representa a la sección y el formulario con los campos necesarios para ingresar la información solicitada por la institución educativa. De igual forma muestra la barra de navegación. El formulario está

compuesto de 8 elementos, los cuales contienen un icono distintivo según la información que soliciten, así como el botón para ejecutar la acción.



INICIO ALUMNOS ▾ DOCENTES ▾ COLOQUIOS ▾ CURSOS ▾ CERRAR SESIÓN


Registrar Alumno

 Matricula

 Nombre

 Direccion

 Telefono

 Celular

 Correo Electronico Personal

 Correo Electronico Institucional

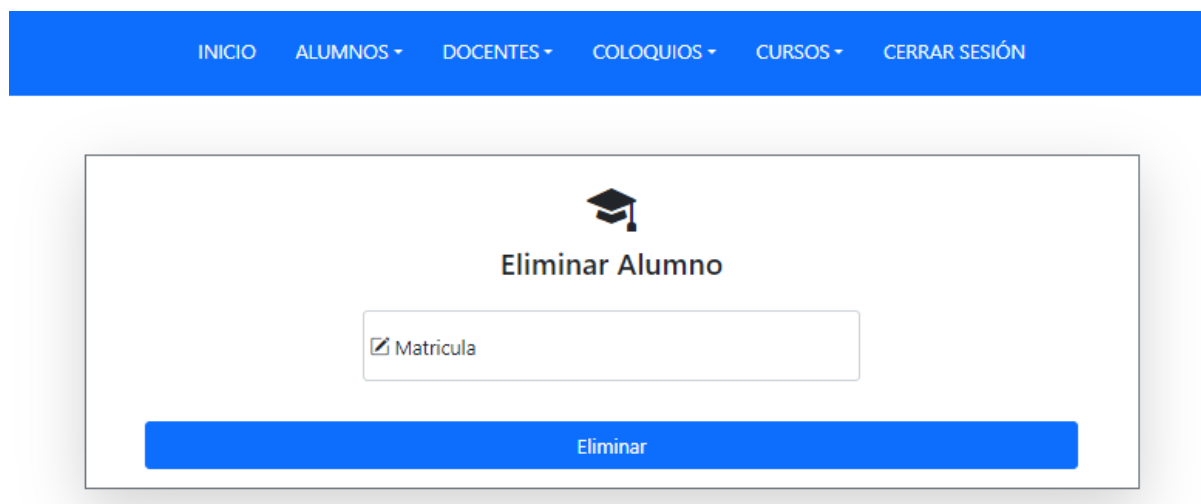
 Correo Electronico Doctorado

Registrar

Figura 79 Interfaz Registrar Alumno

Página Eliminar Alumno

Página que permite eliminar información relacionada con un alumno registrado en la plataforma. Los elementos que conforman la presente página son el icono que representa a la sección y el formulario de eliminación. El formulario está compuesto por un campo que solicita la matrícula del alumno a borrar, así como el botón para realizar la acción.



The screenshot shows a web interface with a blue header bar containing navigation links: INICIO, ALUMNOS, DOCENTES, COLOQUIOS, CURSOS, and CERRAR SESIÓN. Below the header is a white box with a black graduation cap icon and the title 'Eliminar Alumno'. Inside the box, there is a text input field with a checkmark icon and the label 'Matricula'. At the bottom of the box is a blue button labeled 'Eliminar'.

Figura 80 Interfaz Eliminar Alumno

Página Modificar Información de Alumno

Página que permite modificar la información de un alumno registrado en el sistema. Los componentes que conforman la presente página son el icono que representa a la sección y el formulario de modificación. El formulario está compuesto por tres inputs; el primer campo solicita la matrícula del alumno, el segunda muestra las opciones de los rubros válidos a modificar y el tercero se encarga de almacenar el nuevo valor del rubro seleccionado. Así mismo contiene el botón para ejecutar la acción.



The screenshot shows a web interface with a blue header bar containing navigation links: INICIO, ALUMNOS, DOCENTES, COLOQUIOS, CURSOS, and CERRAR SESIÓN. Below the header is a white box with a black graduation cap icon and the title 'Modificar Información de Alumno'. Inside the box, there are three text input fields, each with a checkmark icon and a label: 'Matricula', 'Rubro a Modificar', and 'Modificación'. The 'Rubro a Modificar' field has a dropdown arrow and the word 'Matricula' is visible. At the bottom of the box is a blue button labeled 'Modificar'.

Figura 81 Interfaz Modificar Información de Alumno

Página Crear Expediente de Alumno

Página que permite almacenar quince archivos PDF relacionados con un estudiante registrado en la plataforma web. El objetivo de esta interfaz es facilitar la creación del expediente del alumno. La página está compuesta por la barra de navegación y el formulario de registro. El segundo elemento está formado por un campo que solicita la matrícula del escolar, quince campos que reciben archivos PDF y el botón que ejecuta la acción.

INICIO

ALUMNOS

DOCENTES

COLOQUIOS

CURSOS

CERRAR SESIÓN

Crear Expediente de Alumno

Matricula

Seleccionar archivo

Sin archivos seleccionados

IFE/INE

Seleccionar archivo

Sin archivos seleccionados

Carta de recomendación N°1

Seleccionar archivo

Sin archivos seleccionados

Protocolo de Tesis

Seleccionar archivo

Sin archivos seleccionados

EXANIII

Seleccionar archivo

Sin archivos seleccionados

Título de Maestría

Seleccionar archivo

Sin archivos seleccionados

Cédula de Maestría

Seleccionar archivo

Sin archivos seleccionados

Publicaciones

Seleccionar archivo

Sin archivos seleccionados

Pago de Inscripción

Seleccionar archivo

Sin archivos seleccionados

Carta Compromiso

Seleccionar archivo

Sin archivos seleccionados

Carta de recomendación N°2

Seleccionar archivo

Sin archivos seleccionados

Comprobante TOEFL

Seleccionar archivo

Sin archivos seleccionados

Título de Licenciatura

Seleccionar archivo

Sin archivos seleccionados

Cédula de Licenciatura

Seleccionar archivo

Sin archivos seleccionados

CV

Seleccionar archivo

Sin archivos seleccionados

Kardex Simple

Seleccionar archivo

Sin archivos seleccionados

Crear

Figura 82 Interfaz Crear Expediente de Alumno

Página Alumnos Registrados

Página que permite visualizar la información almacenada de todos los alumnos registrados en el sistema. Los elementos que conforman la presente página son el icono que representa a la sección, el área de botones que sirven para exportar e imprimir información y la tabla de datos.


Alumnos Registrados

Imprimir / Exportar

Matrícula	Nombre	Dirección	Teléfono	Celular	Correo Personal	Correo BUAP	Correo Doctorado
201550602	Valentina Martínez Sanchez	Facultad Física	0000000000	1111111111	valentina@gmail.com	valentina@buap.com	valentina@doctorado.com
201611213	Camila González Cruz	29 Oriente 1605, PUEBLA, PUE.	2222304050	2226803797	camila@gmail.com	camila@buap.com	camila@doctorado.com
201373416	Ximena Jiménez Reyes	Calle 2 Poniente 2511, Amor, Puebla, Pue, C.P.	2221215060	2227131210	ximena@gmail.com	ximena@buap.com	ximena@doctorado.com
201831040	Victoria Flores Gómez	BLVD. Hermanos Serdán No. 279, Aquiles Serdán, Pu	2220503315	2228073613	victoria@gmail.com	victoria@buap.com	victoria@doctorado.com
201713121	Renata Martínez	Calle14 Sur 5126, Jardines De San Manuel	2219845614	2229310009	renata@gmail.com	renata@buap.com	renata@doctorado.com

Figura 83 Interfaz Alumnos Registrados

Selección de Información de Alumnos

Página que permite seleccionar información de un alumno registrado en el sistema, con el objetivo de imprimir o exportar los datos seleccionados. Los elementos que componen esta interfaz son el icono que representa a la sección y el formulario compuesto por el campo que solicita la matrícula del estudiante y de 8 elementos, los cuales permiten escoger los rubros a imprimir o exportar. De igual forma muestra la barra de navegación y el botón para ejecutar la acción.


Selección de Información de Alumnos

☒ Matrícula

☐ Matrícula

☐ Nombre

☐ Dirección

☐ Teléfono

☐ Celular

☐ Correo Personal

☐ Correo Buap

☐ Correo Doctorado

Enviar

Figura 84 Interfaz Selección de Información de Alumnos

Información Seleccionada

Página que muestra la información seleccionada para ser exportada o impresa. Los elementos que conforman la presente página son el área de botones que sirven para exportar, imprimir información y finalizar la operación.

Información Seleccionada

[Imprimir](#)
[Exportar](#)
[Finalizar](#)

201550602 Valentina Martinez Sanchez Facultad Física 0000000000 1111111111 valentina@gmail.com valentina@buap.com valentina@doctorado.com
 201373416 Calle 2 Poniente 2511, Amor , Puebla , Pue , C.P. 2221215060 ximena@doctorado.com
 201916226 Calle Ignacio Allende S/N, Centro , Teziutlán , P 2235500198 diego@gmail.com

Figura 85 Interfaz Información Seleccionada

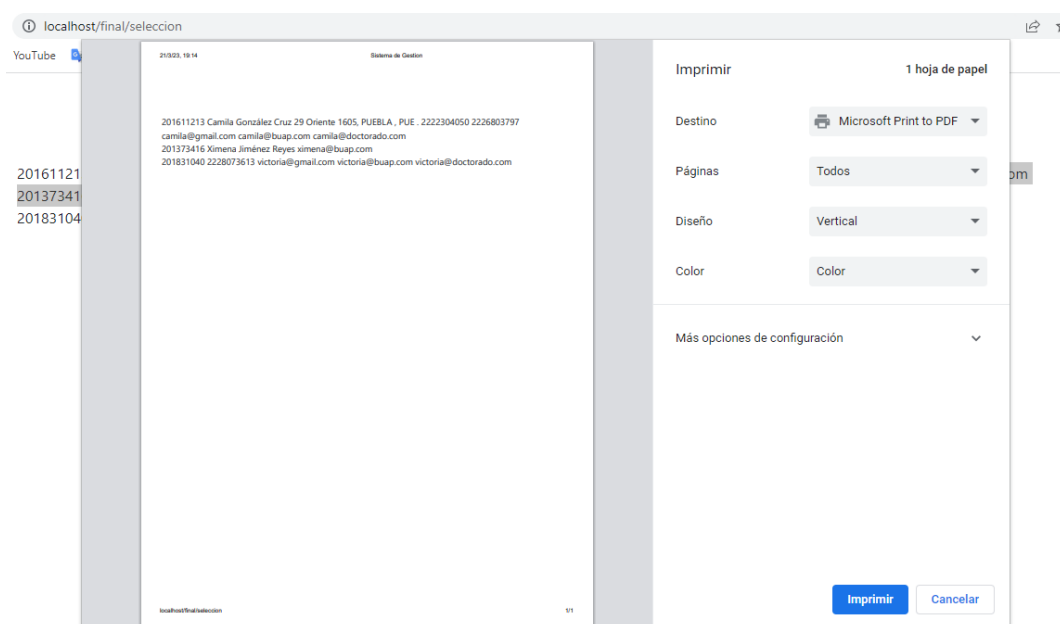


Figura 86 Extensión Imprimir

Página Eliminar Docente

Página que permite eliminar información perteneciente a un escolar registrado en la plataforma. La interfaz está compuesta por la barra de navegación, el icono que representa a la sección y el formulario que recibirá los datos de entrada. El último elemento está compuesto por un campo que solicita el ID del docente a eliminar, así como el botón para realizar la acción.

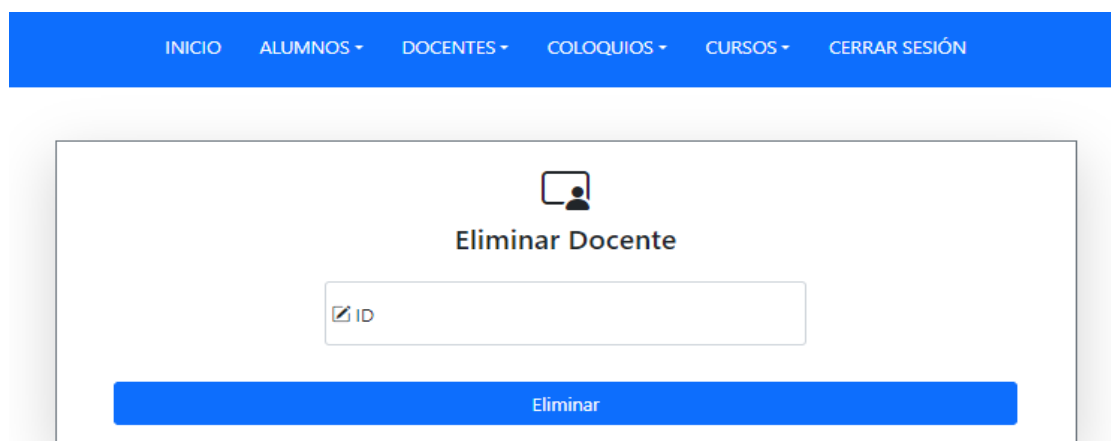


Figura 89 Interfaz Eliminar Docente

Página Modificar Información de Docente

Página que permite modificar la información de un docente registrado en la aplicación. Los componentes que conforman la presente página son la barra de navegación, el icono que representa a la sección y el formulario de modificación. El formulario está compuesto por tres inputs; el primer campo recibe el ID del docente, el segundo presenta una lista las opciones de las propiedades válidas a modificar y el último almacena el nuevo valor. Debajo del formulario aparece el botón que permitirá ejecutar la acción.

Figura 90 Interfaz Modificar Información de Docentes

Página Docentes Registrados

Página que muestra toda la información relacionada con los docentes registrados en el sitio web. La interfaz está formada por la barra de navegación, el icono que representa a la sección, el área de botones que permiten exportar e imprimir datos y la tabla de información.

ID	Nombre	Teléfono	Celular	Fecha de Nacimiento	Facultad de Adscripcion	Cubiculo	Correo BUAP	Correo VIEP	Tipo Tesista
IDM1G	María José García Rodríguez	2222222222	4444444444	2023-03-07	Física	CC05-3	maria@buap.com	jose@viep.com	Tesista 1
IDR2L	Regina López Ramirez	2211838891	2237198114	2023-03-30	Computacion	CC01-7	regina@buap.com	regina@viep.com	Tesista 2

Figura 91 Interfaz Docentes Registrados

Página Selección de Información de Docentes

Página que permite seleccionar información de un docente registrado en el sistema, con el objetivo de imprimir o exportar los datos seleccionados. Los elementos que componen esta interfaz son el icono que representa a la sección y el formulario compuesto por el campo que solicita el id del catedrático y de 10 elementos, los cuales permiten escoger los rubros a

imprimir o exportar. De igual forma muestra la barra de navegación y el botón para ejecutar la acción. La información seleccionada se visualiza en la Interfaz de Información Seleccionada.





Selección de Información de Docentes

☒ ID

☐ ID

☐ Nombre

☐ Teléfono

☐ Celular

☐ Cubiculo

☐ Tipo Tesista

☐ Correo BUAP

☐ Correo VIEP

☐ Fecha de Nacimiento

☐ Facultad de Adscripcion

Enviar

Figura 92 Interfaz Selección de Información de Docentes

Página Agendar Coloquios

Página que permite registrar la información correspondiente a los coloquios. Los elementos que integran la página son la barra de navegación, el icono que representa a la sección y el formulario de registro.

El formulario está integrado por 4 elementos, el primer campo recibe el título o nombre del coloquio, el segundo permite ingresar el lugar donde se efectuará la reunión, el tercero recibe la fecha y el último la hora en la que se llevará a cabo el coloquio. De igual forma contiene el botón que ejecutará la acción.

Figura 93 Interfaz Agendar Coloquio

Página Modificar Información de Coloquio

Página que permite modificar la información de un coloquio registrado en el sistema. Los componentes que conforman la presente página son la barra de navegación, el icono que representa a la sección y el formulario de modificación. El formulario está compuesto por tres inputs; el primer campo solicita el ID del coloquio, el segundo input muestra las opciones de las propiedades válidas a modificar y el tercero se encarga de almacenar el nuevo valor del rubro seleccionado. Así mismo contiene el botón para ejecutar la acción.

Figura 94 Interfaz Modificar Información de Coloquio

Página Coloquios Registrados

Interfaz que permite visualizar toda la información relacionada con los coloquios registrados en el sistema web. La página está compuesta por la barra de navegación, el icono que representa a la sección, el área de botones que permiten exportar e imprimir datos y la tabla de información.



Coloquios Registrados

Imprimir Exportar

ID	TITULO	FECHA	LUGAR	HORA
2	Coloquio4	2023-03-07	Facultad Física	14:30:00
3	Coloquio2	2023-02-14	Facultad Arquitectura	18:26:00
4	Coloquio3	2023-02-14	Facultad Administración	21:30:00
5	Coloquio1	2023-02-09	Facultad Química	19:34:00
6	Coloquio5	2023-03-16	Facultad Contaduría	15:37:00

Figura 95 Interfaz Coloquios Registrados

Página Asignar Curso a Alumnos

Página que permite registrar las matrículas de los alumnos solicitantes a una materia específica. La presente interfaz contiene la barra de navegación, el icono que representa a la sección y la lista de cursos académicos ofertada por la institución. De igual forma incluye un formulario interactivo que permite agregar y eliminar campos para la correcta inserción de información.

Figura 96 Interfaz Asignar Curso a Alumno

Página Asignar Curso a Docentes

Página que permite seleccionar el tipo de curso académico que se le asignará a un docente registrado en el sistema. La interfaz está compuesta por la barra de navegación, el icono representativo de la sección, y el área de botones. Los botones tienen como finalidad presentar el formulario correspondiente al tipo de curso que el usuario haya elegido.

Figura 97 Interfaz Asignar Curso a Docente

Formulario Tutoría de Investigación

La interfaz está compuesta por la barra de navegación, el icono representativo de la sección, y el formulario de registro. El formulario está formado por 4 elementos que reciben la información necesaria para asignar cualquier materia perteneciente a la rama de Tutoría de Investigación.

The screenshot shows a web interface with a blue navigation bar at the top containing links: INICIO, ALUMNOS +, DOCENTES +, COLOQUIOS +, CURSOS +, and CERRAR SESIÓN. Below the navigation bar is a white box titled 'Asignar Curso a Docente' with a pencil icon. Inside the box, there are four input fields: a dropdown menu for 'Curso' (selected: Tutoría de Investigación I), a text field for 'Nº de Solicitantes', a text field for 'ID Docente', and a text field for 'Docente' with a person icon. At the bottom of the box is a blue button labeled 'Asignar'.

Figura 98 Interfaz formulario de Asignación de Tutoría

Formulario Otros Cursos

La interfaz muestra la barra de navegación, el icono representativo de la sección, y el formulario de registro. El formulario está compuesto por 6 elementos que reciben la información necesaria, para asignar cualquier materia que no pertenezca a la rama de Tutoría de Investigación.

The screenshot shows a web interface with a blue navigation bar at the top containing links: INICIO, ALUMNOS +, DOCENTES +, COLOQUIOS +, CURSOS +, and CERRAR SESIÓN. Below the navigation bar is a white box titled 'Asignar Curso a Docente' with a pencil icon. Inside the box, there are six input fields: a dropdown menu for 'Curso' (selected: Disertación Aplicada I), a text field for 'Nº de Solicitantes', a text field for 'ID Docente', a text field for 'Docente' with a person icon, a text field for 'ID Docente Auxiliar', and a text field for 'Docente Auxiliar' with a person icon. At the bottom of the box is a blue button labeled 'Asignar'.

Figura 99 Interfaz formulario para Asignar Materias

Página Consultar Información de Asignación

Página que permite visualizar toda la información referente a la asignación de un curso académico. Los elementos que integran a la interfaz son, la barra de navegación, el icono representativo de la sección y el área de botones que permiten exportar e imprimir datos.

Asimismo, incluye un formulario que permite buscar información de la asignación. Seguido de este elemento muestra la tabla de datos.



Información de Asignación

Q Buscar Curso

Disertación Aplicada I

Curso	Docente	N° de alumnos solicitantes	Docente auxiliar
Alumnos Inscritos			
Matricula	Nombre		

Figura 100 Interfaz Información de Asignación

Página Lista de Cursos

Página que muestra toda la información relacionada con los cursos académicos que ofrece el área de posgrados de la universidad. La interfaz está formada por la barra de navegación, el icono que representa a la sección, el área de botones que permiten exportar e imprimir datos y la tabla de información.



Lista de Cursos

[Imprimir](#)[Exportar](#)

ID	NOMBRE	REQUISITO
DSAE 10100	Seminario de Investigación I	No
DSAE 10200	Tutoría de Investigación I	No
DSAE 10300	Seminario de investigación II	DSAE 10100
DSAE 10400	Tutoría de Investigación II	DSAE 10200
DSAE 10500	Disertación Aplicada I	No
DSAE 10600	Divulgación del conocimiento I	No
DSAE 10700	Seminario de Investigación III	DSAE 10300
DSAE 10800	Tutoría de investigación III	DSAE 10400
DSAE 10900	Seminario de Investigación IV	DSAE 10700

Figura 101 Interfaz Lista de Cursos

Capítulo 5

Pruebas

Pruebas de Caja Negra

Iniciar Sesión

Condiciones

Si el usuario y contraseña ingresados son válidos, el sistema da acceso al usuario a la plataforma.

Si el usuario y contraseña no son válidos, el sistema no permite el acceso a la plataforma.

Si el nombre de usuario es válido y la contraseña es errónea, el sistema no permite el acceso a la plataforma.

Si el nombre de usuario es erróneo y la contraseña es válida, el sistema no permite el acceso a la plataforma.

Entrada: Usuario y Contraseña

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 26 Tabla de decisión Iniciar Sesión

Condiciones	Regla 1	Regla 2	Regla 3	Regla 4
Usuario	V	V	F	F
Contraseña	V	F	V	F
Resultado	PE	PF	PF	PF

Evidencia



Bienvenido

👤 Usuario

👤 Contraseña



Iniciar Sesión

Figura 102 Iniciar Sesión r1 entrada

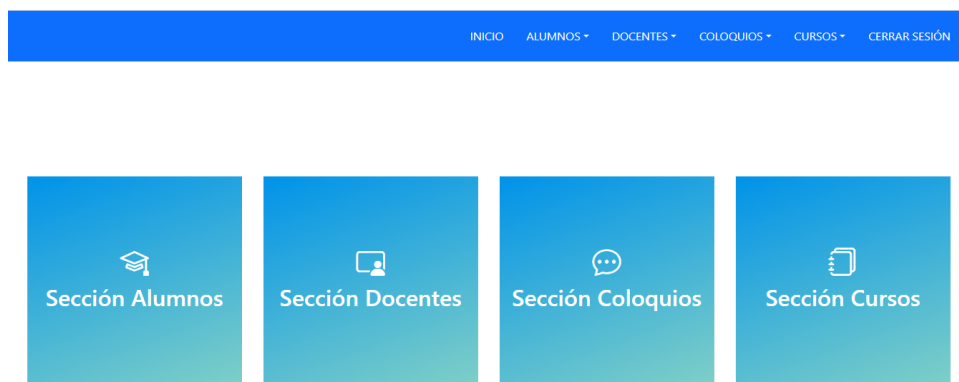


Figura 103 Iniciar Sesión r1 salida



Bienvenido

👤 Usuario

👤 Contraseña



Iniciar Sesión

Figura 104 Iniciar Sesión r2,r3,r4 entrada

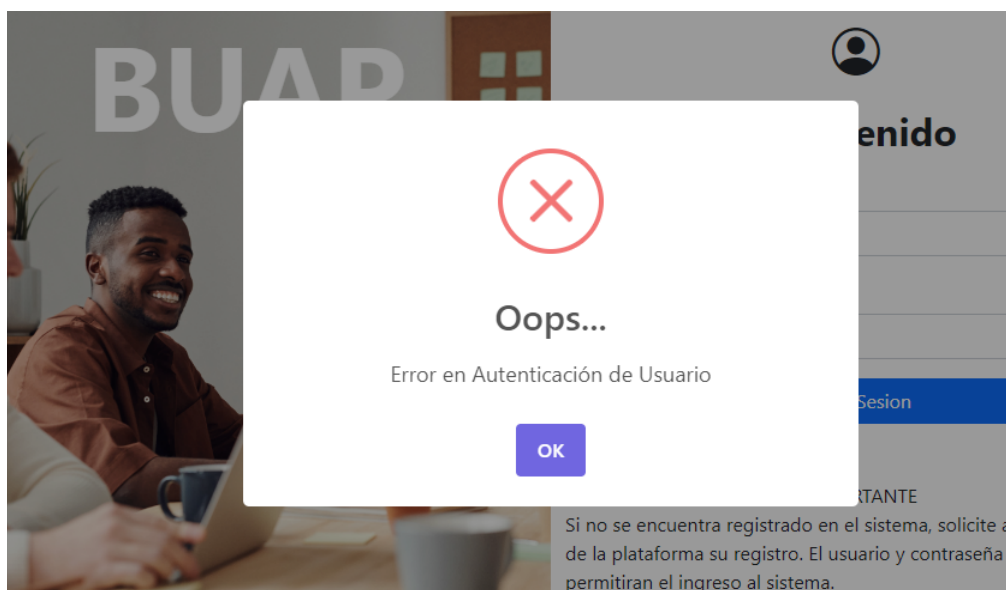


Figura 105 Iniciar Sesión r2,r3,r4 salida

Registrar Alumno

Condiciones

Si la matrícula ya existe en el sistema, la plataforma web no permite el registro del alumno.

Si la matrícula no existe en el sistema, la plataforma web permite el registro del alumno.

Entrada: Matrícula, Nombre, Dirección, Teléfono, Celular, Email personal, Email Institucional y Email del doctorado.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 27 Tabla de decisión Registrar Alumno

Condiciones	Regla 1	Regla 2
Matrícula	V	F
Resultado	PF	PE

Evidencia


Registrar Alumno

☒ Matricula 201611213	Nombre Camila González Cruz
Direccion 29 Oriente 1605, PUEBLA , PUE .	
Telefono 2222304050	Celular 2226803797
Correo Electronico Personal camila@gmail.com	Correo Electronico Institucional camila@buap.com
Correo Electronico Doctorado camila@doctorado.com	

Registrar

Figura 106 Registrar Alumno r1 entrada

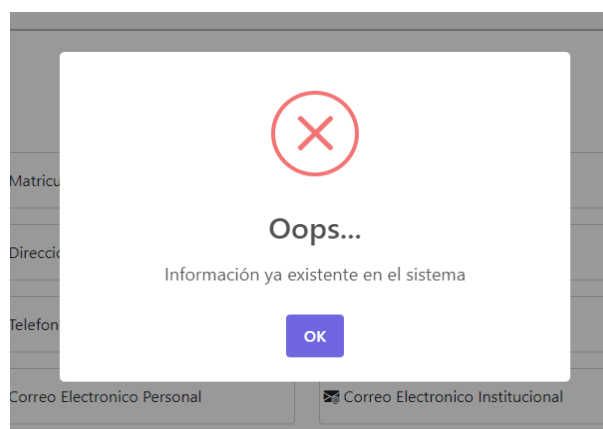
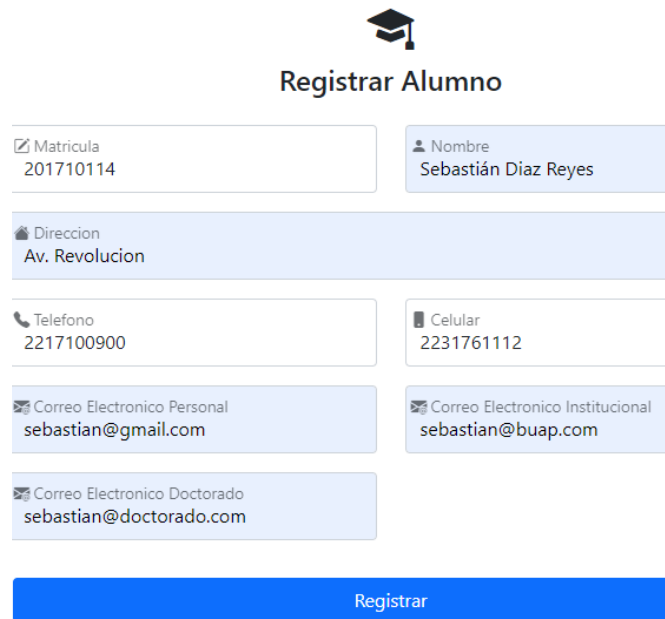


Figura 107 Registrar alumno r1 salida




Registrar Alumno

☒ Matricula 201710114	Nombre Sebastián Díaz Reyes
Direccion Av. Revolucion	
Telefono 2217100900	Celular 2231761112
Correo Electronico Personal sebastian@gmail.com	Correo Electronico Institucional sebastian@buap.com
Correo Electronico Doctorado sebastian@doctorado.com	
Registrar	

Figura 108 Registrar alumno r2 entrada



Figura 109 Registrar alumno r2 salida

Modificar Información de Alumno

Condiciones

Si la matrícula ya existe en el sistema, la plataforma web realiza la operación.

Si la matrícula no existe en el sistema, la plataforma web no realiza la modificación de información.

Entrada: Matricula, Rubro a modificar y Modificación

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 28 Tabla de decisión Modificar Información de Alumno

Condiciones	Regla 1	Regla 2
Matricula	V	F
Resultado	PF	PF

Evidencia



Modificar Información de Alumno

☒ Matricula
201611213

☒ Rubro a Modificar
Telefono Particular

☒ Modificación
1111111111

Modificar

Figura 110 Modificar inf. de alumno r1 entrada

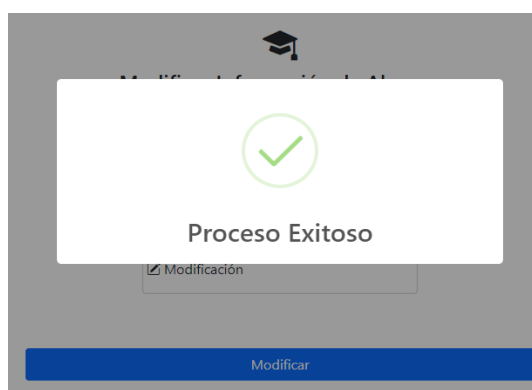


Figura 111 Modificar inf. de alumno r1 salida



Modificar Información de Alumno

☒ Matricula
123456789

☒ Rubro a Modificar
Nombre

☒ Modificación
Juan González Martínez

Modificar

Figura 112 Modificar inf. de alumno r2 entrada

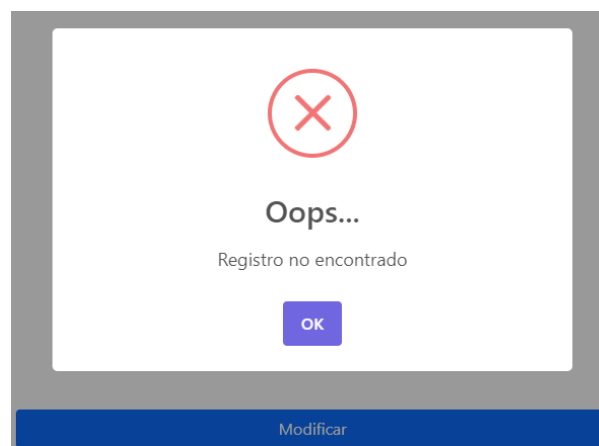


Figura 113 Modificar inf. de alumno r2 salida

Eliminar Alumno

Condiciones

Si la matrícula ya existe en el sistema, la plataforma web ejecuta la operación.

Si la matrícula no existe en el sistema, la plataforma web no realiza la eliminación de información.

Entrada: Matricula de alumno.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 29 Tabla de decisión Eliminar Alumno

Condiciones	Regla 1	Regla 2
Matricula	V	F
Resultado	PE	PF

Evidencia



Eliminar Alumno

☒ Matricula
201710114

Eliminar

Figura 114 Eliminar alumno r1 entrada



Figura 115 Eliminar alumno r1 salida



Eliminar Alumno

☒ Matricula
 201710114

Eliminar

Figura 116 Eliminar alumno r2 entrada

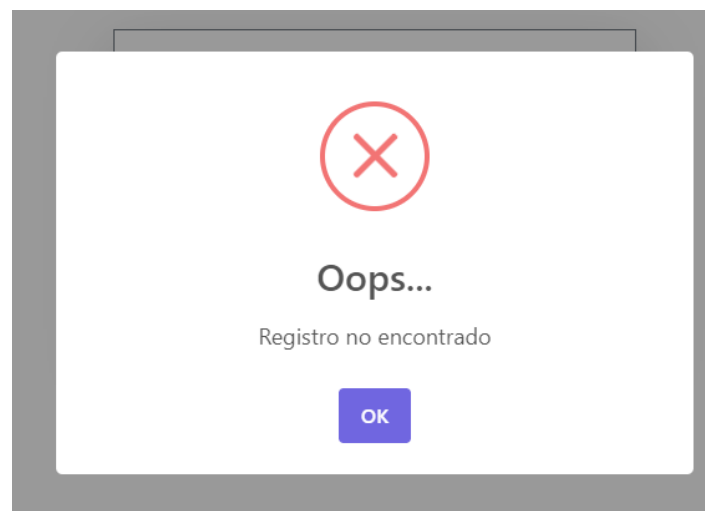


Figura 117 Eliminar alumno r2 salida

Crear Expediente de Alumno

Condiciones

Si la matrícula ya existe en el sistema, la plataforma web ejecuta la operación.

Si la matrícula no existe en el sistema, la plataforma web no registra la información.

Entrada: Matrícula de alumno y archivos PDF.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 30 Tabla de decisión Crear Expediente de Alumno

Condiciones	Regla 1	Regla 2
Matricula	V	F
Resultado	PF	PF

Evidencia




Crear Expediente de Alumno

☒ Matricula
201611213

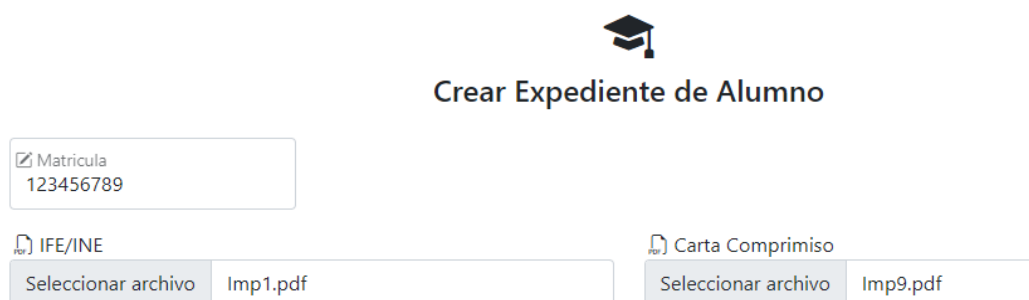
 IFE/INE  Carta Compromiso

Seleccionar archivo Imp1.pdf Seleccionar archivo Imp9.pdf

Figura 118 Crear expediente de alumno r1 entrada



Figura 119 Crear expediente de alumno r1 salida




Crear Expediente de Alumno

☒ Matricula
123456789

 IFE/INE  Carta Compromiso

Seleccionar archivo Imp1.pdf Seleccionar archivo Imp9.pdf

Figura 120 Crear expediente de alumno r2 entrada

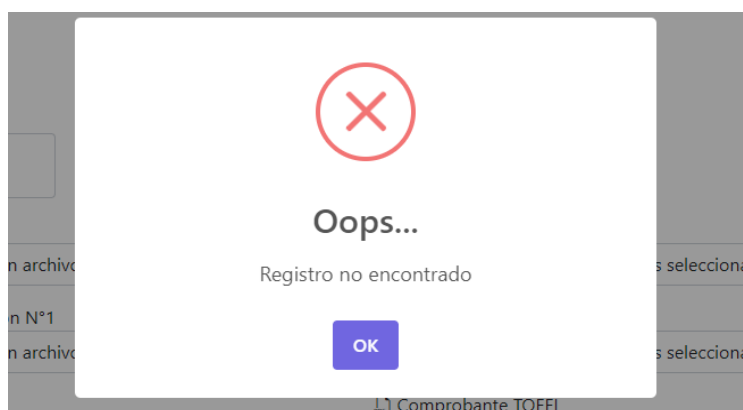


Figura 121 Crear expediente de alumno r2 salida

Seleccionar Información de Alumno

Condiciones

Si la matrícula ya existe en el sistema, la plataforma web ejecuta la operación.

Si la matrícula no existe en el sistema, la plataforma web no almacena la información seleccionada.

Entrada: Matricula de alumno.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 31 Tabla de decisión Seleccionar Información de Alumno

Condiciones	Regla 1	Regla 2
Matricula	V	F
Resultado	PF	PF

Evidencia



Selección de Información de Alumnos

☒ Matricula
201611213

☒ Matricula

☐ Nombre

☒ Dirección

☐ Teléfono

☐ Celular

☒ Correo Personal

☐ Correo Buap

☒ Correo Doctorado

Enviar

Figura 122 Selección inf. de alumnos r1 entrada



Selección de Información de Alumnos

Proceso exitoso

Desea seguir seleccionado información?

Finalizar

Seguir

Figura 123 Selección inf. de alumnos r1 salida



Selección de Información de Alumnos

☒ Matricula
123456789

☒ Matricula

☒ Nombre

☐ Dirección

☒ Teléfono

☒ Celular

☒ Correo Personal

☐ Correo Buap

☐ Correo Doctorado

Enviar

Figura 124 Selección inf. de alumnos r2 entrada

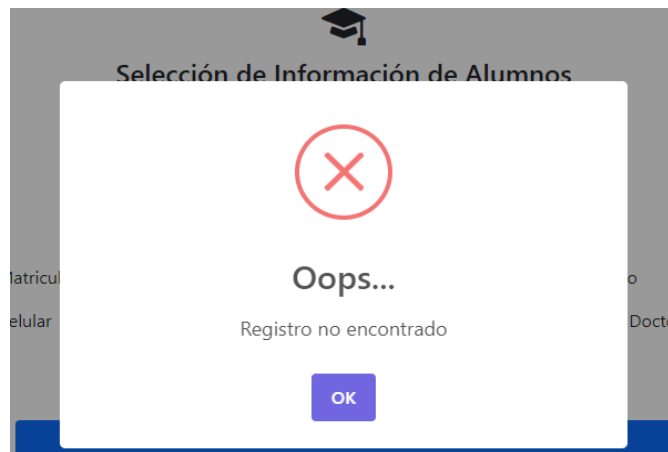


Figura 125 Selección inf. de alumnos r2 salida

Registrar Docentes

Condiciones

Si el ID del docente ya existe en el sistema, la plataforma web no permite el registro del catedrático.

Si el ID del docente no existe en el sistema, la plataforma web permite el registro del catedrático.

Entrada: Id, Nombre, Teléfono, Celular, Fecha de Nacimiento, Facultad de Adscripción, Cubículo, Tipo Tesista, Email Institucional y Email VIEP.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 32 Tabla de decisión Registrar Docentes

Condiciones	Regla 1	Regla 2
ID	V	F
Resultado	PF	PE

Evidencia


Registrar Docente

☒ ID IDD8C	Nombre Miguel Ángel Catillo Romero
Telefono 2223310408	Celular 2240413432
Fecha de Nacimiento 24/01/1998	Facultad de Adscripción Computación
Cubiculo CC02-9	Tipo Tesista Tesista1
Correo Electronico Institucional miguelAngel@buap.com	Correo Electronico VIEP miguelAngel@viep.com

Registrar

Figura 126 Registrar docente r1 entrada

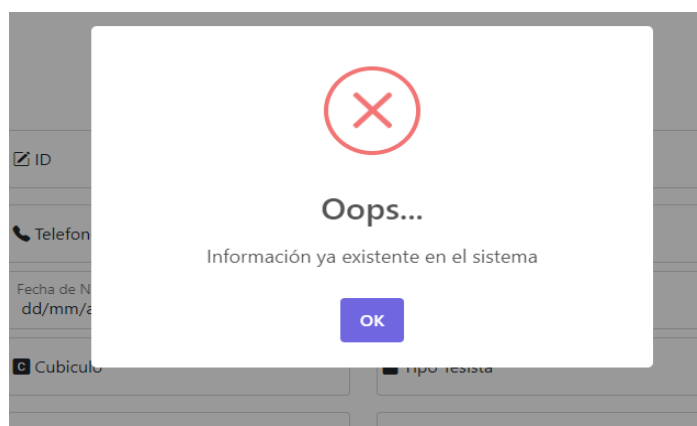


Figura 127 Registrar docente r1 salida

Registrar Docente

ID IDL6T	Nombre Leonardo Torres Morales
Telefono 2229310006	Celular 2210773416
Fecha de Nacimiento 24/06/1997	Facultad de Adscripción Computación
Cubiculo CC02-11	Tipo Tesista Tesista1
Correo Electronico Institucional leonardo@buap.com	Correo Electronico VIEP leonardo@viep.com

Registrar

Figura 128 Registra docente r2 entrada



Figura 129 Registrar docente r2 salida

Modificar Información de Docente

Condiciones

Si el ID del docente ya existe en el sistema, la plataforma web realiza la operación.

Si el ID del docente no existe en el sistema, la plataforma web no realiza la modificación de información.

Entrada: ID, Rubro a modificar y Modificación

Salida: Mensaje que notifica el resultado de la operación.

Tabla de decisión

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 33 Tabla de decisión Modificar Información de Docente

Condiciones	Regla 1	Regla 2
ID	V	F
Resultado	PF	PF

Evidencia



Modificar Información de Docente

Nota: Formatos a seguir
Fecha de Nacimiento: AAAA-MM-DD

☒ ID
IDD8C

☒ Rubro a Modificar
Cubículo

☒ Modificación
CC04-2

Modificar

Figura 130 Modificar inf. de docente r1 entrada

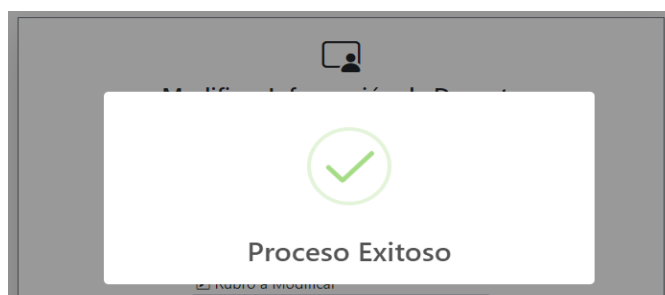


Figura 131 Modificar inf. de docente r1 salida



Modificar Información de Docente

Nota: Formatos a seguir
Fecha de Nacimiento: AAAA-MM-DD

☒ ID
 IDDDD

☒ Rubro a Modificar
 Celular

☒ Modificación
 2211583036

Modificar

Figura 132 Modificar inf. de docente r2 entrada

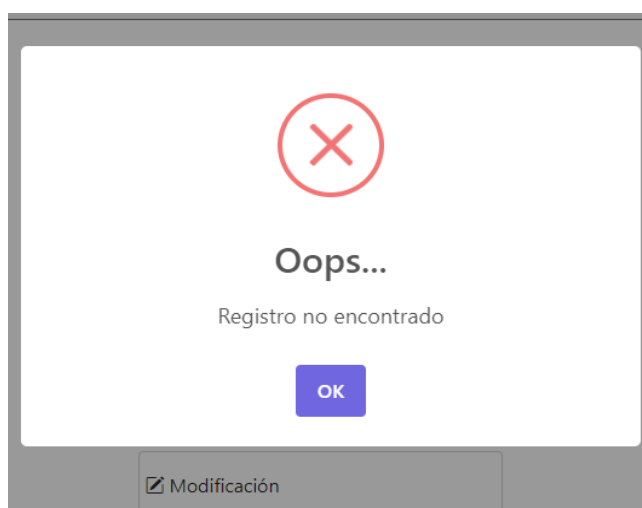


Figura 133 Modificar inf. de docente r2 salida

Eliminar Docente

Condiciones

Si el ID del docente ya existe en el sistema, la plataforma web ejecuta la operación.

Si el ID del docente no existe en el sistema, la plataforma web no realiza la eliminación de información.

Entrada: ID de docente.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 34 Tabla de decisión Eliminar Docente

Condiciones	Regla 1	Regla 2
ID	V	F
Resultado	PF	PF

Evidencia



Eliminar Docente

☒ ID
 IDL6T

Eliminar

Figura 134 Eliminar docente r1 entrada



Figura 135 Eliminar docente r1 salida

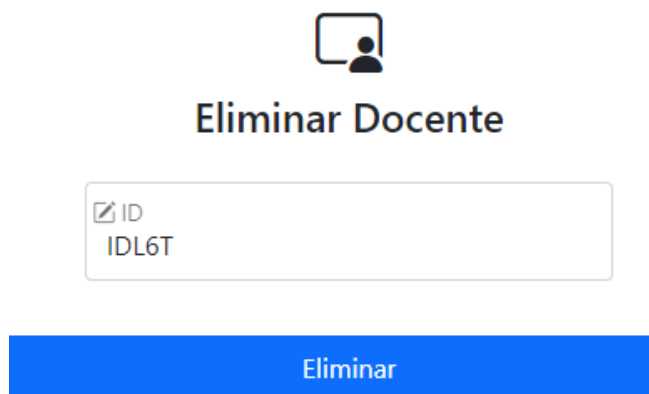


Figura 136 Eliminar docente r2 entrada

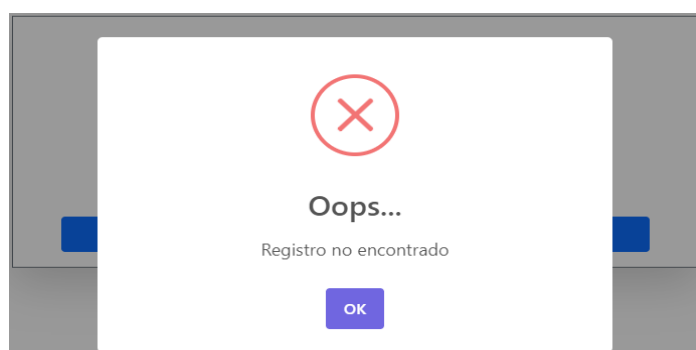


Figura 137 Eliminar docente r2 salida

Seleccionar Información de Docente

Condiciones

Si el ID del docente ya existe en el sistema, la plataforma web ejecuta la operación.

Si el ID del docente no existe en el sistema, la plataforma web no almacena la información seleccionada.

Entrada: ID de docente.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 35 Tabla de decisión Seleccionar Información de Docente

Condiciones	Regla 1	Regla 2
ID	V	F
Resultado	PF	PF

Evidencia



Selección de Información de Docentes

☒ ID
IDM1G

☒ ID	☒ Nombre	☒ Teléfono	☒ Celular
☒ Cubiculo	☒ Tipo Tesista	☒ Correo BUAP	☒ Correo VIEP
☒ Fecha de Nacimiento	☒ Facultad de Adscripcion		

Enviar

Figura 138 Selección de inf. de docente r1 entrada



Selección de Información de Docentes

Proceso exitoso

Desea seguir seleccionado información?

Finalizar

Seguir

Enviar

Figura 139 Selección de inf. de docente r1 salida



Selección de Información de Docentes

☒ ID
IDDDD

☒ ID	☒ Nombre	☐ Teléfono	☐ Celular
☐ Cubiculo	☒ Tipo Tesista	☐ Correo BUAP	☒ Correo VIEP
☐ Fecha de Nacimiento	☐ Facultad de Adscripcion		

Enviar

Figura 140 Selección de inf. de docentes r2 entrada

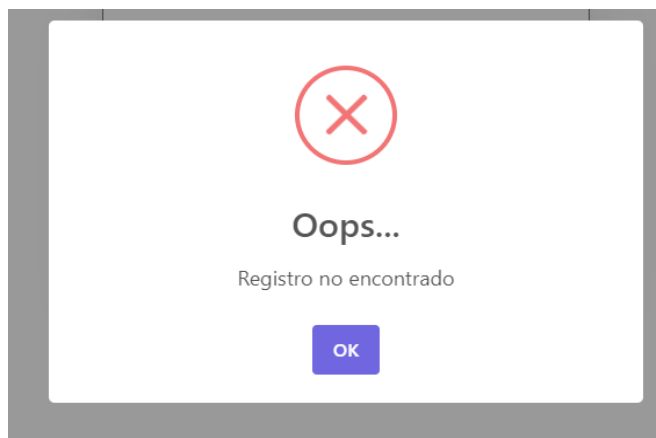


Figura 141 Selección de inf. de docente r2 salida

Modificar información de Coloquio

Condiciones

Si el ID correspondiente al coloquio ya existe en el sistema, la plataforma web realiza la operación.

Si el ID correspondiente al coloquio no existe en el sistema, la plataforma web no realiza la modificación de información.

Entrada: ID, Rubro a modificar y Modificación

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 36 Tabla de decisión Modificar Información de Coloquio

Condiciones	Regla 1	Regla 2
ID	V	F
Resultado	PF	PF

Evidencia



Modificar Información de Coloquio

Nota: Formatos a seguir
Fecha: AAAA-MM-DD
Hora: HH:MM:SS (Formato 24 horas)

☒ ID	2
☒ Rubro a Modificar	Título
☒ Modificación	Reunión

Modificar

Figura 142 Modificar inf. de coloquio r1 entrada

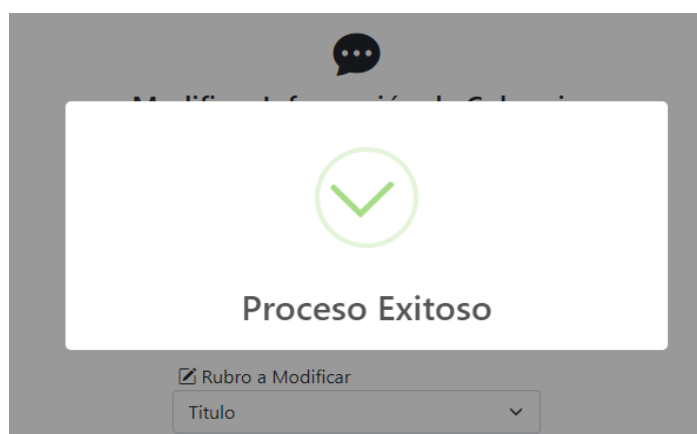


Figura 143 Modificar inf. de coloquio r1 salida



Modificar Información de Coloquio

Nota: Formatos a seguir
 Fecha: AAAA-MM-DD
 Hora: HH:MM:SS (Formato 24 horas)

☒ ID
 10

☒ Rubro a Modificar
 Hora

☒ Modificación
 18:00:00

Modificar

Figura 144 Modificar inf. de coloquio r2 entrada

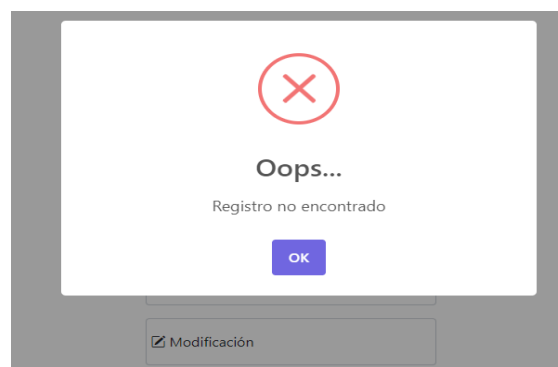


Figura 145 Modificar inf. de coloquio r2 salida

Asignar Curso a Alumno

Condiciones

Si la matrícula no existe en el sistema, la plataforma web no permite la inscripción del alumno al curso seleccionado.

Si la matrícula ya existe en el sistema, la plataforma web permite la inscripción del alumno al curso seleccionado.

Entrada: Nombre del curso y Matrículas de los alumnos aspirantes.

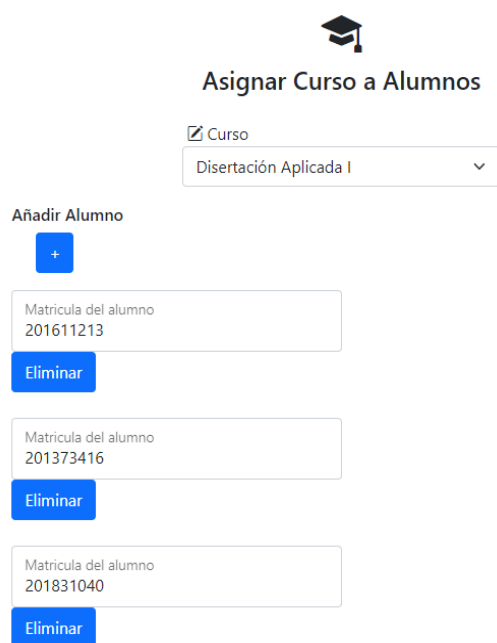
Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 37 Tabla de decisión Asignar Curso a Alumno

Condiciones	Regla 1	Regla 2
Matricula	V	F
Resultado	PF	PE

Evidencia




Asignar Curso a Alumnos

☒ Curso
Disertación Aplicada I

Añadir Alumno

Matricula del alumno
201611213

Matricula del alumno
201373416

Matricula del alumno
201831040

Figura 146 Asignar Curso a Alumnos r1 entrada

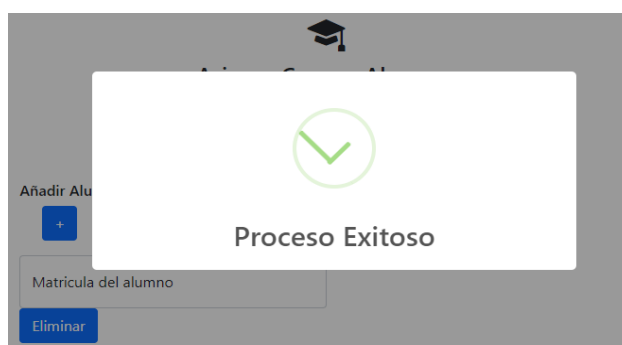


Figura 147 Asignar Curso a Alumnos r1 salida



Asignar Curso a Alumnos

☒ Curso

Tutoría de Investigación III

Añadir Alumno

+

Matricula del alumno
11111111

Eliminar

Matricula del alumno
201401843

Eliminar

Asignar

Figura 148 Asignar Curso a Alumnos r2 entrada

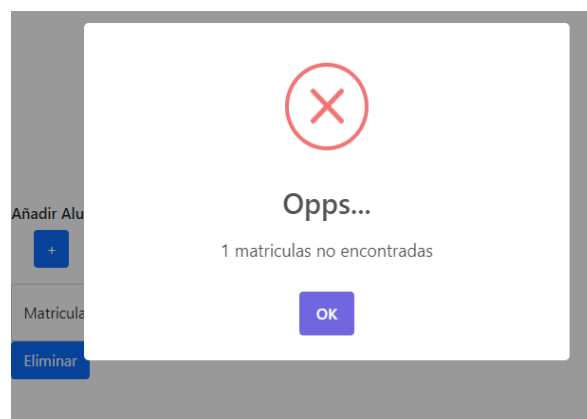


Figura 149 Asignar Curso a Alumnos r2 salida

Asignar Curso a Docente

Condiciones

Si el ID del docente no existe en el sistema, la plataforma web no permite la asignación del docente al curso seleccionado.

Si el ID del docente ya existe en el sistema, la plataforma web permite la asignación del docente al curso seleccionado.

Entrada: Nombre del curso e ID del docente.

Salida: Mensaje que notifica el resultado de la operación.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 38 Tabla de decisión Asignar Curso a Docente

Condiciones	Regla 1	Regla 2
ID	V	F
Resultado	PF	PE

Evidencia



Asignar Curso a Docente

☒ Curso

Tutoría de Investigación V

☒ N° de Solicitantes

5

☒ ID Docente

IDM1G

☐ Docente

María José García Rodríguez

Asignar

Figura 150 Asignar Curso a Docente r1 entrada

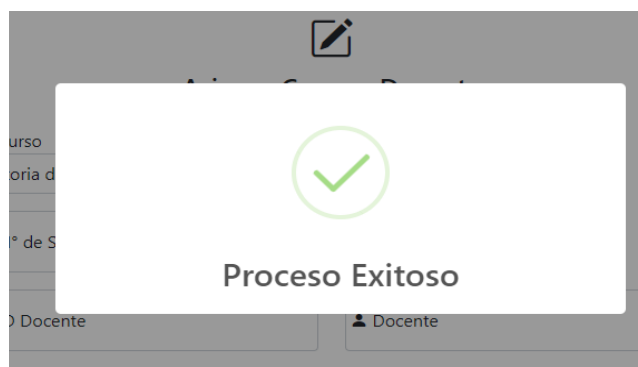


Figura 151 Asignar Curso a Docente r1 salida



Asignar Curso a Docente

☒ Curso

Tutoría de Investigación III

☒ N° de Solicitantes

20

☒ ID Docente

IDDDD

☐ Docente

Oscar Ramirez

Asignar

Figura 152 Asignar Curso a Docente r2 entrada

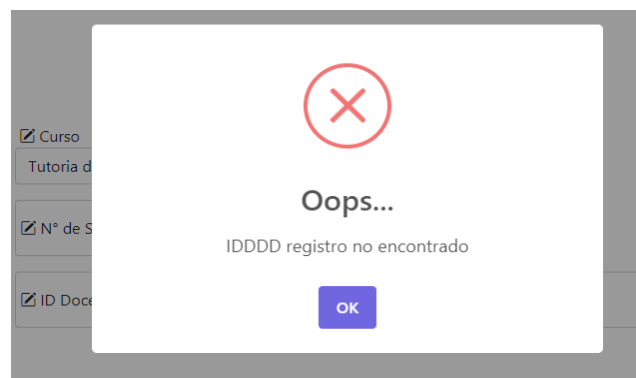


Figura 153 Asignar Curso a Docente r2 salida

Buscar Información de Asignación

Condiciones

Si el nombre del curso seleccionado se encuentra relacionado con información de docentes o alumnos, la plataforma web muestra información.

Si el nombre del curso seleccionado no se encuentra relacionado con información de docentes o alumnos, la plataforma web no muestra información.

Entrada: Curso

Salida: Información asociada a los docentes y alumnos asignados al curso.

F=Falso V=Verdadero PF= Proceso fallido PE= Proceso exitoso

Tabla 39 Tabla de decisión Buscar Información de Asignación

Condiciones	Regla 1	Regla 2
Curso	V	F
Resultado	PE	PF

Evidencia

Q Buscar Curso

Tutoría de Investigación V

▼

Buscar

Curso	Docente	N° de alumnos solicitantes	Docente auxiliar
Tutoría de Investigación V	María José García Rodríguez	5	

Alumnos Inscritos

Matrícula	Nombre
201611213	Camila González Cruz
201373416	Ximena Jiménez Reyes
201831040	Victoria Flores Gómez
201401843	Santiago Mendoza Flores
201916226	Diego Álvarez Ramos

Figura 154 Buscar Información de Asignación r1 entrada y salida

i

Información de Asignación

Imprimir

Exportar

Q Buscar Curso

Seminario de Investigación III

▼

Buscar

Curso	Docente	N° de alumnos solicitantes	Docente auxiliar
-------	---------	----------------------------	------------------

Alumnos Inscritos

Matrícula	Nombre
-----------	--------

Figura 155 Buscar Información de Asignación r2 entrada y salida

Pruebas de Caja Blanca

Iniciar sesión

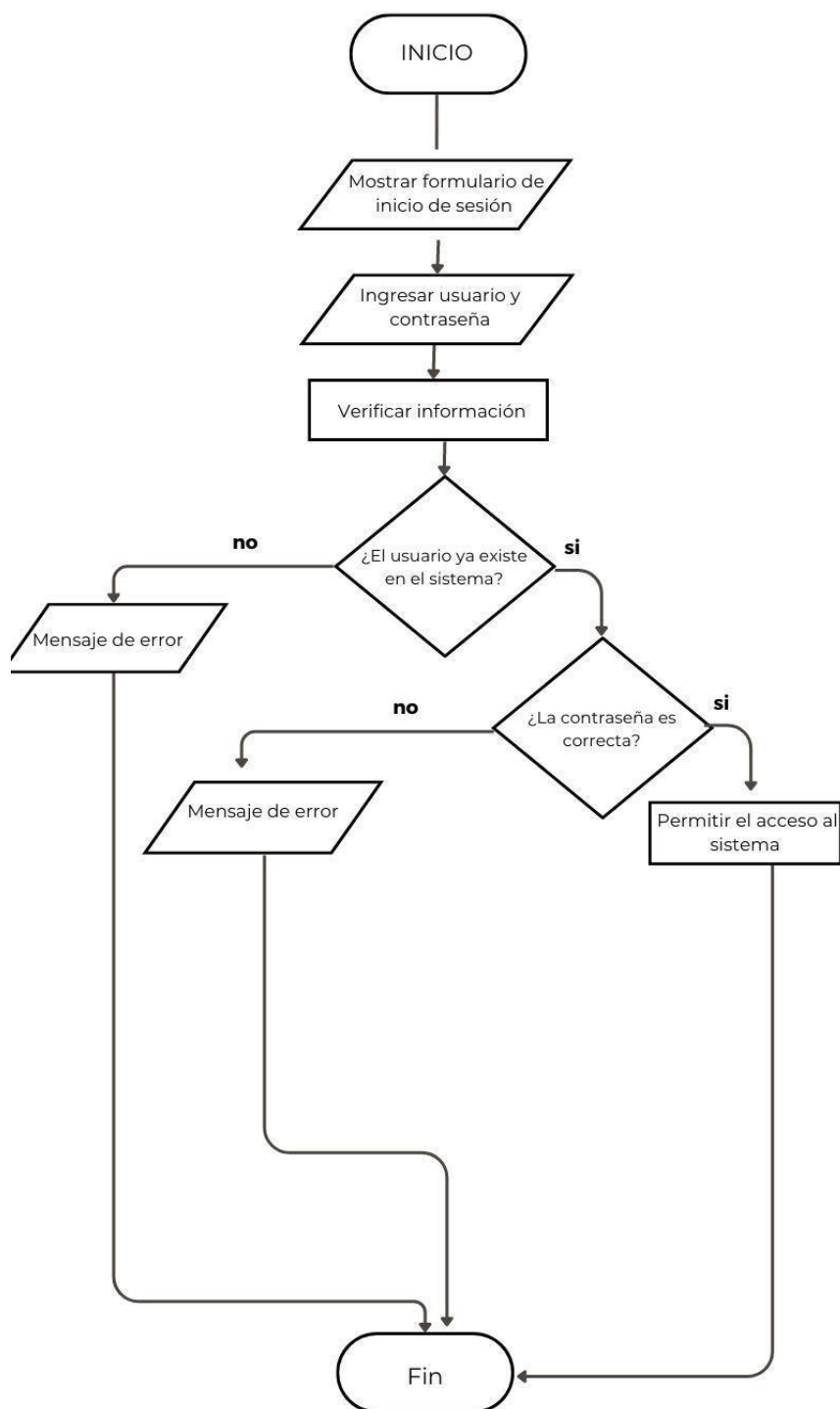


Figura 156 Diagrama de flujo del proceso Iniciar sesión.

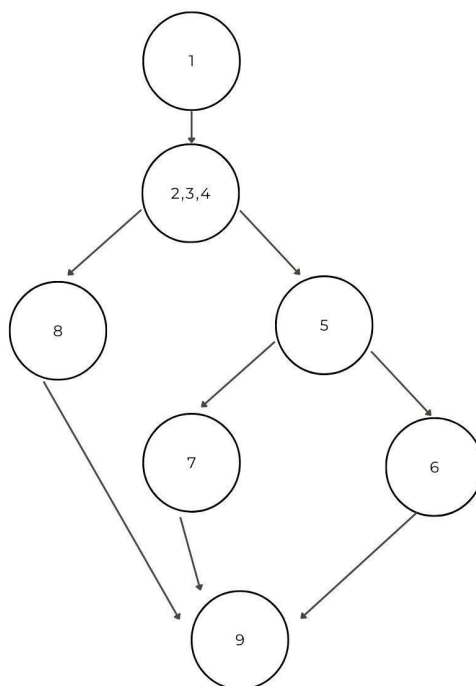


Figura 157 Grafo de flujo del proceso Iniciar sesión

Tabla 40 Complejidad ciclomática del proceso Iniciar sesión.

Datos	Formula	Sustitución	Resultados
A = 8 N = 7	$V(G) = A - N + 2$	$V(G) = 8 - 7 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 41 Caminos básicos del proceso Iniciar sesión.

Camino	Ruta
C1	1,2,3,4,5,6,9
C2	1,2,3,4,5,7,9
C3	1,2,3,4,8,9

Tabla 42 Casos de prueba del proceso Iniciar sesión.

INICIAR SESIÓN					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,9	¿El usuario ya existe en el sistema? ¿La contraseña es correcta?	Si Si	El usuario accede al sistema web.	El usuario accede al sistema web.
2	1,2,3,4,5,7,9	¿El usuario ya existe en el sistema? ¿La contraseña es correcta?	Si No	El sistema muestra mensaje de error.	El sistema no permite al usuario acceder al sistema web.
3	1,2,3,4,8,9	¿El usuario ya existe en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Registrar Información de Alumnos y Docentes.

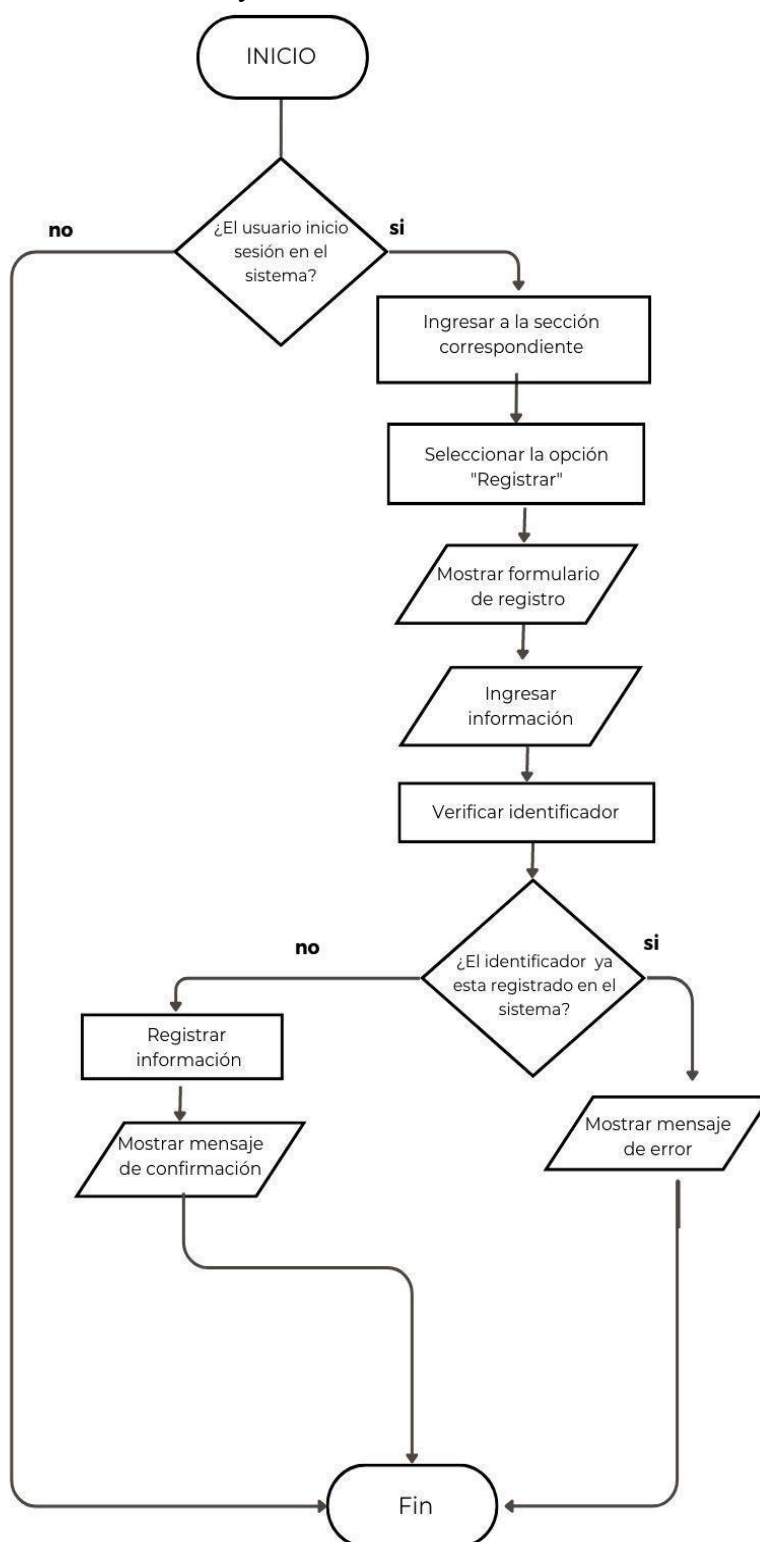


Figura 158 Diagrama de flujo del proceso Registrar.

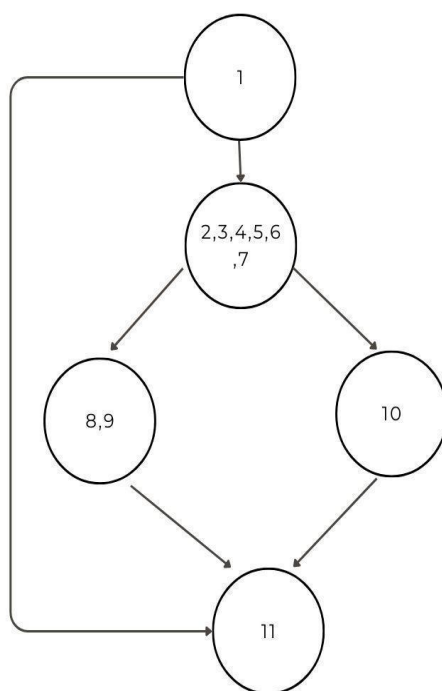


Figura 159 Grafo de flujo del proceso Registrar.

Tabla 43 Complejidad ciclomática del proceso Registrar.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 44 Caminos básicos del proceso Registrar.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,11
C2	1,2,3,4,5,6,7,10,11
C3	1,11

Tabla 45 Casos de prueba del proceso Registrar.

REGISTRAR INFORMACIÓN					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si No	El sistema muestra mensaje de confirmación.	El sistema ejecuta el registro de información y muestra mensaje de confirmación.
2	1,2,3,4,5,6,7,10,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si Si	El sistema muestra mensaje de error.	El sistema no ejecuta el registro de información y muestra mensaje de error.
3	1,11	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Modificar Información de Alumnos, Docentes y Coloquios.

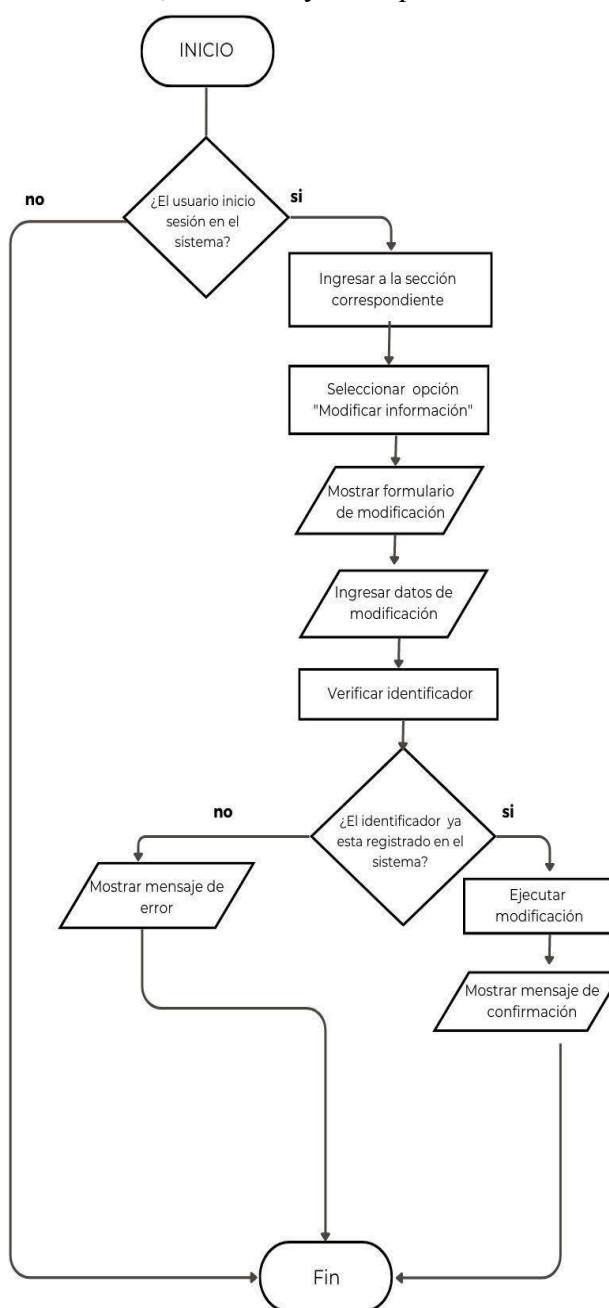


Figura 160 Diagrama de flujo del proceso Modificar.

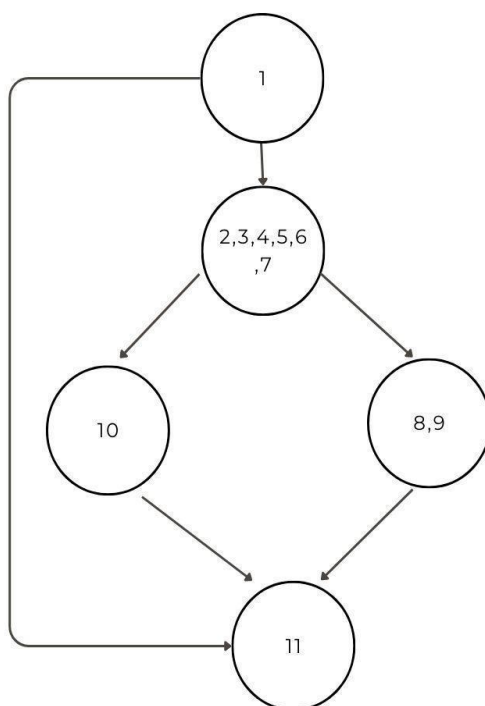


Figura 161 Grafo de flujo del proceso Modificar.

Tabla 46 Complejidad ciclomática del proceso Modificar.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 47 Caminos básicos del proceso Modificar.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,11
C2	1,2,3,4,5,6,7,10,11
C3	1,11

Tabla 48 Casos de prueba del proceso Modificar.

MODIFICAR INFORMACIÓN					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si Si	El sistema muestra mensaje de confirmación.	El sistema ejecuta la modificación de información y muestra mensaje de confirmación.
2	1,2,3,4,5,6,7,10,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si No	El sistema muestra mensaje de error.	El sistema no ejecuta la modificación de información y muestra mensaje de error.
3	1,11	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Eliminar Alumnos o Docentes

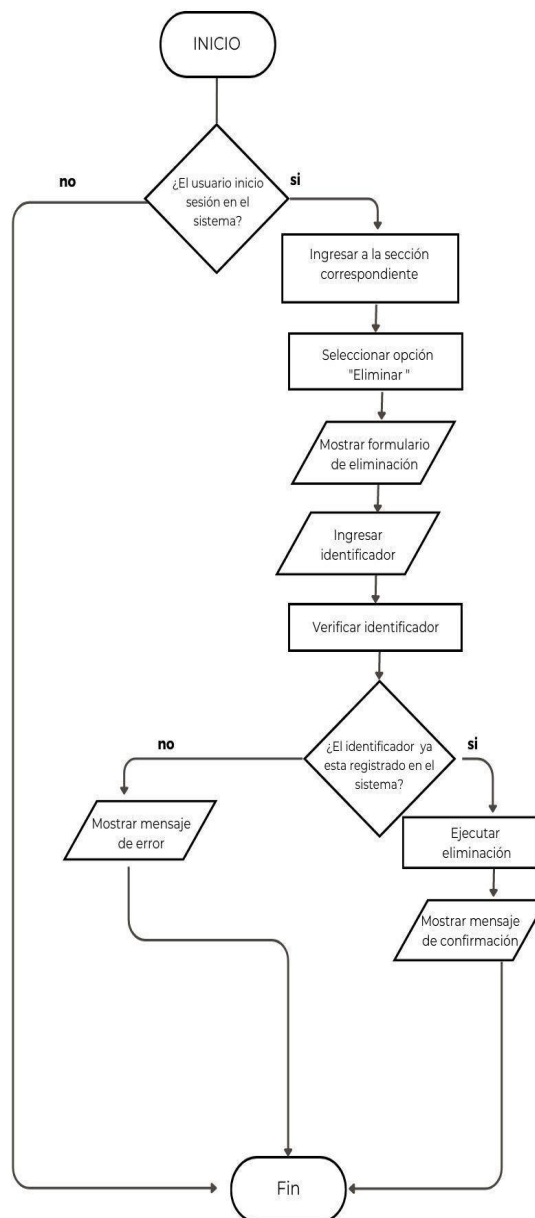


Figura 162 Diagrama de flujo del proceso Eliminar.

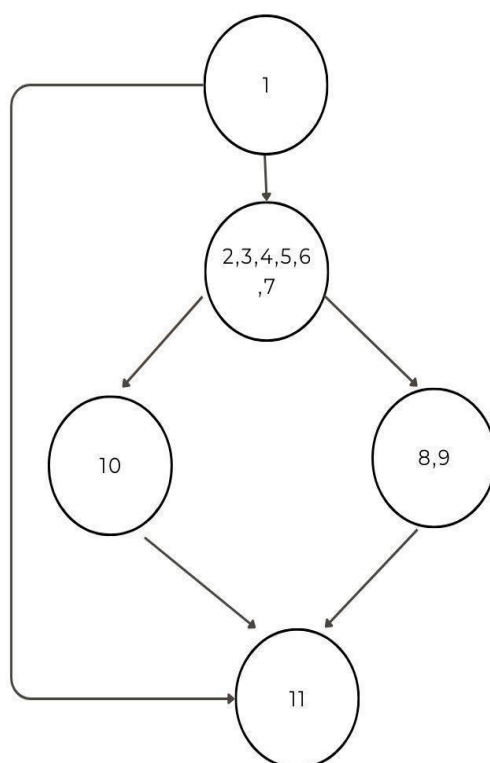


Figura 163 Grafo de flujo del proceso Eliminar.

Tabla 49 Complejidad ciclomática del proceso Eliminar.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 50 Caminos básicos del proceso Eliminar.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,11
C2	1,2,3,4,5,6,7,10,11
C3	1,11

Tabla 51 Casos de prueba del proceso Eliminar.

ELIMINAR INFORMACIÓN					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si Si	El sistema muestra mensaje de confirmación.	El sistema ejecuta la eliminación de información y muestra mensaje de confirmación.
2	1,2,3,4,5,6,7,10,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si No	El sistema muestra mensaje de error.	El sistema no ejecuta la eliminación de información y muestra mensaje de error.
3	1,11	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Crear expediente de alumno.

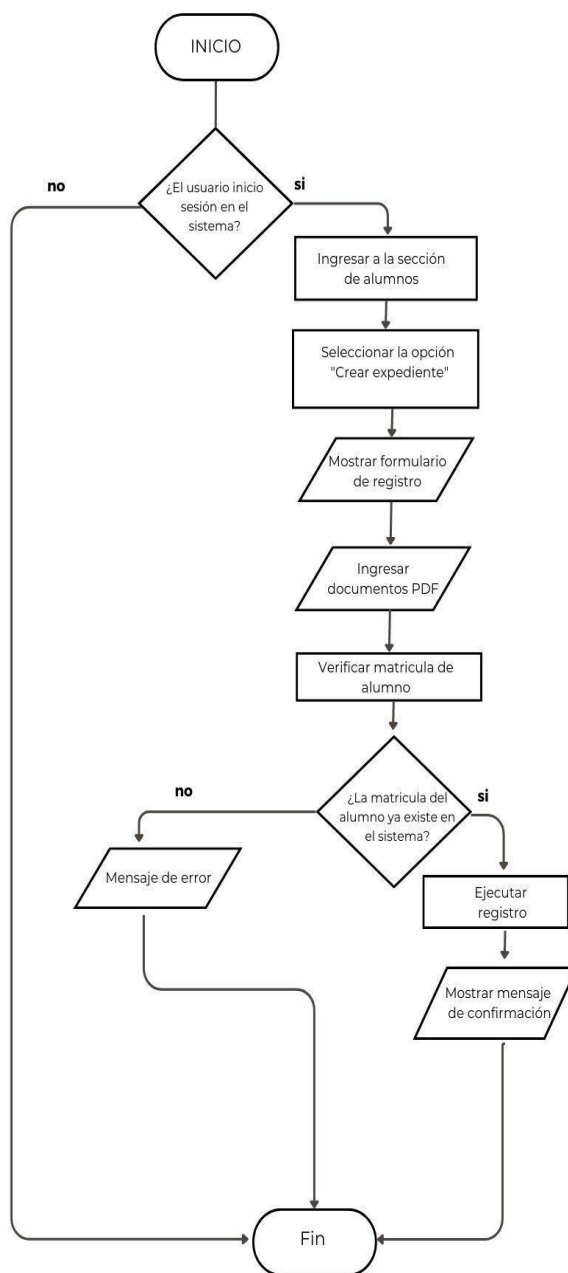


Figura 164 Diagrama de flujo del proceso de Crear expediente de alumno.

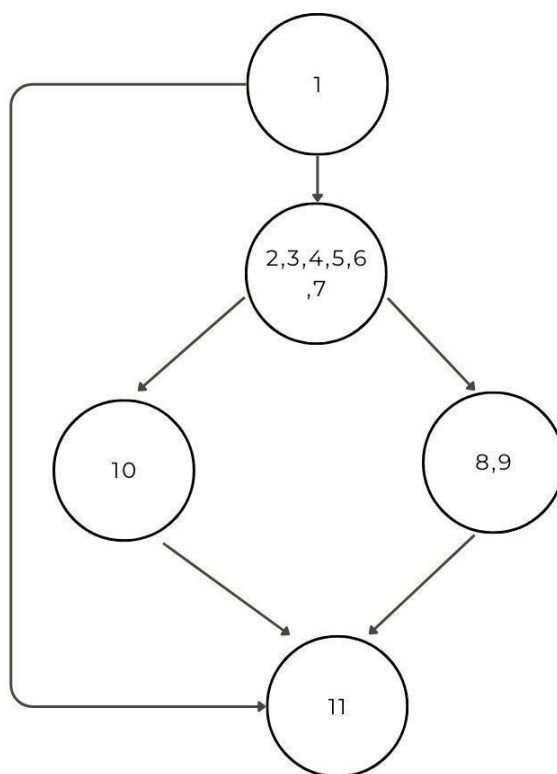


Figura 165 Grafo de flujo del proceso Crear expediente de alumno.

Tabla 52 Complejidad ciclomática del proceso Crear expediente de alumno.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP=2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 53 Caminos básicos del proceso Crear expediente de alumno.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,11
C2	1,2,3,4,5,6,7,10,11
C3	1,11

Tabla 54 Casos de prueba del proceso Crear expediente de alumno.

CREAR EXPEDIENTE DE ALUMNO					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si Si	El sistema muestra mensaje de confirmación.	El sistema ejecuta el registro de los documentos PDF y muestra mensaje de confirmación.
2	1,2,3,4,5,6,7,10,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si No	El sistema muestra mensaje de error.	El sistema no ejecuta el registro de los documentos PDF y muestra mensaje de error.
3	1,11	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Agendar coloquios

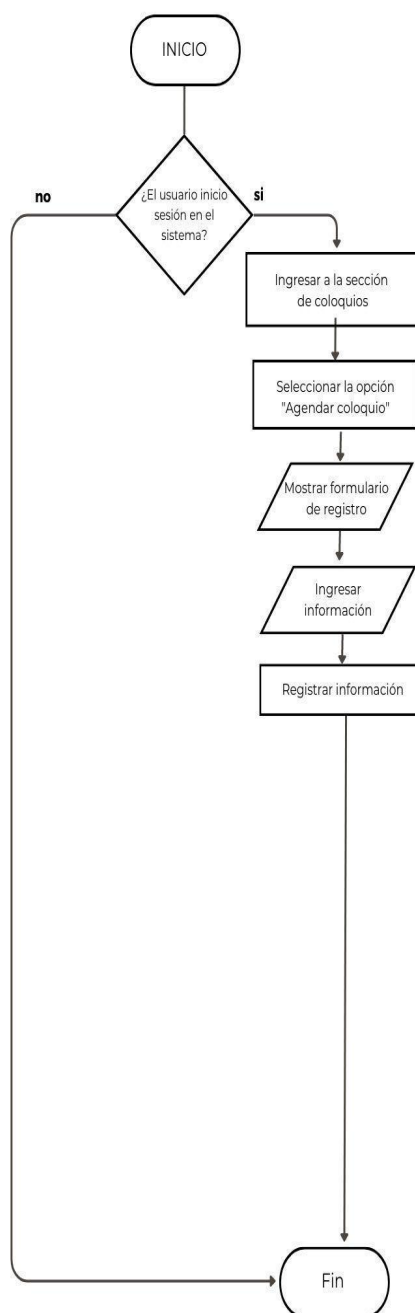


Figura 166 Diagrama de flujo del proceso de Agendar coloquios.

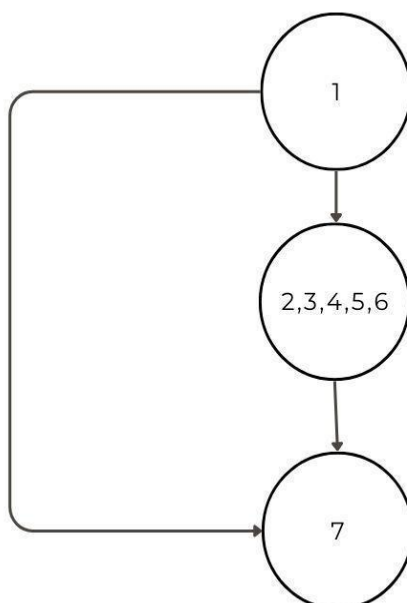


Figura 167 Grafo de flujo del proceso Agendar coloquios.

Tabla 55 Complejidad ciclomática del proceso Agendar coloquios.

Datos	Formula	Sustitución	Resultados
A = 3 N = 3	$V(G) = A - N + 2$	$V(G) = 3 - 3 + 2$	$V(G) = 2$
NP=1	$V(G) = NP + 1$	$V(G) = 1 + 1$	$V(G) = 2$

Tabla 56 Caminos básicos del proceso Agendar coloquios.

Camino	Ruta
C1	1,2,3,4,5,6,7
C3	1,7

Tabla 57 Casos de prueba del proceso Agendar coloquios.

AGENDAR COLOQUIOS					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7	¿El usuario inicio sesión en el sistema?	Si	El sistema permite agendar coloquios al usuario	El sistema permite agendar coloquios al usuario.
3	1,7	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Asignación de cursos a alumnos y docentes

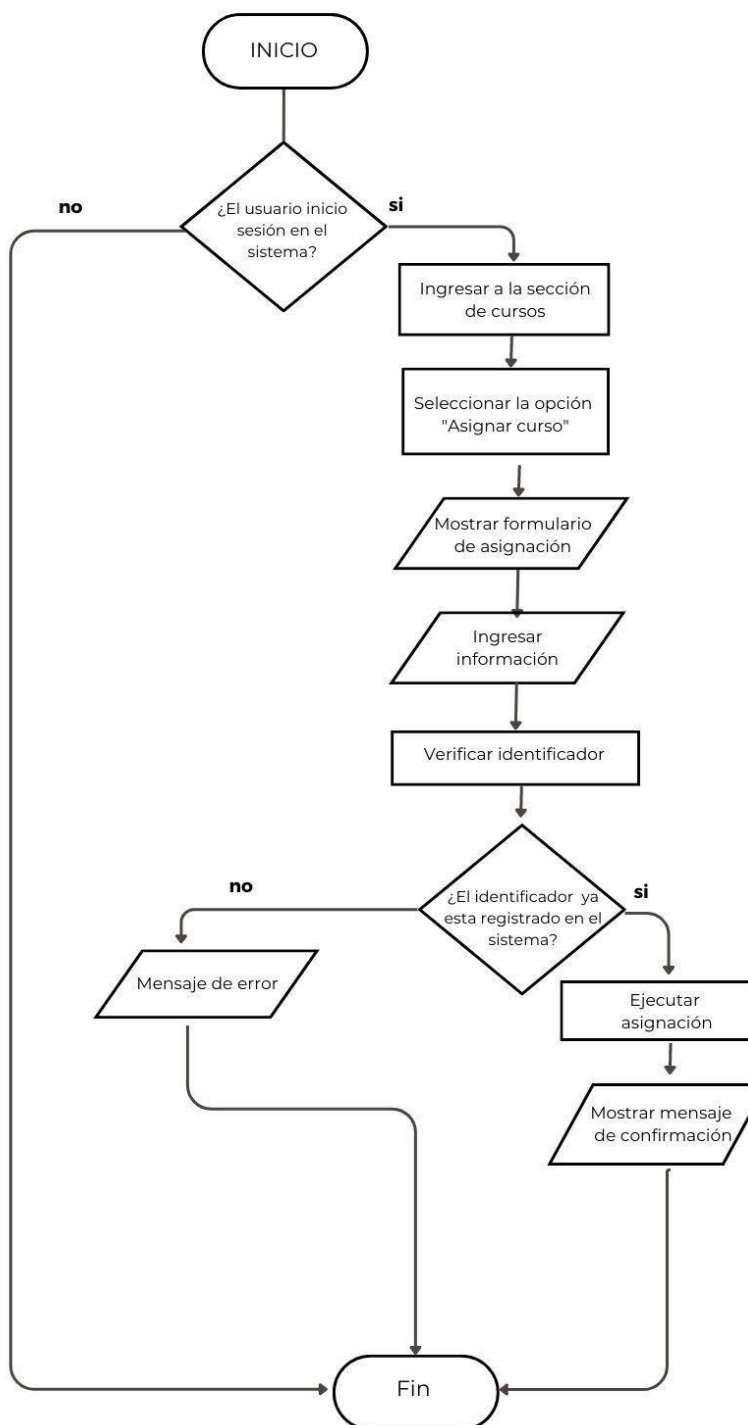


Figura 168 Diagrama de flujo del proceso Asignación de cursos.

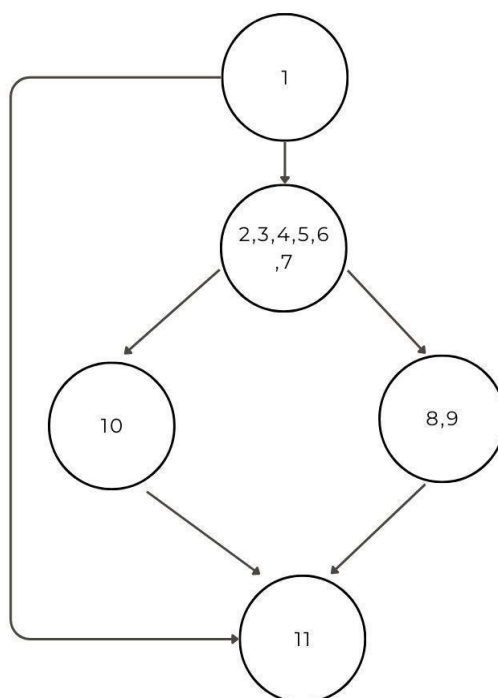


Figura 169 Grafo de flujo del proceso Asignación de cursos.

Tabla 58 Complejidad ciclomática del proceso Asignación de cursos.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 59 Caminos básicos del proceso Asignación de cursos.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,11
C2	1,2,3,4,5,6,7,10,11
C3	1,11

Tabla 60 Casos de prueba del proceso Asignación de cursos.

ASIGNACIÓN DE CURSOS					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si Si	El sistema muestra mensaje de confirmación.	El sistema ejecuta la asignación y muestra mensaje de confirmación.
2	1,2,3,4,5,6,7,10,11	¿El usuario inicio sesión en el sistema? ¿El identificador ya esta registrado en el sistema?	Si No	El sistema muestra mensaje de error.	El sistema no ejecuta la asignación y muestra mensaje de error.
3	1,11	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Consultar información de alumnos, docentes, coloquios y cursos.

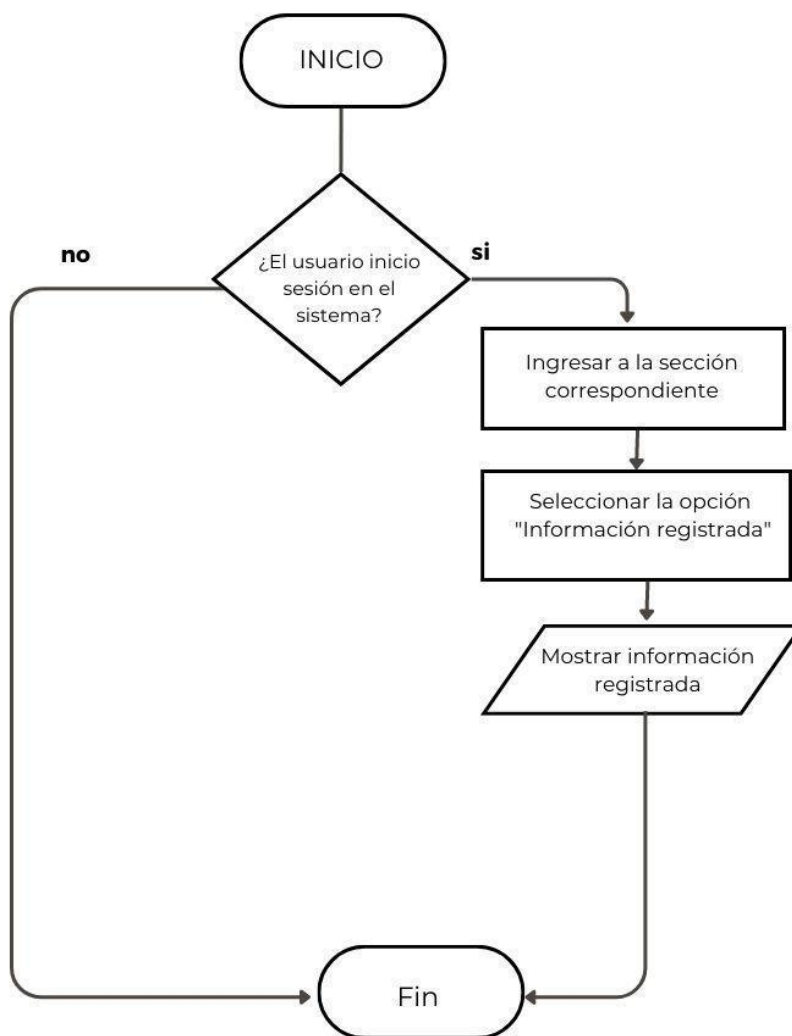


Figura 170 Diagrama de flujo del proceso de Consultar información.

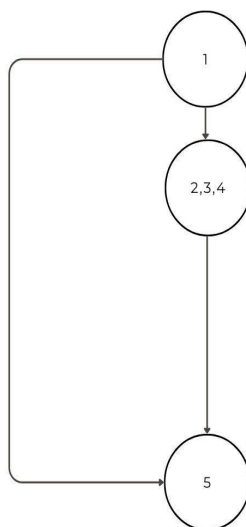


Figura 171 Grafo de flujo del proceso Consultar información.

Tabla 61 Complejidad ciclomática del proceso Consultar información.

Datos	Formula	Sustitución	Resultados
A = 3 N = 3	$V(G) = A - N + 2$	$V(G) = 3 - 3 + 2$	$V(G) = 2$
NP = 1	$V(G) = NP + 1$	$V(G) = 1 + 1$	$V(G) = 2$

Tabla 62 Caminos básicos del proceso Consultar información.

Camino	Ruta
C1	1,2,3,4,5
C2	1,5

Tabla 63 Casos de prueba del proceso consultar información.

CONSULTAR INFORMACIÓN					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5	¿El usuario inicio sesión en el sistema?	Si	El sistema muestra tabla de información.	El sistema muestra tabla de información.
2	1,5	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Imprimir información de alumnos y docentes.

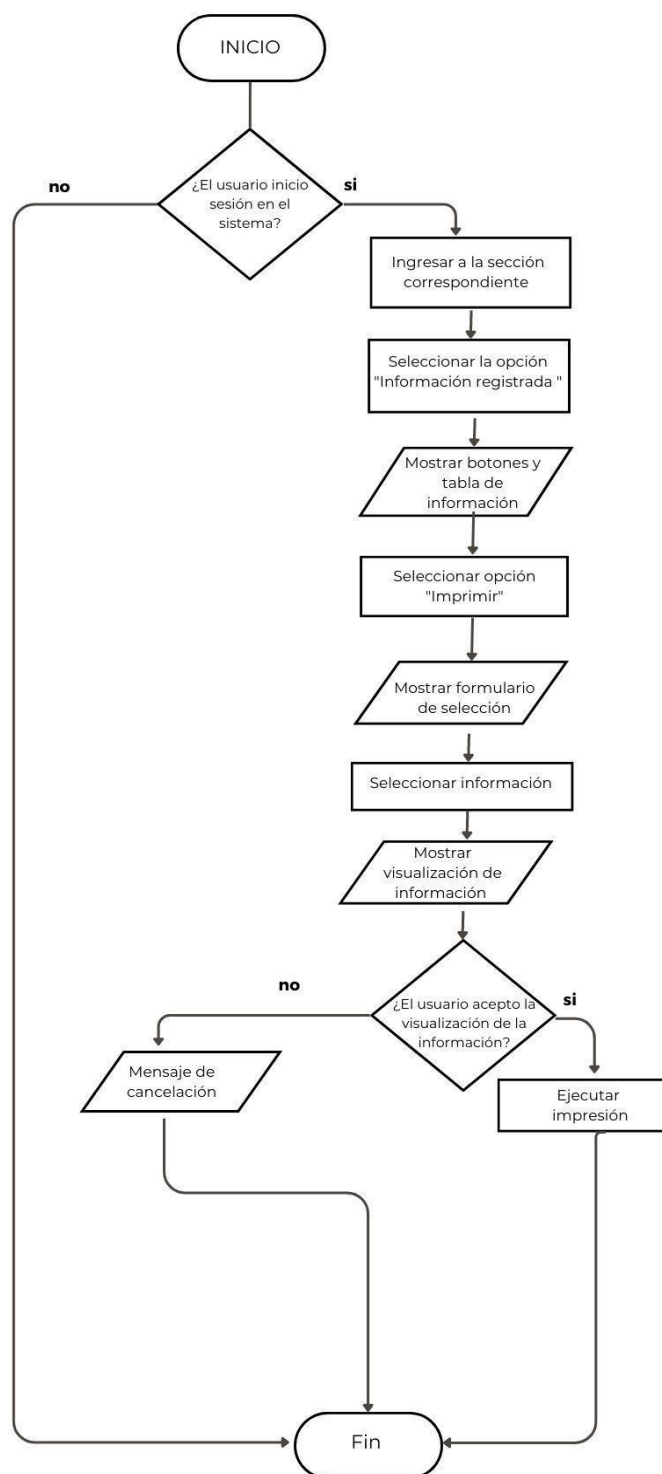


Figura 172 Diagrama de flujo del proceso de Imprimir y seleccionar información.

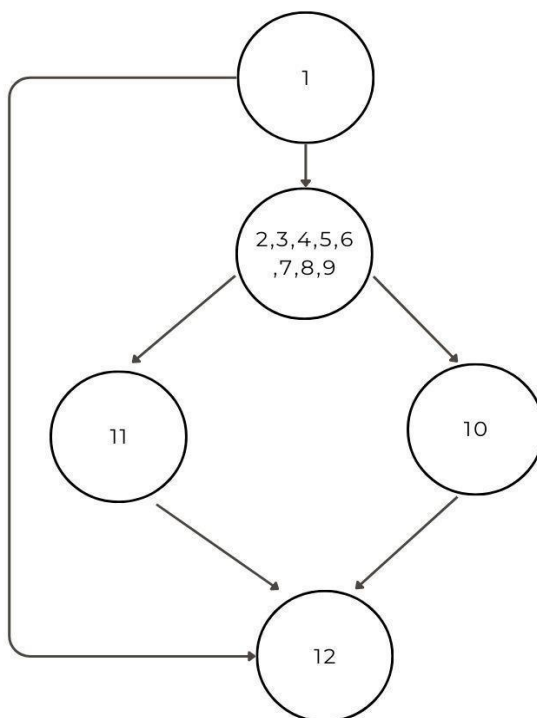


Figura 173 Grafo de flujo del proceso de Imprimir y seleccionar información.

Tabla 64 Complejidad ciclomática del proceso Imprimir y seleccionar información.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 65 Caminos básicos del proceso Imprimir y seleccionar información.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,10,12
C2	1,2,3,4,5,6,7,8,9,11,12
C3	1,12

Tabla 66 Casos de prueba del proceso Imprimir y seleccionar información.

IMPRIMIR INFORMACIÓN DE ALUMNOS O DOCENTES					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,10,12	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la visualización de la información?	Si Si	El sistema imprime la información seleccionada.	El sistema imprime la información seleccionada.
2	1,2,3,4,5,6,7,8,9,11,12	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la visualización de la información?	Si No	El sistema muestra mensaje de cancelación.	El sistema no imprime la información seleccionada.
3	1,12	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Exportar información de alumnos y docentes

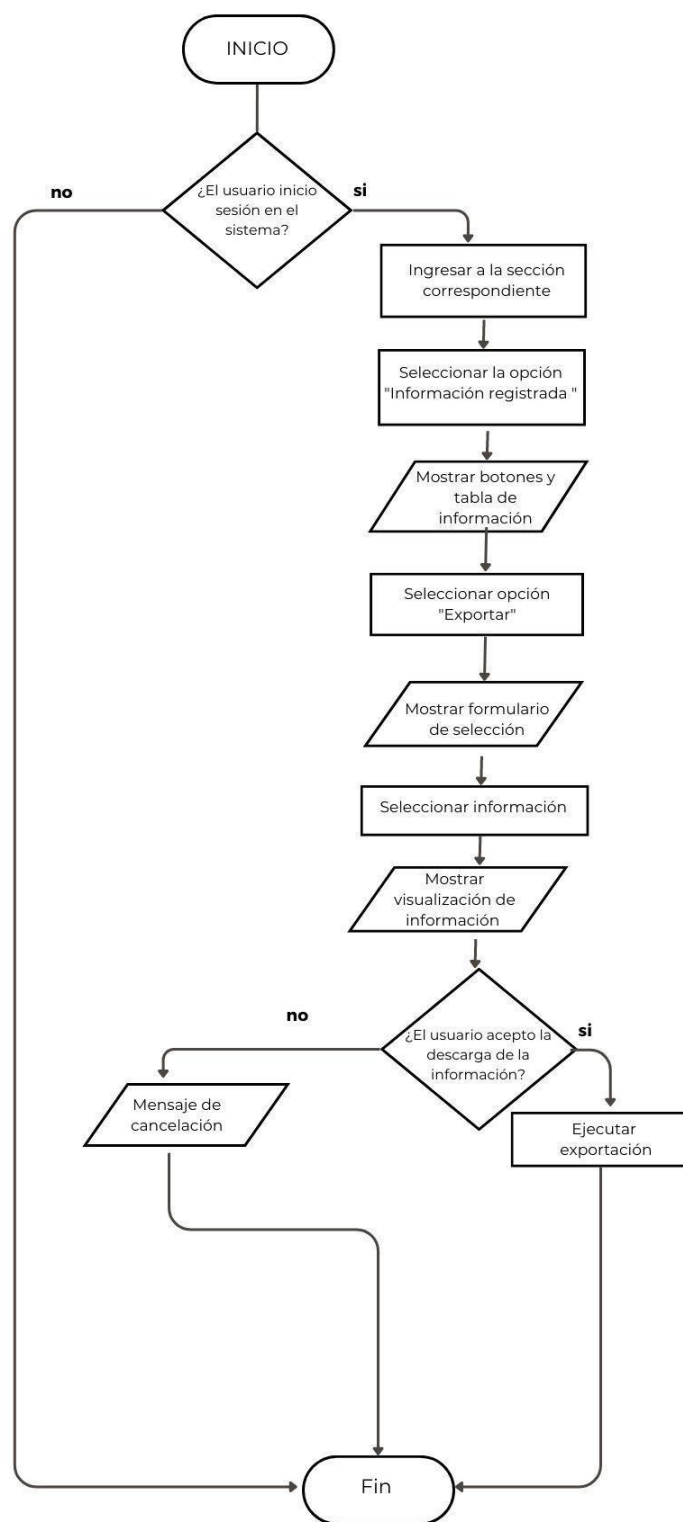


Figura 174 Diagrama de flujo del proceso de Exportar y seleccionar información.

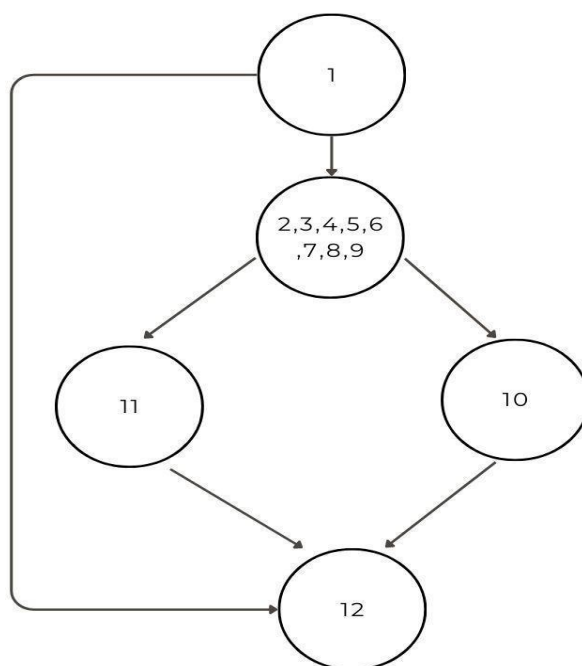


Figura 175 Grafo de flujo del proceso de Exportar y seleccionar información.

Tabla 67 Complejidad ciclomática del proceso Exportar y seleccionar información.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP = 2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 68 Caminos básicos del proceso Exportar y seleccionar información.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,9,10,12
C2	1,2,3,4,5,6,7,8,9,11,12
C3	1,12

Tabla 69 Casos de prueba del proceso Exportar y seleccionar información.

EXPORTAR INFORMACIÓN DE ALUMNOS O DOCENTES					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,9,10,12	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la descarga de la información?	Si Si	El sistema exporta la información seleccionada.	El sistema exporta la información seleccionada.
2	1,2,3,4,5,6,7,8,9,11,12	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la descarga de la información?	Si No	El sistema muestra mensaje de cancelación.	El sistema no exporta la información seleccionada.
3	1,12	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Imprimir información de coloquios, cursos y asignaciones.

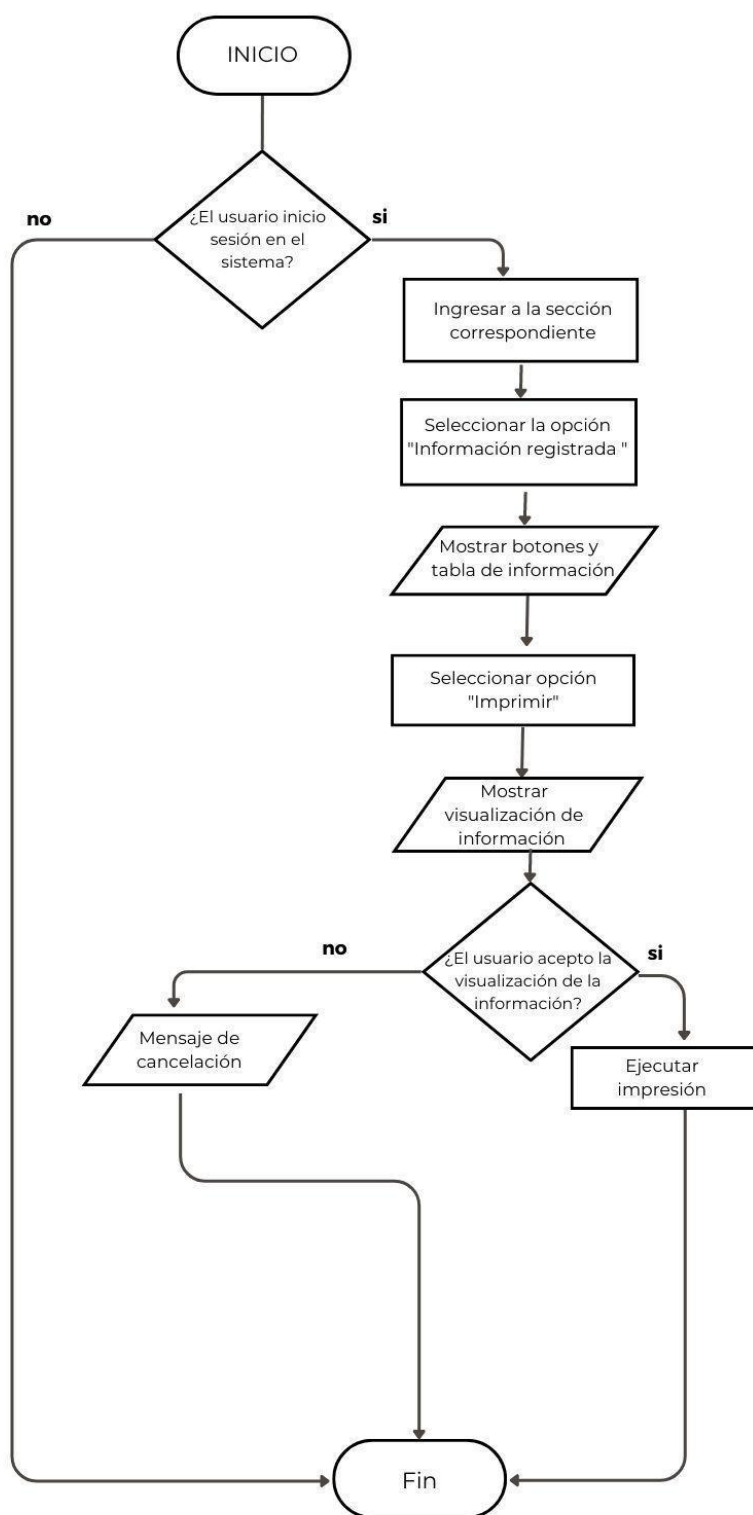


Figura 176 Diagrama de flujo del proceso Imprimir información.

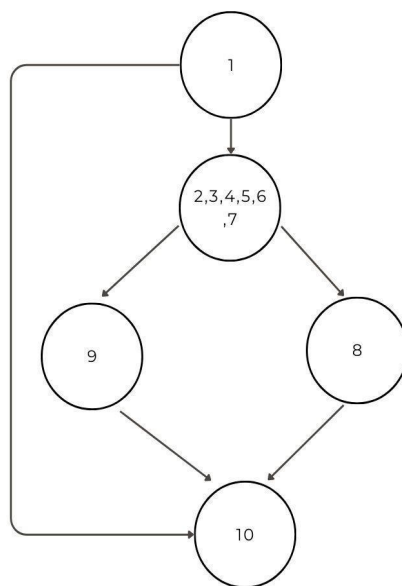


Figura 177 Grafo de flujo del proceso Imprimir información.

Tabla 70 Complejidad ciclomática del proceso Imprimir información.

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP=2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 71 Caminos básicos del proceso Imprimir información.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,10
C2	1,2,3,4,5,6,7,9,10
C3	1,10

Tabla 72 Casos de prueba del proceso Imprimir información.

IMPRIMIR INFORMACIÓN DE COLOQUIOS O CURSOS					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,10	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la visualización de la información?	Si Si	El sistema imprime toda la información de la sección	El sistema imprime toda la información de la sección correspondiente.
2	1,2,3,4,5,6,7,9,10	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la visualización de la información?	Si No	El sistema muestra mensaje de cancelación.	El sistema no ejecuta la exportación de la información.
3	1,1	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Exportar información de coloquios, cursos y asignaciones.

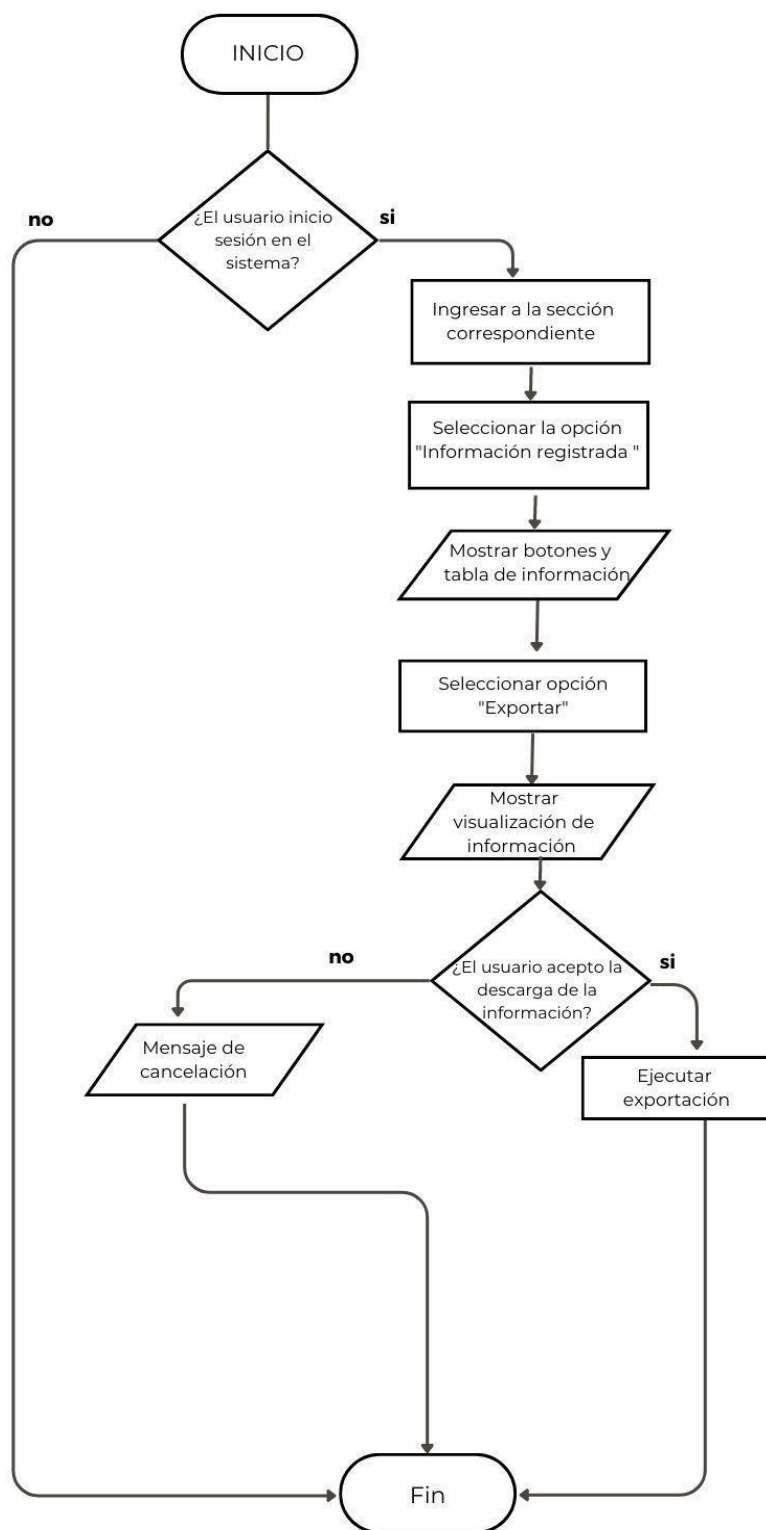


Figura 178 Diagrama de flujo del proceso Exportar información.

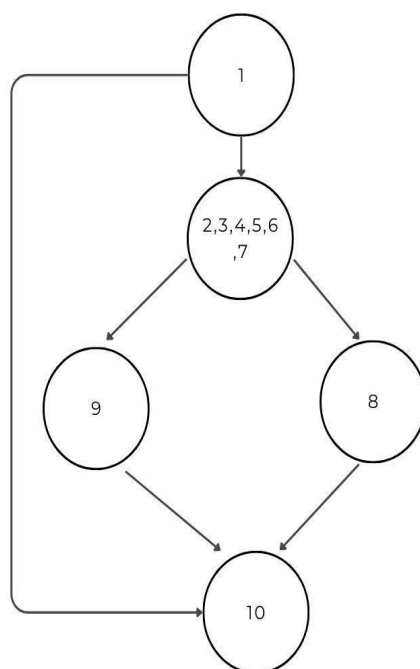


Figura 179 Grafo de flujo del proceso Exportar información.

Tabla 73 Complejidad ciclomática del proceso Exportar información

Datos	Formula	Sustitución	Resultados
A = 6 N = 5	$V(G) = A - N + 2$	$V(G) = 6 - 5 + 2$	$V(G) = 3$
NP=2	$V(G) = NP + 1$	$V(G) = 2 + 1$	$V(G) = 3$

Tabla 74 Caminos básicos del proceso Exportar información.

Camino	Ruta
C1	1,2,3,4,5,6,7,8,10
C2	1,2,3,4,5,6,7,9,10
C3	1,10

Tabla 75 Casos de prueba del proceso Exportar información.

EXPORTAR INFORMACIÓN DE COLOQUIOS O CURSOS					
ID	CAMINO	ENTRADA	DATOS DE PRUEBA	RESULTADO ESPERADO	RESULTADO REAL
1	1,2,3,4,5,6,7,8,10	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la visualización de la información?	Si Si	El sistema exporta toda la información de la sección correspondiente.	El sistema exporta toda la información de la sección correspondiente.
2	1,2,3,4,5,6,7,9,10	¿El usuario inicio sesión en el sistema? ¿El usuario acepto la visualización de la información?	Si No	El sistema muestra mensaje de cancelación.	El sistema no ejecuta la exportación de la información.
3	1, 10	¿El usuario inicio sesión en el sistema?	No	El usuario no tiene acceso al sistema web.	El usuario no tiene acceso al sistema web.

Conclusión

Como se mencionó anteriormente, la gestión de programas educativos es de importancia para quien funge el rol de coordinador, así como para los docentes y alumnos que forman parte de cada programa educativo.

En este trabajo de tesis se diseña y desarrolla un sistema web que permite mejorar la manera de gestionar la información de los alumnos de posgrado de la facultad de Ciencias de la Computación perteneciente a la Benemérita Universidad Autónoma de Puebla.

La plataforma está conformada por las secciones de Alumnos, Docentes, Coloquios y Cursos. Las operaciones que permite realizar son inserción, eliminación, modificación y consulta de información, así como la asignación de cursos a estudiantes y catedráticos. Asimismo permite imprimir y exportar datos de coloquios, asignaciones y cursos registrados en el sistema, también posibilita seleccionar información de alumnos y docentes para su posterior exportación o impresión.

El sistema descrito anteriormente simplifica las diferentes operaciones que componen el proceso de gestión escolar, reduciendo tiempos de ejecución y la cantidad de esfuerzo que el usuario final dedica normalmente a la mencionada acción.

De igual forma le permite al usuario coordinador tener accesibilidad a la información desde cualquier lugar y a cualquier hora, favorece la disminución de errores humanos, así como su productividad laboral.

Para trabajos futuros se recomienda desarrollar el método referente a la recuperación de contraseña, así como reforzar la seguridad del sistema web y realizar el alojamiento de la plataforma en un web hosting.

Se sugiere detallar la sección de información, ya que requiere una mejor descripción del funcionamiento general del sistema, y de la forma de uso del mismo. También se puntualiza en perfeccionar los formularios respectivos a la captura de información de alumnos y docentes, e implementar la funcionalidad que permita seleccionar en qué formato se quiere exportar la información contenida en el sistema. Por último se incita a realizar pruebas de usabilidad para mejorar la experiencia de usuario.

En conclusión, el presente proyecto es una herramienta que optimiza tiempo y esfuerzo con el propósito de facilitar el trabajo que realiza el coordinador de posgrados de la FCC. La aplicación implementada permite mantener la integridad de los datos, es escalable y cuenta con una interfaz amigable con el usuario, haciéndola apta para manejar grandes cantidades de datos de forma simple y eficiente.

Referencias

Anderson, D.J. (s.f.). Revisando los principios y prácticas generales del método Kanban. Fecha de consulta: 28 de agosto de 2023. <https://djaa.com/revisando-los-principios-y-practicas-generales-del-metodo-kanban/>

Apache Friends. (s. f.). XAMPP Installers and Downloads for Apache Friends. <https://www.apachefriends.org/es/index.html>

Apolaya Ñaupá, L. E. (2018). Aplicación web para la mejora de la gestión académica del Instituto de Educación Superior Tecnológico Público “Chincha”. [Tesis de Licenciatura Universidad Inca Garcilaso de la Vega. Lima, Perú]. <http://repositorio.uigv.edu.pe/bitstream/handle/20.500.11818/3443/TESIS-LUIS%20APOLAYA%20%C3%91AUPA.PDF?sequence=2&isAllowed=y>

Armao O., Carbajo J., Espitia N. (2016). MODELO VISTA-CONTROLADOR (MVC). Miranda. República Bolivariana de Venezuela. Universidad Alejandro de Humboldt. <https://espejodeantagno.files.wordpress.com/2016/04/modelo-vista-controlador-mvc.pdf>

Asale, R.-. (s. f.). gestión | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/gesti%C3%B3n?m=form>

Asale, R.-. (s. f.). gestionar | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/gestionar%20?m=form>

Asale, R.-. (s. f.). interfaz | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/interfaz%20?m=form>

Asale, R.-. (s. f.). lenguaje | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/lenguaje?m=form>

Asale, R.-. (s. f.). metodología | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/metodolog%C3%ADa?m=form>

Asale, R.-. (s. f.). modelo | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/modelo?m=form>

Asale, R.-. (s. f.). sistema | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/sistema%20?m=form>

Asale, R.-. (s. f.). sitio | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/sitio?m=form>

Asale, R.-. (s. f.-i). técnico, técnica | Diccionario de la lengua española. Edición del Tricentenario. <https://dle.rae.es/t%C3%A9cnico>

Azure. (s.f). ¿Qué es una base de datos relacional? <https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-a-relational-database/#whatis>

Barrientos Rodríguez, M., Castro Peraza, M.A., Zacca González, G. y Álvarez Gainza, D. (2022). Sistema Web para la gestión de los programas de maestrías del Instituto “Pedro Kouri”. Revista Cubana de Informática Médica 2022:14(1):e468

Booch, G., Rumbaugh, J., Jacobson, I. (1999). El Lenguaje Unificado de Modelado. Addison Wesley.

Castellano Lendínez, L. (2019). Kanban. Metodología para aumentar la eficiencia de los procesos. 3C Tecnología. Glosas de innovación aplicadas a la pyme, 8(1), pp. 30-41. doi: <http://dx.doi.org/10.17993/3ctecno/2019.v8n1e29/30-41>

Contributors, P. (s. f.). phpMyAdmin. phpMyAdmin. <https://www.phpmyadmin.net/>

Fowler, M. y Scott, K. (1999). UML gota a gota. Pearson Education.

Garcés, G. (2022). Ventajas de usar Figma como herramienta de diseño UI - Blog de Hiberus Tecnología. Blog de Hiberus Tecnología. <https://www.hiberus.com/crecemos-contigo/ventajas-de-usar-figma-como-herramienta-de-diseno-ui/>

García Peñalvo, F.J. y García Holgado, A. (2018). FUNDAMENTOS DE LA VISTA DE CASOS DE USO. INGENIERÍA DE SOFTWARE I. Salamanca, España: Grupo GRIAL, Universidad de Salamanca. <https://repositorio.grial.eu/bitstream/grial/1155/1/UML%20-%20Casos%20de%20uso.pdf>

García Peñalvo, F.J. y Vázquez Ingelmo, A. (2019). FUNDAMENTOS DE LA VISTA DE INTERACCIÓN. INGENIERÍA DE SOFTWARE I. Salamanca, España: Grupo GRIAL, Universidad de Salamanca. <https://repositorio.grial.eu/bitstream/grial/1566/1/UML%20-%20Vista%20de%20interaccion.pdf>

Gómez, C. (2023, enero 19). Técnicas de prueba de caja negra. Diario de QA. <https://www.diariodeqa.com/post/tecnicas-de-prueba-de-caja-negra>

Gutiérrez Díaz A. (s. f.). BASES DE DATOS. Ciudad de México, México. Centro Cultural Itaca S.C. <https://docplayer.es/1667114-Bases-de-datos-mis-308-bases-de-datos-clave-mis-308-profesor-m-c-alejandro-gutierrez-diaz.html>

IEEE Std 610.12-1990 (R2002). (1990). IEEE Standard Glossary of Software Engineering Terminology, IEEE.

Kotonya, G. & Sommerville, I. (2000). Requeriments Engineering: Processes and Techniques. John Wiley & Sons.

Maida, EG, Pacienza, J. (2015). Metodologías de desarrollo de software [en línea]. Tesis de Licenciatura en Sistemas y Computación. Facultad de Química e Ingeniería “Fray Rogelio Bacon”. Universidad Católica Argentina.
<https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>

Mantilla Oliva J.A. (2019). Propuesta de un sistema complementario de gestión académica del docente y alumno universitario. Perú. Universidad de Piura.
https://pirhua.udep.edu.pe/bitstream/handle/11042/4267/ING_633.pdf?sequence=1&isAllowed=y

Maquieira, G. R. (2023). Qué es black box testing o pruebas de caja negra. OpenWebinars.net
<https://openwebinars.net/blog/que-es-black-box-testing-o-pruebas-de-caja-negra/>

Marqués, M. (2011). Bases de Datos. Castellón de la Plana, Castelló, España. Universidad Jaime I. <https://bdigital.uvhm.edu.mx/wp-content/uploads/2020/05/Bases-de-Datos.pdf>

MDN. (s.f. a). Conceptos básicos de HTML - Aprende desarrollo web.
https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics

MDN. (s. f. b). CSS básico.
https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/CSS_basics

MDN. (s. f. c). Fundamentos de JavaScript.
https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics

Mendoza Rivilla, J. E, (2017). Implementación de un sistema web para la gestión y control de los procesos de la unidad de titulación de la carrera de Ingeniería en Sistemas de la Universidad Salesiana, Sede Guayaquil. [Tesis de Licenciatura Universidad Politécnica Salesiana. Guayaquil, Ecuador].
<https://dspace.ups.edu.ec/bitstream/123456789/14482/4/UPS-GT001930.pdf>

Núñez F. J.(2018).Tipos de Requerimientos Ingeniería de Software. Escuela Superior de Huejutla Ciencias Computacionales.
https://www.uaeh.edu.mx/docencia/P_Presentaciones/huejutla/sistemas/2018/tipos-de-requerimientos-FelipeDeJNu%C3%B1ezCardenas.pdf

Oppel A.,Sheldon R. (2010). Fundamentos de SQL (3ra ed.).McGRAW-HILL.
https://pedrobeltrancanessa-biblioteca.weebly.com/uploads/1/2/4/0/12405072/fundamentos_de_sql_3edi_oppel.pdf

Pexels. (s. f.). Quiénes somos - Pexels. <https://www.pexels.com/es-es/about/>

PHP. (s. f.). ¿Qué es PHP? - Manual. <https://www.php.net/manual/es/intro-what-is.php>

PHP: PDO. (s. f.). Manual. <https://www.php.net/manual/es/book.pdo.php>

Pressman, R. S. (2010). Ingeniería del software. Un enfoque práctico. Séptima Edición. McGraw Hill. ISBN: 978-607-15-0314-5. México.

Pressman, R. S. (2010). Ingeniería del software. Un enfoque práctico. Séptima Edición. McGraw Hill. ISBN: 978-607-15-0314-5. México.

Quenguan Acosta, J.C. (2011). Informe: Mapa de navegación.
<https://bagatelaproductiva.files.wordpress.com/2011/03/informemapasdenavegacion.pdf>

Robledano, A. (2023, 13 abril). Qué es MySQL: Características y ventajas.
OpenWebinars.net. <https://openwebinars.net/blog/que-es-mysql/>

Rocha C. R. (2017). EL MODELO ENTIDAD.RELACIÓN.
<https://www.medellin.unal.edu.co/~fjmoreno/bd1/ModeloERRochav8.pdf>

Sánchez Peño J. M.(2015). Pruebas de Software.Fundamentos y Técnicas. Madrid. España.
Universidad Politécnica de Madrid.
https://oa.upm.es/40012/1/PFC_JOSE_MANUEL_SANCHEZ_PENO_3.pdf

Singureanu, C. (2023). Pruebas de Caja Blanca: ¡Qué es, ¡Cómo funciona, Retos, Métricas, Herramientas y Más! ZAPTEST.
<https://www.zaptest.com/es/pruebas-de-caja-blanca-que-es-como-funciona-retos-metricas-herramientas-y-mas>

Urrutia, D. (2021). Qué es Bootstrap - Definición, significado y ejemplos. Arimetrics.
<https://www.arimetrics.com/glosario-digital/bootstrap>

Visual Studio Code. (s.f). Recuperado el 30 de agosto de 2023. <https://code.visualstudio.com/>

Zurita Lara B.N. (2020). SISTEMA WEB PARA LA GESTIÓN ACADÉMICA Y ADMINISTRATIVA DE EMPRESA DE CAPACITACIÓN PROFESIONAL DIENAV.Ecuador. Universidad Tecnológica Israel.
<http://repositorio.uisrael.edu.ec/bitstream/47000/2489/1/UISRAEL-EC-SIS-378.242-2020-007.pdf>

Anexos

Anexo 1. Glosario

Gestión

“Acción y efecto de gestionar” (Asale (s. f.)).

Gestionar

“Llevar adelante una iniciativa o un proyecto” (Asale (s. f.)).

Interfaz

“Conexión, física o lógica, entre una computadora y el usuario, un dispositivo periférico o un enlace de comunicaciones” (Asale (s. f.)).

Lenguaje de programación

“Conjunto de instrucciones codificadas que una computadora interpreta y ejecuta directamente” (Asale (s. f.)).

Metodología

“Conjunto de métodos que se siguen en una investigación científica o en una exposición doctrinal” (Asale (s. f.)).

Metodología ágil

“Enfoque de trabajo, propio de la gestión de proyectos. Este permite adaptar el trabajo o desarrollo que se pretende hacer, con base en las actividades por hacer, en proceso o terminadas” (Aguirre, 2022).

Modelo

“Arquetipo o punto de referencia para imitarlo o reproducirlo” (Asale (s. f.)).

Relación

“Correspondencia que existe entre dos conjuntos” (J & Gardey, 2021).

Sitio web

“Conjunto de páginas web agrupadas bajo un mismo dominio de internet” (Asale (s. f.)).

Sistema operativo

“Programa o conjunto de programas que realizan funciones básicas y permiten el desarrollo de otros programas”(Asale (s. f.)).

Técnica

“Conjunto de procedimientos y recursos de que se sirve una ciencia o un arte”(Asale (s. f.)).

Anexo 2. Abreviaturas

CSS	Hojas de Estilo en Cascada
HTML	Lenguaje de Marcado de Hipertexto
MVC	Modelo Vista Controlador
MySQL	Mi Lenguaje de Consulta Estructurado
PDO	Objetos de Datos de PHP
PHP	Pre-procesador de Hipertexto
SQL	Lenguaje de Consulta Estructurado
UML	Lenguaje Unificado de Modelado
VSC	Visual Studio Code
XAMPP	Multiplataforma, Apache, MySQL, PHP, Perl