

ESCUELA POLITÉCNICA DEL EJÉRCITO

DPTO. DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB ACADÉMICO- ADMINISTRATIVA PARA LA ESCUELA SION INTERNATIONAL CHRISTIAN SCHOOL, MEDIANTE LA UTILIZACIÓN DE LA PLATAFORMA JAVA ENTERPRISE APPLICATION JEE5 WEB APLICANDO LA METODOLOGÍA U.W.E.

Previa a la obtención del Título de:

INGENIERO EN SISTEMAS E INFORMÁTICA

POR:

ALEXIS PATRICIO ALCOCER ERAZO

PAÚL ABELARDO CUICHÁN AYO

SANGOLQUÍ, Diciembre 2012

CERTIFICACIÓN

Certifico que el presente trabajo fue Realizado en su totalidad por los Señores ALEXIS PATRICIO ALCOCER ERAZO y PAÚL ABELARDO CUICHÁN AYO como requerimiento parcial a la obtención del título de INGENIEROS EN SISTEMAS E INFORMÁTICA

Diciembre del 2012

Ing. Mario Ron

ESCUELA POLITÉCNICA DEL EJÉRCITO

Ingeniería en Sistemas e Informática

AUTORIZACIÓN

Nosotros:

Alexis Patricio Alcocer Erazo

Paúl Abelardo Cuichán Ayo

Autorizamos a la Escuela Politécnica del Ejército la publicación, en la biblioteca virtual de la institución del trabajo “ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB ACADÉMICO-ADMINISTRATIVA PARA LA ESCUELA SION INTERNATIONAL CHRISTIAN SCHOOL, MEDIANTE LA UTILIZACIÓN DE LA PLATAFORMA JAVA ENTERPRISE APPLICATION JEE5 WEB APLICANDO LA METODOLOGÍA U.W.E.” cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría

Sangolquí Diciembre del 2012.

Alexis Alcocer

Paúl Cuichán

DEDICATORIA

Esta tesis está dedicada a nuestros padres, quienes con su Amor y Sacrificios, supieron guiarnos y apoyarnos para la culminación de nuestra carrera universitaria.

Alexis Patricio Alcocer Erazo

Paúl Abelardo Cuichán Ayo

AGRADECIMIENTOS

La presente Tesis va dirigida con una expresión de gratitud para nuestros distinguidos Profesores, que con nobleza y entusiasmo vertieron sus conocimientos en nosotros.

Y a nuestra querida Institución porque en sus aulas recibimos los más grandes e inolvidables recuerdos.

Alexis Patricio Alcocer Erazo

Paúl Abelardo Cuichán Ayo

Tabla de Contenidos

1.1. Tema	18
1.2. Introducción	18
1.3. Planteamiento del problema	19
1.4. Antecedentes.....	20
1.5. Justificación e importancia	20
1.6. Objetivos.....	21
1.6.1. Objetivo general.....	21
1.6.2. Objetivos específicos	22
1.7. Alcance.....	22
1.8. Metodología	22
1.9. Herramientas	23
1.10. Factibilidad.....	23
1.10.1. Factibilidad técnica	23
1.10.2. Factibilidad operativa	24
1.10.3. Factibilidad económica.....	24
2.1. Aplicaciones Web	25
2.1.1. Introducción	25
2.1.2. Definición	25
2.1.3. Características.....	26
2.2. Herramientas y Bases de datos	26
2.2.1. Herramientas de modelado.....	26
2.2.1.1. MagicDrawUML	26
2.2.1.2. PowerDesigner	27
2.2.2. Herramientas de Desarrollo	28
2.2.2.1. Lenguaje de Programación Java.....	28
2.2.2.2. IDE de desarrollo Netbeans	28
2.2.2.3. Java Server Faces (JSF)	29
2.2.2.3.1. Introducción	29
2.2.2.3.2. Definición	30

2.2.2.3.3. Características	30
2.2.2.4. Icefaces	30
2.2.2.4.1. Introducción	30
2.2.2.4.2. Definición	30
2.2.2.4.3. Características	31
2.2.2.5. PostgreSQL	31
2.2.2.5.1. Introducción	31
2.2.2.5.2. Definición	32
2.2.2.5.3. Características	32
2.2.2.6. Servidor de Aplicaciones Glassfish	33
2.2.3. Pruebas de software	33
2.2.4. Bases de Datos	34
2.2.4.1. Estructura de una Base de Datos	34
2.2.4.2. Tipos de Bases de datos.....	35
2.2.4.3. Modelos de Bases de Datos	35
2.2.4.4. Diseño de una base de datos relacional	36
2.3. METODOLOGÍA U.W.E.....	37
2.3.1. Definición.....	37
2.3.2. Características.....	37
2.3.3. Fases de la Metodología UWE.....	37
2.3.3.1. Análisis de Requisitos	38
2.3.3.2. Modelo Conceptual	38
2.3.3.3. Modelo Navegacional.....	38
2.3.3.4. Modelo de Presentación	39
2.3.3.5. Modelo de Tareas	39
2.3.4. Artefactos	39
2.3.4.1. Diagramas de Casos de Uso	39
2.3.4.2. Diagrama de Clases	39
2.3.4.3.1. Diagrama de Secuencia.....	40
2.3.4.3.2. Diagrama de Estado	40
2.3.4.5. Diagrama de Despliegue.....	41
2.3.4.6. Diagrama Navegacional.....	41
2.3.4.7. Diagrama de Presentación.....	41
2.3.4.8. Diagrama de Actividades	42

2.4. Lenguaje de Modelamiento Unificado (UML)	42
2.4.1. Diagramas de Estructuras.....	43
2.4.2. Diagramas de comportamiento	43
2.4.3. Diagramas de Interacción	44
2.5. Ingeniería de Requerimientos	45
2.6. Lenguaje de programación orientado a objetos	46
2.6.1. Introducción	46
2.6.2. Definición	46
2.6.3. Pilares de la programación orientada a objetos	46
3.1 Captura de Requerimientos	48
3.1.1 Estándar IEE830 especificación de Requerimientos	48
3.1.1.1. Introducción	48
3.1.1.2. Identificación de usuarios participantes.....	48
3.1.1.3. Catálogo de requisitos del sistema.....	48
3.1.1.3.1. Objetivos y alcance del sistema	49
3.1.1.3.2. Definiciones, acrónimos y abreviaturas	49
3.1.1.3.3. Descripción general	50
3.1.1.3.4. Requisitos funcionales:	51
3.1.1.3.4.1 Módulo de Administración Académica.....	51
3.1.1.3.4.1.1 Materias (Malla Curricular)	51
3.1.1.3.4.1.2 Contenido.....	53
3.1.1.3.4.1.3 Cursos	55
3.1.1.3.4.1.4 Período Académico.....	56
3.1.1.3.4.1.5 Docentes.....	58
3.1.1.3.4.1.6 Aulas.....	61
3.1.1.3.4.1.7 Organización curricular	62
3.1.1.3.4.1.8 Inscripción.....	64
3.1.1.3.4.1.9 Matriculación	70
3.1.1.3.4.1.10 Evaluación Estudiantes	77
3.1.1.3.4.1.11 Ficha de Seguimiento de un Estudiante	81
3.1.1.3.4.1.12 Asistencia Docentes.....	83
3.1.1.3.4.1.13 Asistencia Estudiantes	84
3.1.1.3.4.1.14 Comunicados	86
3.1.1.3.4.2. Módulo de Administración Logística y Financiera.....	86

3.1.1.3.4.2.1 Inventarios	86
3.1.1.3.4.2.2 Transporte.....	88
3.1.1.3.4.2.3 Productos.....	92
3.1.1.3.4.2.4 Proveedores.....	93
3.1.1.3.4.2.5 Pedido.....	95
3.1.1.3.4.2.6 Entrada de Producto	96
3.1.1.3.4.2.7 Presupuesto.....	98
3.1.1.3.4.2.8 Facturación Servicios.....	99
3.1.1.3.4.2.9 Facturación Pedidos	102
3.1.1.3.4.3 Módulo de Seguridad	103
3.1.1.3.4.3.1. Usuarios.....	103
3.1.1.3.4.4. Parámetros	106
3.1.1.3.4.5. Suposiciones y dependencias.....	106
3.1.1.3.4.6. Requisitos de usuarios y tecnológicos.....	106
3.1.1.3.4.7. Requisitos de interfaces externas	107
3.1.1.3.4.8. Requisitos de rendimiento.....	107
3.1.1.3.4.9. Requisitos de desarrollo.....	107
3.1.1.3.4.10. Restricciones de diseño	108
3.2 Definición de Actores.....	109
3.3 Modelo de Casos de Uso y Descripción de casos de uso.....	114
3.3.1 Plan de estudios	114
3.3.2 Asignación académica	128
3.3.3. Admisión.....	132
3.3.3.1 Inscripción.....	132
3.3.4. Evaluación	142
3.3.5 Asistencia	149
3.3.6 Comunicar Eventos.....	152
3.3.7 Logística	153
3.3.7.1 Inventario.....	153
3.3.7.2 Transporte	156
3.3.7.3 Almacén.....	159
3.3.8 Financiera.....	166
3.3.8.1 Presupuesto.....	166
3.3.8.2 Facturación.....	168

3.3.9 Seguridad	173
3.3.10 Parámetros	176
3.4 Diagrama de Clases	177
3.5 Diagramas de Secuencia	177
3.6 Diagramas de Estado	177
3.7 Diagrama de Despliegue	178
3.8 Diagramas Navegacionales	178
3.9 Diagramas de Presentación.....	178
3.10 Diagramas de Actividades	178
3.11 Diagramas Conceptual de la Base de Datos.....	178
3.12 Diagramas Físico de la Base de Datos	178
4.1 Arquitectura a Implementarse	179
4.2 Estándares de Implementación.....	180
4.2.1 Estándares de codificación	180
4.2.1.1 Estructura del archivo	180
4.2.1.2 Nomenclatura de nombres.....	180
4.2.1.3 Declaración de variables.....	181
4.2.1.4 Sentencias	181
4.3 Desarrollo de los módulos del sistema.....	181
4.4 Integración del Código	182
4.5 Pruebas	182
4.5.1 Pruebas de Caja Blanca	182
4.5.1.1 Buscar usuario para logearse	182
4.5.1.2 Buscar el código del último estudiante inscrito.....	184
4.5.1.3 Buscar matrícula de un estudiante en un período académico específico	185
4.5.1.4 Buscar si existe una solicitud de transporte para una matrícula del período lectivo en curso.....	186
4.5.1.5 Buscar el código del último pedido generado.....	187
4.5.1.6 Verificar que la fecha no posea registros de asistencias de docentes.....	188
4.5.2 Pruebas de caja negra.....	189
4.5.2.1 Guardar Docente	189

4.5.2.2 Modificar Contraseña	190
4.5.2.3 Guardar Curso	190
4.5.2.4 Buscar Aula	190
4.5.2.5 Guardar Producto	191
4.5.2.6 Guardar Inscripción.....	191
4.5.3 Pruebas de Stress	193
5.1 Conclusiones	197
5.2 Recomendaciones	198
BIBLIOGRAFÍA Y WEBGRAFÍA	199

Listado de Tablas

Tabla 3. 1 : Definición de actores.....	109
Tabla 3. 2. : Descripción casos de uso Actualizar Materia	114
Tabla 3. 3. : Descripción casos de uso Buscar Materia	116
Tabla 3. 4. : Descripción casos de uso Actualizar Contenido	117
Tabla 3. 5. : Descripción casos de uso Buscar Contenido	118
Tabla 3. 6. : Descripción casos de uso Actualizar Curso	119
Tabla 3. 7. : Descripción casos de uso Buscar Curso	120
Tabla 3. 8. : Descripción casos de uso Actualizar Período Académico	121
Tabla 3. 9. : Descripción casos de uso Actualizar Rango Nota	122
Tabla 3. 10. : Descripción casos de uso Actualizar Aula	123
Tabla 3. 11. : Descripción casos de uso Buscar Aula	124
Tabla 3. 12. : Descripción casos de uso Actualizar Docente	125
Tabla 3. 13. : Descripción casos de uso Buscar docente	127
Tabla 3. 14. : Descripción casos de uso Asignación de aulas a cursos	128
Tabla 3. 15. : Descripción casos de uso Asignación de recursos a aulas	129
Tabla 3. 16. : Descripción casos de uso Asignación de Docentes a materias, cursos y paralelos	130
Tabla 3. 17. : Descripción casos de uso Actualizar Padre de Familia	132
Tabla 3. 18. : Descripción casos de uso Búsqueda de los Padres de Familia.....	134
Tabla 3. 19. : Descripción casos de uso Actualizar estudiante inscrito.....	135
Tabla 3. 20. : Descripción casos de uso Búsqueda de los estudiantes.	137
Tabla 3. 21. : Descripción casos de uso Matriculación.....	138
Tabla 3. 22. : Descripción casos de uso Actualizar Solicitud Servicio	140
Tabla 3. 23. : Descripción casos de uso Actualizar Presentación de Documentos...	141
Tabla 3. 24. : Descripción casos de uso Abrir Período para el registro de Notas	143
Tabla 3. 25. : Descripción casos de uso Cerrar Período para el Registro de Notas.	143
Tabla 3. 26. Descripción casos de uso Registrar Nota	144
Tabla 3. 27. : Descripción casos de uso Corregir Nota	145
Tabla 3. 28. : Descripción casos de uso Notificación de estudiantes al DOBE.....	146
Tabla 3. 29. :Descripción casos de uso Actualizar la Ficha de Seguimiento de un estudiante matriculado	147
Tabla 3. 30. :Descripción casos de uso Registrar cumplimiento contenido.	148
Tabla 3. 31. : Descripción casos de uso Registrar Asistencia Docente.....	149
Tabla 3. 32. : Descripción casos de uso Registrar Asistencia Estudiante	150
Tabla 3. 33. : Descripción casos de uso Justificar Falta	151

Tabla 3. 34. :Descripción casos de uso Comunicar evento.....	152
Tabla 3. 35. : Descripción casos de uso Actualizar Inventario	153
Tabla 3. 36. : Descripción casos de uso Buscar Inventario	154
Tabla 3. 37. : Descripción casos de uso Actualizar Ruta de Transporte	156
Tabla 3. 38. : Descripción casos de uso Búsqueda de rutas de transporte	157
Tabla 3. 39. : Descripción casos de uso Asignación de ruta de transporte a estudiante.	158
Tabla 3. 40. : Descripción casos de uso Actualizar Producto	160
Tabla 3. 41. : Descripción casos de uso: Buscar Producto	161
Tabla 3. 42. : Descripción casos de uso Actualizar Proveedor	162
Tabla 3. 43. : Descripción casos de uso: Buscar Proveedor	163
Tabla 3. 44. : Descripción casos de uso Registrar Pedido	164
Tabla 3. 45. : Descripción casos de uso Registrar Entrada de Productos	165
Tabla 3. 46. : Descripción casos de uso Registrar Presupuesto Anual	166
Tabla 3. 47. : Descripción casos de uso Reajustar Presupuesto.	167
Tabla 3. 48. : Descripción casos de uso Actualizar Servicio.	168
Tabla 3. 49. : Descripción casos de uso Buscar Servicio	170
Tabla 3. 50. : Descripción casos de uso Registro de Pago de Pedidos	171
Tabla 3. 51. : Descripción casos de uso Registrar Cobro de Servicio	172
Tabla 3. 52. : Descripción casos de uso: Actualizar Usuario.....	173
Tabla 3. 53. : Descripción casos de uso: Cambiar Contraseña.....	175
Tabla 3. 54. : Descripción casos de uso: Validar Usuario	175
Tabla 3. 55. : Descripción casos de uso: Actualizar Parámetro.....	177
Tabla 4. 1 : (Prueba de caja blanca - Buscar usuario para logearse).....	183
Tabla 4. 2. :(Prueba de caja blanca - Buscar el código del último estudiante inscrito)	184
Tabla 4. 3. :(Prueba de caja blanca - Buscar matrícula de un estudiante en un período académico específico)	185
Tabla 4. 4. :(Prueba de caja blanca - Buscar si existe una solicitud de transporte para una matrícula del período lectivo en curso).....	186
Tabla 4. 5. :(Prueba de caja blanca - Buscar el código del último pedido generado)	187
Tabla 4. 6. :(Prueba de caja blanca - Verificar que la fecha no posea registros de asistencias de docentes)	188
Tabla 4. 7. : (Pruebas de caja negra-Guardar docente).....	189
Tabla 4. 8. : (Pruebas de caja negra-Modificar Contraseña).....	190
Tabla 4. 9. : (Pruebas de caja negra-Guardar Curso).....	190

Tabla 4. 10. : (Pruebas de caja negra-Buscar Aula)	190
Tabla 4. 11. : (Pruebas de caja negra-Guardar Producto)	191
Tabla 4. 12. : (Pruebas de caja negra- Guardar Inscripción)	192

Listado de Figuras

Figura 3. 1 : (Caso de Uso – Currícula)	114
Figura 3. 2: (Caso de Uso – Asignación académica)	128
Figura 3. 3: (Caso de Uso – Admisión)	132
Figura 3. 4: (Caso de Uso – Evaluación)	142
Figura 3. 5: (Caso de Uso – Asistencia).....	149
Figura 3. 6: (Caso de Uso – Comunicar Evento).....	152
Figura 3. 7: (Caso de Uso – Inventario)	153
Figura 3. 8: (Caso de Uso –Transporte).....	156
Figura 3. 9: (Caso de Uso –Almacén)	159
Figura 3. 10: (Caso de Uso – Presupuesto)	166
Figura 3. 11: (Caso de Uso - Seguridad)	173
Figura 3. 12: (Caso de Uso - Parámetros)	176
Figura 4. 1. : (Arquitectura a Implementarse).....	179
Figura 4. 2. :(Estructura del archivo Bean).....	180
Figura 4. 3. :(Declaración de variables)	181
Figura 4. 4. : (Sentencias)	181
Figura 4. 5. : (Código Junit testbuscarUsuarioLogin)	182
Figura 4. 6. : (Resultado Junit testbuscarUsuarioLogin)	183
Figura 4. 7. : (Código Junit testRetornaUltimoCodigoEstudiante)	184
Figura 4. 8. : (Resultado Junit testRetornaUltimoCodigoEstudiante)	184
Figura 4. 9. : (Código Junit testExisteMatricula).....	185
Figura 4. 10. : (Resultado Junit testExisteMatricula).....	185
Figura 4. 11. : (Código Junit testExisteSolicitudtransporte).....	186
Figura 4. 12. : (Resultado Junit testExisteSolicitudtransporte).....	186
Figura 4. 13. : (Código Junit testRetornaUltimoCodigoPedido).....	187
Figura 4. 14. : (Resultado Junit testRetornaUltimoCodigoPedido)	187
Figura 4. 15. : (Código Junit testAceptaFechaAsistenciadocente)	188
Figura 4. 16. : (Resultado Junit testAceptaFechaAsistenciadocente)	188
Figura 4. 17. : (Resultado Junit testAceptaFechaAsistenciadocente)	194
Figura 4. 18. :(Pruebas de Stress -Response data de la página login).....	194
Figura 4. 19. :(Pruebas de Stress-Gráfico tiempos de respuesta).....	195
Figura 4. 20. :(Pruebas de Stress- Tiempo Máximo y Mínimo de respuesta)	195

Listado de Anexos

ANEXO A: Diagrama de Clases, Diagramas de Secuencia, Diagramas de Estado. Diagrama de Despliegue, Diagramas Navegacionales, Diagramas de Presentación, Diagramas de Actividades, Diagrama Conceptual de la base de datos, Diagrama Físico de la base de datos.	196
ANEXO B: Manual técnico del sistema SSION	196
ANEXO C: Manual del usuario del sistema SSION.....	196
ANEXO D: Manual del Programador del Sistema SSION	196
ANEXO E: Carta de auspicio, Carta de Aceptación	196

RESUMEN

El presente proyecto de Tesis tiene como finalidad el desarrollo de una aplicación para la Escuela SION International Christian School cuyo principal propósito será la automatización de los procesos académicos y administrativos.

La aplicación a desarrollarse permitirá disponer de información en tiempo real que sea confiable y oportuna, permitiendo mejorar la toma de decisiones y agilizando la ejecución de los procesos.

Para la construcción de la aplicación se utilizará la norma IEEE 830 en lo que se refiere al levantamiento de requerimientos. Para el diseño y modelamiento de la aplicación se utilizará a la metodología UWE debido a que es una metodología orientada a objetos.

INTRODUCCIÓN

1.1. Tema

Análisis, Diseño e Implementación de una Aplicación Web Académico-Administrativa para la Escuela SION International Christian School, mediante la utilización de la plataforma JAVA Enterprise Edition JEE5 web aplicando la Metodología UWE.

1.2. Introducción

El vertiginoso avance de los sistemas de información y comunicaciones en el Ecuador, ha facilitado la automatización de los procesos, brindando confiabilidad, velocidad y precisión en el manejo de cada proceso automatizado. Desde este enfoque se puede decir que el uso eficiente y oportuno de la tecnología en la automatización de procesos define una gran ventaja competitiva.

A través del tiempo la tecnología ha demostrado su gran efectividad en el mejoramiento de las actividades que se realizan en diversas organizaciones tanto públicas como privadas. Los sistemas informáticos son herramientas tecnológicas que en los últimos años han evolucionado enormemente dando nuevas posibilidades para mejorar la eficiencia y eficacia de una empresa.

En esta nueva era, las organizaciones han alcanzado a comprender la gran importancia de la información y administrarla es una prioridad, es por eso que se invierten cantidades extraordinarias de dinero en la adquisición de sistemas seguros, robustos y fiables.

Las organizaciones que no comprenden la importancia de su información y la de apoyar sus tareas en nuevas herramientas tecnológicas llegan a aislarse y quedar con el tiempo obsoletas. Generalmente estas

organizaciones llegan a sucumbir debido a su baja competitividad frente a otras organizaciones que si están preparadas para enfrentarse a las exigencias que el medio les impone. La adquisición o desarrollo de aplicaciones no es un lujo, es una necesidad para brindar a los clientes una imagen institucional fuerte, de prestigio y sobre todo confiable.

Hoy las organizaciones se encuentran frente a una sociedad exigente, en donde tienen que ofrecer productos y servicios de alta calidad que brinden satisfacción a sus clientes. Las organizaciones que han comprendido esto, saben que el único modo de sobrevivir y adaptarse al medio que los rodea es apoyar el desarrollo de sus tareas con herramientas tecnológicas, tales como los sistemas informáticos que actualmente han llegado a ser el puntal principal de las organizaciones, el éxito o el fracaso de estas, depende mucho de la información que ofrezcan los sistemas.

La información que suministran los sistemas a cualquier organización puede llegar a ser muy crítica y es vital que las instituciones dispongan de nuevos sistemas que provean información global y resumida de forma rápida, segura y verídica.

Conscientes de la importancia de la tecnología, la escuela SION International Christian School busca mejorar su efectividad al llevar a cabo sus procesos mediante la integración de un sistema que será elaborado en la presente tesis.

1.3. Planteamiento del problema

El problema surge debido a la carencia de un sistema que automatice los procesos académicos-administrativos que se ejecutan en la escuela SION International Christian School de la ciudad de Quito.

Actualmente la información académica y administrativa de la escuela SION se encuentra desorganizada y esparcida en diferentes documentos físicos y digitales, lo que produce lentitud en la recopilación de información y dificulta la ejecución de los procesos.

Los problemas se agudizan debido a la falta de información histórica que se produce por la carencia de un sistema que les permita el almacenamiento organizado y centralizado de la información.

1.4. Antecedentes

La Escuela SION International Christian School, creada el 1 de Septiembre de 1998, se ha empeñado en cumplir su responsabilidad con la sociedad, formando estudiantes con altos niveles de conocimientos, conjugándola siempre con valores éticos y religiosos.

La escuela SION International Christian School, se fundamenta en el respeto, la armonía y el compromiso en ofrecer un servicio de calidad a sus estudiantes y padres de familia, que responsables y conocedores del noble trabajo de la institución confían la educación de sus hijos.

Su misión es formar individuos que estén dispuestos a distinguirse del mundo y cuyo propósito sea el cambiar la sociedad. Para ello la escuela brinda una formación integral que le permita al individuo crecer en sabiduría, una formación que integra el conocimiento y el desarrollo de destrezas con el amor a Dios y a su prójimo.

1.5. Justificación e importancia

Actualmente, la Escuela SION International Christian School ubicada en la ciudad de Quito no cuenta con un sistema que cumpla con los requerimientos fundamentales que la escuela exige y por tanto en muchas

ocasiones ha tenido problemas en realizar tareas críticas como llevar el historial de un estudiante a través de los años, obtener distintos datos de los respectivos estudiantes y otras tareas que actualmente se las ejecuta de forma lenta y complicada.

Los requerimientos de información de la escuela son un punto crítico para la institución, ya que como cualquier organización dependen de la información para cumplir de forma eficiente el logro de sus actividades.

Debido a la problemática que se genera a partir de la ausencia de un sistema informático que brinde un soporte global para cada una de las actividades que en la escuela se lleva de forma periódica, es necesario de forma urgente implementar un sistema que satisfaga los requerimientos de la Escuela.

Es por eso que de forma inmediata se debe dar solución a los requerimientos de la escuela, para que esta institución pueda ofrecer a sus estudiantes, profesores y padres de familia una plataforma que pueda ofrecer datos confiables y que garanticen una efectiva administración.

El sistema ayudara a la administración de información, generación de reportes, registro de calificaciones, control de asistencia, con lo cual se pretende cumplir con las expectativas y necesidades que tiene la Escuela SION International Christian School.

1.6. Objetivos

1.6.1. Objetivo general

“Analizar, Diseñar e Implementar una aplicación web académico-administrativa para la Escuela SION International Christian School, mediante

la utilización de la Plataforma JAVA Enterprise Edition JEE5 Web Aplicando la Metodología U.W.E.”.

1.6.2. Objetivos específicos

- Analizar y automatizar, los procesos administrativos y académicos que se realizan en la Escuela SION International Christian School.
- Desarrollar, un ambiente Web en el que los miembros de la Escuela SION International Christian School puedan disponer de información en forma rápida y confiable.
- Probar e implementar el sistema con todas las funcionalidades que requiere la Escuela SION International Christian School.

1.7. Alcance

El tema “Análisis, Diseño e Implementación de una aplicación web académico-administrativa para la Escuela SION International Christian School, mediante la utilización de la plataforma JAVA Enterprise Edition JEE5 web aplicando la Metodología UWE” comprende de los siguientes módulos:

- Módulo de Administración Académica.
- Módulo de Administración Logística y Financiera.
- Módulo de Seguridad de Acceso al Sistema.

1.8. Metodología

El uso de una metodología para la creación de una aplicación es indispensable porque ofrece las pautas para llevar de una manera organizada y sistemática las fases y artefactos necesarios para su desarrollo.

En las aplicaciones Web es muy importante un buen análisis y diseño del sistema para cumplir con todos los requerimientos del cliente, por lo cual

se ha decidido utilizar la Metodología UWE (Metodología de Desarrollo Web basada en UML), apoyándose en el lenguaje de modelamiento UML para la creación de los distintos artefactos a realizarse en cada una de las fases de desarrollo.

En la parte teórica se da una descripción más detallada de la metodología.

1.9. Herramientas

Las herramientas necesarias para el desarrollo del sistema son:

- Se utilizará Netbeans, el cual es un entorno de desarrollo java y cumple perfectamente con las exigencias del sistema a desarrollar.
- En lo concerniente a la Base de datos del sistema se utilizará PostgreSQL debido a seguridad, estabilidad y escalabilidad.
- Los sistemas operativos en los cuales van a ser instaladas las herramientas de desarrollo serán Microsoft Windows 7 Home y Ultimate por ser estable y el más difundido en la actualidad.
- El sistema desarrollado se hospedará en un servidor de aplicaciones glassfish, el cual estará instalado en un sistema operativo Linux. En el mismo sistema operativo se encontrará la base de datos del sistema desarrollado.

1.10. Factibilidad

1.10.1. Factibilidad técnica

La Escuela SION International Christian School otorgará la información necesaria para el desarrollo de las páginas web así como de la base de datos, también dará a conocer sus requerimientos.

El Hardware con que se cuenta para la realización del proyecto es:

- ✓ 2 Laptops, cuyas características principales son:

Marca: HP

Procesador: Intel core 2 duo 64 bits

Memoria RAM: 4 GB

Disco Duro: 160 GB.

Requisitos en redes y equipos de comunicación

- ✓ Tarjetas de red 10/100 RJ-45.
- ✓ 3 cables punto-punto UTP RJ-45 CAT5E.
- ✓ 1 Hub de 8 puertos 10/100 Mbps.

1.10.2. Factibilidad operativa

Las instalaciones en las que se va llevar a cabo el levantamiento y aprobación de los requisitos del proyecto serán en las oficinas de la Escuela SION International Christian School.

La consultoría y seguimiento de este proyecto se realizará por parte de los coordinadores de tesis, proporcionados por la ESPE, los cuales poseen vasta experiencia en aplicaciones de esta índole.

1.10.3. Factibilidad económica

Para la realización de esta aplicación web se utilizará software libre y versiones trial porque sus licencias son gratuitas, razón por la cual la Escuela SION International Christian School no tendrá que sobrellevar grandes gastos por la adquisición de licencias en herramientas de desarrollo. Tanto las laptops, como los dispositivos de comunicación (tarjetas de red) serán de propiedad de los desarrolladores (tesistas). El servidor en el cual se alojará la aplicación será una PC de escritorio de acuerdo a las capacidades de la Escuela SION.

MARCO TEÓRICO

2.1. Aplicaciones Web

2.1.1. Introducción

Al principio la web era sencillamente un listado de páginas estáticas para su consulta o descarga. Con el paso del tiempo la web incluyó métodos para elaborar páginas que permitieran desplegar información dinámica, es decir que la información se genere a partir de una petición.

El primer método que se diseñó para la elaboración de la web dinámica fue conocido como CGI ¹ (Common Gateway Interface), este era un mecanismo por el cual se podía pasar información entre el servidor y la aplicación.

Debido a que los CGI tenían graves problemas de concurrencia, que eran causados por la excesiva carga que asumía un servidor al atender una petición, se empezaron a desarrollar soluciones alternativas a los CGI.

Para solucionar el problema de los CGI se dieron dos propuestas que son:

[1]

1. Diseñar sistemas mejor integrados con el servidor, que eviten la instanciación y ejecución de varios programas.
2. Proveer a los servidores la capacidad de trabajar con algún lenguaje de programación que permita añadir código en las páginas.

A partir de estas dos propuestas, se produce un aumento del número de arquitecturas y lenguajes para desarrollar aplicaciones web.

2.1.2. Definición

¹Common Gateway Interface

Las aplicaciones web son soluciones informáticas que se encuentran codificadas en algún lenguaje de programación y que permiten al usuario interactuar con la información por medio de un navegador web (Mozilla, Internet Explorer, Chrome, etc.).

2.1.3. Características

Las características que poseen las Aplicaciones Web son:

Las Aplicaciones Web evitan la dependencia del Sistema Operativo, ya que se pueden ejecutar mediante un navegador web.

Las Aplicaciones Web optimizan el tiempo y ayudan al ahorro del esfuerzo en el momento de llevar a cabo el proceso de actualización debido a que para actualizar una aplicación web no se necesita hacerlo en cada terminal.

Al no tener que instalarse ningún software no consumen ningún espacio en el disco. Además tampoco consumen recursos, ya que el encargado de ejecutar los procesos es el servidor.

2.2. Herramientas y Bases de datos

2.2.1. Herramientas de modelado

“Son herramientas que se emplean para la creación de modelos de sistemas que ya existen o que se desarrollarán” [2].

2.2.1.1. MagicDrawUML

MagicDraw es una herramienta diseñada para el modelado UML de una manera Visual. Este es principalmente utilizado por los desarrolladores de software para facilitar el análisis y diseño de programación orientada a objetos de los sistemas y sus bases de datos.

MagicDraw es una herramienta CASE² de modelado UML, que brinda soporte para el diseño de aplicaciones web. Esta herramienta está diseñada para mejorar la productividad en el desarrollo de software permitiendo la reducción de costos en valores de tiempo y dinero.

MagicDraw soporta diagramas como: casos de uso, clases, secuencia, estado, actividades, etc.

MagicDraw es una herramienta de Licencia Propietario que fue implementada en Java permitiendo así ser multiplataforma, además de poseer la capacidad de instalación de plugins (complementos) aportándole así nuevas funciones .

MagicUwe es un complemento de MagicDraw que integra todos los modelos de datos destinados a la aplicación de la metodología U.W.E.

2.2.1.2. PowerDesigner

Es una herramienta que permite realizar un buen análisis, diseño y construcción de una base de datos, ya que se enfoca en un modelo de datos tanto a nivel físico como conceptual.

PowerDesigner nació como un producto para ayudar en el diseño de base de datos para los principales RDBMS.³ Actualmente es una herramienta de tipo Propietario desarrollada por la empresa Sybase.

Como características PowerDesigner permite tener una visión total de todas las asociaciones definidas en un modelo, además permite crear asociaciones de una manera simple mediante el método de drag and drop.⁴

² Ingeniería de Software Asistida por computadora (ComputerAided Software Engineering)

³ Sistema de Gestión de Base de Datos Relacional (RelationalDatabase Management System)

⁴ Arrastrar y Soltar

2.2.2. Herramientas de Desarrollo

2.2.2.1. Lenguaje de Programación Java

“Java es un lenguaje de Programación orientada a Objetos.” [3]. Las ideas fundamentales en la creación de Java fueron que sea orientado a objetos, independiente del sistema operativo en el cual se esté trabajando, además de ser seguro para trabajar en red al ejecutar el código de sistemas remotos de forma segura.

La principal característica de este lenguaje es que todo programa desarrollado en java debe ser compilado transformándolo en código bytecodes⁵ y a la vez este es interpretado por una máquina virtual.

En Java las fugas de memoria son controladas mediante un recolector de basura, puesto que este se encarga de liberar la memoria de los objetos que ya no están referenciados.

En java se usa el manejo de excepciones para evitar problemas durante el tiempo de ejecución del programa. Estas excepciones son procedimientos para tratar errores.

Java cuenta con un cargador de clases que permite diferenciar entre las clases que se están accediendo de forma remota y las clases locales.

2.2.2.2. IDE de desarrollo Netbeans

Netbeans es un entorno de desarrollo integrado (IDE), kit de desarrollo de software, con licencia libre, dirigido hacia el lenguaje de programación Java.

La plataforma de desarrollo Netbeans crea las aplicaciones en base a un conjunto de componentes de software denominados módulos. Estos son

⁵Bytecode es un código intermedio más abstracto que el código máquina.

archivos que poseen clases Java las cuales interactúan con la interfaz de programación de Netbeans, además dichos módulos poseen un archivo de identificación para diferenciarse unos de otros.

Netbeans permite a los desarrolladores administrar las interfaces de usuario de una manera visual haciéndolas más sencillas, dejando mayor cabida al enfoque de la lógica específica de su aplicación. Además permite utilizar un sin número de tipos de datos para el procesamiento de información, su interfaz gráfica se basa en ventanas para una mejor navegabilidad, posee la característica del uso de asistentes para la creación de componentes.

Netbeans está escrito en Java pero brinda la posibilidad de trabajar con otros lenguajes de programación para la creación de aplicaciones de escritorio, web, móviles, etc. Al poseer tecnología Java es portable, es decir que no se limita a un sistema operativo en particular.

2.2.2.3. Java Server Faces (JSF)

2.2.2.3.1. Introducción

JSF es un framework que se centra en el desarrollo de interfaces gráficas de usuario.

Permite desarrollar aplicaciones bien estructuradas, ya que realiza una separación entre el comportamiento y presentación, además posee su propio servlet⁶ como controlador, implementando así el patrón Modelo Vista Controlador (MVC).

Debido a que la JSF, posee una librería de etiquetas para componentes UI⁷ y un modelo de programación bien estructurado, ayuda al desarrollo ágil

⁶Servlets son objetos que corren dentro y fuera del contexto de un contenedor de servlets

⁷UI (Interfaz de Usuario)

de las aplicaciones y facilita de forma significativa la tarea de la construcción y mantenimiento de aplicaciones web con Interfaces de Usuario en el lado del servidor.

2.2.2.3.2. Definición

Java Server Faces es un framework de desarrollo de aplicaciones web que implementa el patrón MVC⁸ cuyo objetivo es el diseño ágil de interfaces de usuarios.

2.2.2.3.3. Características

La tecnología JSF da la posibilidad de fabricar aplicaciones web que separan la lógica del negocio de la presentación, esto permite la distribución de tareas para el diseño de la aplicación y proporciona un modelo sencillo de programación para la integración.

“JSF permite crear nuestros propios componentes personalizados, así como generar salidas para diferentes dispositivos clientes.” [4]

2.2.2.4. Icefaces

2.2.2.4.1. Introducción

Icefaces es un framework de desarrollo basado en el estándar Java Server Faces (JSF). Las características innovadoras del framework simplifican el desarrollo y superan las capacidades de estándar Java Server Faces.

2.2.2.4.2. Definición

Icefaces EE es un framework de código abierto que está basado en el estándar JSF, que facilita la creación de aplicaciones de Internet enriquecidas (RIA).

⁸Modelo Vista Controlador (MVC)

2.2.2.4.3. Características

Posee una multitud de funciones que los programadores pueden usar para desarrollar aplicaciones Java EE nuevas o ya existentes, sin ningún tipo de coste, ya que Icefaces posee una multitud de funciones que facilitan y agilizan el desarrollo de las aplicaciones.

Icefaces al ser una extensión de JSF es una herramienta que aprovecha la experiencia en el desarrollo de aplicaciones desarrolladas con Java Server Faces.

Para efectuar la ejecución de las aplicaciones Icefaces no es necesario la instalación de ningún plugin de navegador o applets.

Con Icefaces ya no se envían los formularios como se hacía antiguamente, en un POST de HTTP, sino que sólo se envían los cambios que ha hecho el usuario del cliente al servidor, y los cambios en la pantalla del servidor al cliente.

Icefaces dispone de efectos tales como arrastrar, soltar, parpadear entre otros debido a que tiene la librería Scriptaculous.

2.2.2.5. PostgreSQL

2.2.2.5.1. Introducción

PostgreSql es el Sistema Gestor de Bases de Datos de código abierto.

Los cimientos de PostgreSql fueron hechos por académicos y alumnos universitarios quienes pretendieron desarrollar un gestor de base de datos que brinde una solución a los problemas que poseían los demás gestores de base de datos.

A través del tiempo se desarrolla una comunidad, cuya labor ha sido trabajar sobre los cimientos de PostgreSql y desarrollar un avanzado

Sistema Gestor de Base de Datos que brinde solución y eficiencia a los problemas del mundo empresarial. Con ese fin, se han desarrollado y añadido a PostgreSQL múltiples y útiles características que antes sólo se podían hallar en sistemas manejadores de bases de datos comerciales cuyos costos son muy elevados.

2.2.2.5.2. Definición

Es un poderoso sistema manejador de bases de datos publicado bajo licencia BSD, el cual se puede ejecutar en la mayoría de los Sistemas Operativos más utilizados incluyendo, Linux, Unix y Windows.

2.2.2.5.3. Características

Algunas de las características que posee PostgreSQL son: [5]

PostgreSql cumple la prueba ACID (Atomicity, Consistency, Integrity, Durability) y tiene soporte completo para:

- ✓ Llaves foráneas
- ✓ Joins
- ✓ Vistas
- ✓ Subconsultas
- ✓ Triggers
- ✓ Procedimientos almacenados (en varios lenguajes).
- ✓ Incluye la mayoría de los tipos de datos de los estándares (integer, numeric, boolean, char, varchar, date, interval, timestamp, entre otros).
- ✓ También soporta almacenamiento de objetos grandes (imágenes, sonido y video).

Posee un lenguaje nativo llamado (PL/PGSQL) que es muy similar al PL/SQL de Oracle

2.2.2.6. Servidor de Aplicaciones Glassfish

“Glassfish es un servidor de aplicaciones de software libre desarrollado por Sun Microsystems en 2005, que implementa las tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación.” [6]

Este servidor es independiente del sistema operativo que se vaya a usar, debido a que está programado en Java. Tuvo su auge al fusionarse con los plugins de Netbeans y Eclipse.

Glassfish es un buen servidor para desplegar aplicaciones web con Java y posee compatibilidad con lenguajes dinámicos como PHP, Java Script, Ruby, etc. Ya que posee un contenedor Web que se encarga de manejar dichos componentes.

Glassfish posee varias características entre las cuales se destaca su capacidad de procesar funciones de alta velocidad, escalabilidad, manejo de instancias, bajo consumo de memoria, posee una consola de administración teniendo así una administración centralizada.

2.2.3. Pruebas de software

Las pruebas de software también denominadas testing se realizan cuando se dispone del código ejecutable del sistema, este es evaluado a través de la ejecución en circunstancias controladas, para la búsqueda de errores.

En las pruebas de software se define un conjunto de datos y se procede a hacer corridas de prueba.

Las pruebas de software sirven para la localización y corrección de fallos en el sistema.

Entre los objetivos de las pruebas de software es la de corroborar a las personas que desarrollan y están creando un sistema de calidad, ya que estas pruebas se pueden realizar en cualquier fase de la programación.

Existen diferentes clases de pruebas, como son las estáticas que no necesitan que el código sea ejecutado para su corrección (Ejemplo pruebas de escritorio) y las dinámicas que necesitan la ejecución de la aplicación.

2.2.4. Bases de Datos

“Una base de datos es una serie de datos relacionados que forman una estructura reconocible desde un sistema” [7]. A continuación algunas características de las bases de datos:

Con una base de datos se tiene independencia con la información almacenada, el uso de bases de datos reduce el costo de mantenimiento de los sistemas puesto que el cambio en los datos almacenados no genera modificaciones en el código del programa.

Con el uso de las bases de datos se controla la redundancia de los datos, evitando así tener datos repetidos.

Los datos tienen una gran disponibilidad puesto que estos no son públicos o privados como en programación.

Mejoramiento en la seguridad de los datos puesto que se da acceso a la información dependiendo de los usuarios. Además existe seguridad en las operaciones puesto que estas solo se implementarán en los datos en los cuales tiene acceso dicha operación.

2.2.4.1. Estructura de una Base de Datos

Estas están compuestas de datos y metadatos. Los metadatos son datos que especifican la estructura de la base de datos, dando a conocer el tipo de

dato que se va a almacenar, cual es el nombre de cada dato, como se relacionan, etc.

Existen de esta manera dos visiones de la base de datos:

Estructura Lógica: Es una estructura que permite crear una idea de la base de datos. Está compuesta de objetos, entidades, relaciones, etc.

Estructura Física: Es la estructura de los datos que van a ser almacenados en el disco.

2.2.4.2. Tipos de Bases de datos

Los tipos de bases de datos diferenciados según la variabilidad de sus datos son:

Bases de datos estáticas: Estas bases de datos tienen la característica de ser solo de lectura, cuyo propósito es la de estudiar el comportamiento de dichos datos en un determinado período de tiempo, para una buena toma de decisiones dentro de una empresa.

Bases de datos dinámicas: En este tipo de bases de datos la información almacenada cambia con el transcurso del tiempo, al realizarse sobre estas operaciones de actualización, borrado y adición de datos.

2.2.4.3. Modelos de Bases de Datos

Los modelos de datos intentan captar una información del mundo real y almacenarla en un sistema informático. Los más importantes son:

Bases de datos relacionales: Es el modelo utilizado en la actualidad para administrar datos dinámicos, como su nombre lo indica se basa en relaciones, es decir permite tener interconexiones entre los datos que se encuentran en las tablas, para a través de estas conexiones relacionar los datos de dos tablas. Relaciona tablas entre sí por columnas comunes.

Bases de datos de red: Se basa en representar datos reales mediante el uso de registros (conjunto de atributos) lógicos tomándolos como una entidad y se relacionan entre sí por medio de enlaces en una red.

Base de datos Jerárquica: En este modelo los datos se organizan como un árbol invertido. El cual consta de un nodo padre de información que puede tener varios nodos hijo dándose una relación de uno a varios.

2.2.4.4. Diseño de una base de datos relacional

Fase de Recolección y Análisis de Requerimientos: Los diseñadores del sistema realizan las entrevistas correspondientes para conocer las necesidades de información de cada uno de los usuarios del sistema.

Además se procede al levantamiento de requerimientos funcionales, que serán las operaciones que interactuarán con la base de datos, ya sea creando, modificando o eliminando los datos.

Diseño conceptual: Luego de obtenidos los requerimientos funcionales, se crea un esquema conceptual de la base de datos mediante el modelo de datos conceptual.

En el esquema conceptual se creará una descripción de los requerimientos de cada uno de los usuarios, así como la definición de los tipos de datos, relaciones entre ellos y sus restricciones.

Para el diseño de los esquemas conceptuales se utiliza el modelo entidad-relación, en donde se describen los datos como entidades, vínculos (relaciones) y atributos.

Diseño lógico de la base de datos (transformación de modelo de base de datos): El paso siguiente es implementar la base de datos con un sistema de

gestión de base de datos (DBMS), transformando así el modelo conceptual en modelo de base de datos el cual puede ser relacional, jerárquico, red, etc.

Diseño físico de la base de datos: En esta fase se definen las estructuras de almacenamiento y como van organizados los archivos de la base de datos.

2.3. METODOLOGÍA U.W.E.

2.3.1. Definición

“UWE es un método de ingeniería del software para el desarrollo de aplicaciones web basado en UML⁹. Cualquier tipo de diagrama UML puede ser usado, porque UWE es una extensión de UML.” [8]

2.3.2. Características

- ✓ Uso de una notación estándar (UML)
- ✓ Definición de métodos
- ✓ Especificación de restricciones

2.3.3. Fases de la Metodología UWE

Esta metodología consta de una notación basada en UML y un método que consta de 5 fases que son: [9]

- ✓ Análisis de Requerimientos
- ✓ Modelo Conceptual
- ✓ Modelo Navegacional
- ✓ Modelo de Presentación
- ✓ Modelo de Tareas

⁹ Lenguaje Unificado de Modelado

2.3.3.1. Análisis de Requisitos

El modelo de requerimientos tiene como objetivo principal comprender los procesos a realizarse en el sistema y delimitar su alcance.

Los requerimientos se pueden clasificar en funcionales y no funcionales.

Entre los funcionales existen los de contenido, presentación y usuarios.

Para obtener estos requisitos se usan medios de recolección de información como entrevistas y cuestionarios.

2.3.3.2. Modelo Conceptual

Este modelo muestra cómo se encuentran relacionados los contenidos del sistema. Aquí se especifican las clases y sus relaciones dentro del sistema Web.

Este modelo usa los diagramas de clases para definir la estructura de los datos que se encuentran alojados en el sitio web. En los cuales se representan los conceptos, las unidades de información y usuarios.

Se usan los diagramas de secuencia y de estado para visualizar los mensajes entre objetos y las acciones que llevan a una transición de estado.

2.3.3.3. Modelo Navegacional

Este modelo indica como las páginas web del sitio están relacionadas internamente.

El modelo de navegación se enfoca en los puntos donde el usuario puede llevar a cabo una acción y estos puntos se los llama nodos. Además estudia los enlaces que llevan a un nodo (acción), o que se originaron en el mismo.

El objetivo del modelo de navegación es representar el diseño y estructura de las rutas de navegación al usuario.

2.3.3.4. Modelo de Presentación

En este modelo se representan las clases y los procesos que pertenecen a cada página web. Este modelo permite crear una vista de la interfaz de usuario de la aplicación web.

2.3.3.5. Modelo de Tareas

Este modelo especifica las acciones que realiza cada clase, y este modelo incluye:

- ✓ Las relaciones entre las diferentes clases.
- ✓ Las actividades conectadas con cada proceso.

2.3.4. Artefactos

2.3.4.1. Diagramas de Casos de Uso

Un diagrama de casos de uso es una colección de situaciones que se documentan a partir del punto de vista del usuario. Por lo tanto los casos de uso es una de las herramientas más importantes, en la conversión de los requisitos vistos desde el lado del usuario en requisitos a ser implementados por los profesionales del desarrollo de Software.

La ventaja principal es que debido a la facilidad para interpretarlos, también es útil en la comunicación con el cliente. Es decir el cliente no debe preocuparse por tecnicismos al momento de ayudar en el desarrollo del sistema.

2.3.4.2. Diagrama de Clases

Es un diagrama estático el cual se encarga de describir en forma gráfica la estructura de un sistema usando objetos conceptuales. A los diagramas de clases se les utiliza durante la etapa de análisis y diseño de los sistemas.

El elemento fundamental en este diagrama son las clases, las cuales

son simplemente la representación conceptual del mundo real. Estas clases también deben tener relaciones, las mismas que deben evidenciar la interacción de los objetos en el mundo real.

Entre las relaciones que se pueden añadir a un diagrama de clases están:

- Herencia.
- Composición.
- Agregación.
- Asociación.

2.3.4.3.1. Diagrama de Secuencia

El diagrama de secuencia muestra la secuencialidad de las tareas que se lleva a cabo para cumplir con la funcionalidad de un caso de uso. En este diagrama se puede ver la distribución de los módulos y como el sistema va a actuar cuando un usuario requiera algún caso de uso en especial.

Los diagramas de secuencias se componen de los elementos siguientes:

- El curso de acción.
- Los objetos.
- Los mensajes.
- Los métodos.

2.3.4.3.2. Diagrama de Estado

El diagrama de estado es aquel que muestra el dinamismo de una aplicación. Este diagrama refleja los cambios de estados que sufre un objeto en función a los sucesos y al tiempo.

Este diagrama es usado para encontrar cada una de los caminos que puede tomar un flujo de información luego de ejecutarse cada proceso

(Casos de uso).

Los componentes existentes en el diagrama de estados son:

- Eventos.
- Acciones.
- Actividades.
- Transiciones.
- Estados.

2.3.4.5. Diagrama de Despliegue

Define la relación existente entre los componentes, archivos y elementos de hardware que conforman el sistema.

2.3.4.6. Diagrama Navegacional

Aquí se plantea una topología navegacional que permite ejecutar todas las tareas de un usuario dentro del sistema.

Este diagrama tiene dos finalidades, la primera es la de especificar que objetos dentro del sistema van a ser visitados por el usuario. La segunda es la de definir como se alcanzan estos objetos en la aplicación.

Se toman como referencias los requerimientos expresados en los casos de uso para la creación del diagrama navegacional.

2.3.4.7. Diagrama de Presentación

En el diagrama de presentación se define como está estructurada la interfaz que interactuará con el usuario de la aplicación web y demás elementos que permiten la ejecución de acciones dentro de la aplicación como son botones, textbox, etc.

En este modelo las clases representan la página web de la aplicación y dentro de las mismas se realiza una organización de los elementos de la interfaz de usuario.

2.3.4.8. Diagrama de Actividades

El diagrama de actividades permite ver el comportamiento de cada proceso de la aplicación web y las interfaces que permiten manejar dichos procesos.

En este diagrama se representan los flujos operacionales de los componentes de un sistema web.

2.4. Lenguaje de Modelamiento Unificado (UML)

“UML es un lenguaje de modelado de sistemas de software.” [10] Entre sus funciones permite especificar, construir y documentar un sistema.

UML permite describir los procesos de negocio, funciones del sistema, esquema de bases de datos, etc., de manera que se permita cumplir con todos los requerimientos del sistema. Al ser UML un lenguaje de modelado se utiliza para documentar el sistema creando los artefactos para las diferentes metodologías de desarrollo de software, es decir ayuda a diagramar los requerimientos.

UML es una herramienta visual que permite la creación de artefactos más puntuales como son los diagramas de modelos, esquemas de bases de datos, componentes reusables de software, etc.

Pero UML no solo se enfoca en el diseño de diagramas sino también ayuda a una comprensión detallada del sistema, el cual se lo realiza mediante los casos de uso.

Las bases en las que se enfoca UML son modelar sistemas de software con conceptos de orientación a objetos, utilizar un lenguaje de modelado entendible a nivel de máquina y de personas, poseer una fuerte relación entre los conceptos y los artefactos creados.

Los diagramas UML ayudan en el modelado de los sistemas en lo que corresponde a la definición de clases, objetos, atributos, operaciones y relaciones; además de las colaboraciones existentes entre los diferentes objetos, así como los cambios de estado que este objeto presenta en el transcurso del sistema.

Los diagramas que posee UML se clasifican en: [11]

2.4.1. Diagramas de Estructuras

Los diagramas de estructuras muestran como los elementos definidos en los requerimientos y casos de uso deben ser. Los más importantes son:

- *Diagrama de Clases*: En estos diagramas se definen las clases (objetos) del sistema, sus atributos (características de los objetos), las operaciones que poseen y las relaciones entre estas.
- *Diagrama de Componentes*: Define como los módulos físicos (archivos, librerías, etc.) denominados componentes se relacionan entre sí para formar componentes más complejos.
- *Diagrama de Despliegue*: Define la relación existente entre los componentes, archivos y elementos de hardware que conforman el sistema.

2.4.2. Diagramas de comportamiento

Los diagramas de comportamiento definen la secuencia de estados por los que pasa un objeto dentro del sistema. Los más importantes son:

- *Diagrama de Actividades:* Muestran de manera secuencial las acciones o actividades de los diferentes objetos dentro del sistema. Determina el flujo de trabajo de un proceso desde el inicio hasta el final detallando durante este recorrido rutas de decisiones dentro del sistema.
- *Diagrama de Casos de Uso:* Se basa en la especificación de los diagrama de casos de uso para describir gráficamente que funciones tiene cada uno de los actores o usuarios dentro del sistema, los objetivos y dependencia entre los actores.
- *Diagrama de Estados:* Define los posibles estados de un objeto (que pueden ser estado inicial o estado final) después de cumplirse algún evento o transición.

2.4.3. Diagramas de Interacción

Los diagramas de interacción definen o muestran como un grupo de objetos colaboran entre sí para el logro de un fin, en sistemas podría ser para el logro de un proceso. Estos diagramas muestran a los objetos del sistema, así como los mensajes que se envían entre ellos especificados dentro de los casos de uso o sea el comportamiento de los casos de uso. Los más importantes son:

- *Diagrama de Colaboración:* Aquí se definen gráficamente las relaciones existentes entre los objetos y los mensajes que se intercambian.
- *Diagrama de Secuencia:* Muestran las interacciones entre los objetos en secuencias temporales.

2.5. Ingeniería de Requerimientos

La ingeniería de requisitos es una herramienta cuyo objetivo es satisfacer las expectativas del usuario en base al análisis de las necesidades y negociando una solución razonable libre de ambigüedades.

El proceso de ingeniería de requerimientos se divide en 5 etapas: [12]

- *Identificación de Requisitos:* Es la primera etapa de la ingeniería de requerimientos, la cual aunque parece ser extremadamente simple, es una de las etapas más críticas y complejas. En esta etapa se hace una recolección de todos los requisitos vistos desde el lado del usuario.
- *Análisis de Requisitos y Negociación:* En esta etapa se analizan cada uno de los requisitos agrupándolos por categorías y en función a las necesidades de los clientes. Además cada requisito debe cumplir las propiedades de consistencia y completitud.

Los requisitos deben ser delimitados y libres de dobles interpretaciones.

- *Especificación de requisitos:* Se define un documento o modelo gráfico en el cual se describen las funciones, limitaciones y características que han de controlar el desarrollo del sistema. Es recomendable que en el desarrollo de sistemas grandes se utilicen documentos escritos y modelos gráficos.
- *Modelado del Sistema:* Se desarrollan modelos del sistema, los cuales sirven para comprender y evaluar los componentes y relaciones que existen en el sistema.
- *Validación de Requisitos y gestión de Requisitos:* la validación de requisitos es una etapa de verificación, la cual se asegura de que todos y cada uno de los requisitos hayan sido establecidos, sin

ambigüedades, sin inconsistencias y que los errores detectados en la etapa de análisis de requisitos, hayan sido corregidos.

La etapa de Gestión administra cada uno de los requisitos mediante la asignación de un identificador, el cual permitirá identificar, controlar, dar seguimiento a los requisitos y sus cambios en cualquier momento.

2.6. Lenguaje de programación orientado a objetos

2.6.1. Introducción

En la actualidad, la programación orientada a objetos es utilizada ampliamente en el desarrollo de aplicaciones de escritorio y aplicaciones web debido a que ofrece mejoras en la forma de diseño, desarrollo y mantenimiento del software.

Este paradigma de programación concibe la idea de un mundo lleno de objetos y que las soluciones a los problemas se deben realizar mediante el análisis de los objetos que están involucrados en un sistema.

2.6.2. Definición

La programación orientada a objetos es un paradigma de programación el cual tiende a plantear las soluciones a través del análisis de los objetos que interactúan en un sistema.

2.6.3. Pilares de la programación orientada a objetos

Los pilares de la programación orientada a objetos son: la herencia, abstracción, polimorfismo y encapsulación.

- *Abstracción:* La abstracción es el proceso clave a través del cual se transforma un objeto físico en un objeto conceptual. Es decir que mediante este proceso se obtiene las características relevantes y se identifica los comportamientos de un objeto.

- *Herencia:* Es una propiedad que permite generar una estructura jerárquica (clases y subclases) y también permite que los objetos hereden características y comportamientos a partir de otros ya existentes. La herencia se lo realiza a través de una relación entre una clase general y una más específica. Es decir que una subclase contiene los atributos y métodos de la clase primaria.
- *Encapsulación:* La encapsulación es un mecanismo que garantiza la integridad de los datos que contiene un objeto, mediante el ocultamiento de los componentes internos de una clase. Es decir que trata a una clase como una caja negra, la cual aísla los datos y métodos, de forma que se pueda reutilizar el código de forma sencilla sin la necesidad de saber cómo está constituido o como funciona una clase.
- *Polimorfismo:* El polimorfismo es uno de los conceptos fundamentales de la programación orientada a objetos. Así como la herencia está relacionada con las clases y su jerarquía.

El polimorfismo se relaciona con la posibilidad de definir distintos métodos con un mismo identificador.

ANÁLISIS Y DISEÑO DE LA APLICACIÓN

3.1 Captura de Requerimientos

3.1.1 Estándar IEE830 especificación de Requerimientos

3.1.1.1. Introducción

Esta especificación tiene como objetivos analizar y documentar los requisitos que ayudarán al desarrollo de la aplicación web, para lo cual se llevarán a cabo entrevistas y un estudio de las necesidades actuales de la institución (Escuela SION).

3.1.1.2. Identificación de usuarios participantes

Aquí se definen cada uno de los usuarios que utilizarán el sistema.

- Personal directivo: Formado por la Directora y Director de la Escuela.
- Personal operativo: Formado por la Secretaria, el Administrador del área de almacén, el Administrador del Departamento de Orientación y Bienestar Estudiantil.
- Grupo de Docentes: Formado por los docentes de la institución.

Los usuarios antes mencionados colaborarán activamente en las actividades de desarrollo del sistema, para la correcta definición de sus necesidades y rápida implementación del sistema.

3.1.1.3. Catálogo de requisitos del sistema

El objetivo de la especificación de requerimientos es definir de una manera clara y concisa las necesidades y restricciones del sistema a construirse. Esta documentación estará sujeta a revisiones de los usuarios del sistema detallados anteriormente. Para las revisiones se crearán varias versiones de este documento, las cuales serán aprobadas por los usuarios. Una vez

terminadas las revisiones este documento servirá de base para el diseño y construcción del sistema.

Este modelo de especificación de Requerimientos está basado en el estándar “IEEE Recommended Practice for Software Requirements Specifications (IEEE/ANSI 830-1993)”, y toma como base las entrevistas a los usuarios o interesados y el estudio de la documentación existente.

3.1.1.3.1. Objetivos y alcance del sistema

Los principales objetivos son:

- Analizar y automatizar, los procesos administrativos y académicos que se realizan en la Escuela SION International Christian School.
- Desarrollar un ambiente en el que los miembros de la Escuela SION International Christian School puedan disponer de información de forma rápida y confiable.
- Probar e implementar el sistema con las funcionalidades que requiere la Escuela SION International Christian School.

La institución actualmente presta sus servicios aproximadamente a 300 estudiantes, así como también a docentes y administrativos.

El sistema permitirá a los usuarios el registro de los datos personales de los estudiantes, de sus padres y también de los docentes.

Además permitirá realizar consultas de los estudiantes matriculados, un registro del pago de las matriculas, un control de la asistencia de los estudiantes, el registro de sus calificaciones y registrar las ventas de los productos que expide la escuela (libros, uniformes, materiales, etc.).

3.1.1.3.2. Definiciones, acrónimos y abreviaturas

- Definiciones:

Moroso: Persona que adeuda a la institución.

Recurso: Artículo que posee la escuela (Ejemplo: silla, mesa)

Dado de baja: Que ese objeto no estará disponible para su utilización en alguna asignación.

- Acrónimos:

JPA: Java Persistence Api

DBA: Data Base Administrator

- Abreviaturas:

IEEE: Institute of Electrical & Electronics Engineers

DOBE: Departamento de Orientación y Bienestar Estudiantil

3.1.1.3.3. Descripción general

Esta sección del documento permite visualizar una descripción general de las funciones que se automatizarán en el sistema, los datos con los cuales se trabajará y las limitaciones del mismo.

Las funciones a automatizarse se dividen en dos grupos.

- 1) Administración Académica: permite la gestión de los datos del estudiante, gestión de los datos de los padres de familia, gestión de la matriculación de los estudiantes, gestión de datos de los docentes, gestión de las materias, gestión de asignación de docentes, gestión de las matriculas, control de la asistencia de los estudiantes y gestión de las calificaciones o notas de los estudiantes.
- 2) Administración Logística y Financiera: permite la gestión del cobro de matrículas, inventario, gestión de proveedores y presupuesto.

3.1.1.3.4 Requisitos funcionales:

3.1.1.3.4.1 Módulo de Administración Académica

3.1.1.3.4.1.1 Materias (Malla Curricular)

Requerimiento 1. Creación de Materia

Introducción: El sistema permitirá crear una materia de acuerdo a las disposiciones emitidas por el Ministerio de Educación, los cuales determinarán que materias son las obligatorias para cada curso.

Entradas: Nombre de la materia, Curso, Tipo de materia, Objetivo de la materia, estado de la materia, Observación de la materia.

Procesos:

- La secretaria académica ingresará los datos de una materia.
- La secretaria académica debe solicitar a la aplicación que se realice la creación de una materia.
 - El sistema comprobará que la materia no esté creada, dándola de alta. (No pueden existir dos materias con el mismo código o nombre.)
- El sistema inicializa por defecto las materias con un estado disponible.
- El sistema notificará el éxito o el fracaso de su solicitud.

Salida: Materia creada.

Requerimiento 2. Modificar Materia

Introducción: El sistema permitirá modificar los datos de una materia existente.

Entradas: Nombre de la materia, Curso, Tipo de materia, Objetivo de la materia, estado de la materia, Observación de la materia.

Procesos:

- La secretaria académica debe buscar la materia, mediante el sistema. (Requerimiento 3).
- La secretaria académica registrará nuevos datos para una materia.
 - El sistema permitirá inhabilitar una materia. Cuando una materia es inhabilitada no estará disponible en los siguientes años lectivos. La inhabilitación de las materias lo podrá hacer, la secretaria académica previa orden del rectorado.
- La secretaria académica deberá solicitar a la aplicación que se realice los cambios correspondientes a dicha materia.
- El sistema comunicará el éxito o el fracaso de la solicitud.

Salida: Materia actualizada.

Requerimiento 3. Búsqueda de las Materias de un determinado curso.

Introducción: El sistema permitirá la búsqueda de las materias pertenecientes a un curso.

Entradas: Curso

Procesos:

- El usuario solicitará al sistema buscar las materias existentes a un curso.
 - Los filtros para la búsqueda será el curso.
- El sistema desplegará las materias que correspondan al curso ingresado.

Salida: Listado de materias por curso.

Reporte 1: Listado de Materias por Curso

Introducción: El sistema debe mostrar un listado de las materias que posee un determinado curso. Los campos a mostrarse son: curso, paralelo, nombre materia.

Interesados: Secretaria Académica.

Entradas: Nombre curso.

Procesos: El reporte debe permitir un filtrado por nombre curso.

Salida: Lista completa de todas las materias existentes en un curso.

3.1.1.3.4.1.2 Contenido

Requerimiento 4. Crear contenido

Introducción: El sistema permitirá registrar cada uno de los contenidos que tiene una materia.

Entradas: Materia, Contenido, estado contenido.

Procesos:

- La secretaria académica debe buscar las materias activas de un curso. (Requerimiento 3).
- La secretaria académica debe seleccionar la materia a la cual desea agregar contenido.
- La secretaria académica debe agregar los contenidos que sean necesarios.
- La secretaria académica solicita a la aplicación que se guarde el contenido.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Contenido creado.

Requerimiento 5. Modificar Contenido

Introducción: El sistema permitirá modificar el contenido de una materia.

Entradas: Materia, Contenido, estado contenido.

Procesos:

- La secretaria académica debe buscar los contenidos de una materia.
(Requerimiento 7).
- La secretaria académica debe registrar nuevos datos para el contenido.
- La secretaria académica solicita a la aplicación que se guarden los cambios realizados.
- El sistema notificará el éxito o fracaso de su solicitud.

Salida: Contenido actualizado.

Requerimiento 6. Dar de baja al Contenido

Introducción: El sistema permitirá dar de baja el contenido de una materia que se imparte en un curso.

Entradas: Materia, Contenido, estado contenido.

Procesos:

- La secretaria académica debe buscar los contenidos de una materia.
(Requerimiento 7).
- La secretaria académica debe dar de baja los contenidos que desea cambiando su estado a dado de baja.
- El sistema notificará el éxito o fracaso de su solicitud.

Salida: Contenido dado de baja.

Requerimiento 7. Buscar Contenidos por Materia

Introducción: El sistema permitirá buscar el contenido que posee una materia.

Entradas: Curso, Materia.

Procesos:

- El usuario buscará por medio del sistema la materia que se imparte en un curso. (Requerimiento 3).
- El usuario deberá seleccionar la materia.
- El sistema desplegará todos los contenidos de esa materia.

Salida: Listado de contenidos por materia.

3.1.1.3.4.1.3 Cursos

Requerimiento 8. Creación de Cursos

Introducción: El sistema permitirá crear un curso con su respectivo paralelo.

Entradas: Nombre del curso, paralelo, estado, tipo curso.

Procesos:

- La secretaria académica debe ingresar los datos de un curso.
- La secretaria académica solicita a la aplicación que se realice la creación de un curso.
 - El sistema comprobará que el curso y el paralelo no estén creados, dándolo de alta. (No se pueden repetir los paralelos de un curso).
- El sistema inicializa por defecto los cursos con un estado disponible.
- El sistema comunicará el éxito o fracaso de su solicitud.

Salida: Curso guardado.

Requerimiento 9. Modificar Curso

Introducción: El sistema permitirá modificar los datos de un curso existente.

Entradas: Nombre del curso, paralelo, estado, tipo curso.

Procesos:

- La secretaria académica debe buscar los cursos. (Requerimiento 10).
- La secretaria académica debe registrar nuevos datos para el curso.

- El sistema permitirá inhabilitar un curso con su respectivo paralelo, es decir que este curso y paralelo no estará disponibles en el siguiente año lectivo.
- La secretaria académica solicita a la aplicación que se guarden los cambios correspondientes a dicho curso.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Curso actualizado.

Requerimiento 10. Buscar Curso

Introducción: El sistema permitirá buscar los cursos.

Entrada: Nombre Curso.

Procesos:

- El usuario deberá solicitar la búsqueda de los cursos al sistema.
- El sistema académico desplegará un listado de cursos.

Salida: Curso o listado de cursos.

3.1.1.3.4.1.4 Período Académico

Requerimiento 11. Crear Período Académico

Introducción: El sistema permitirá crear un período académico.

Entrada: Año lectivo, Fecha de inicio del período académico, Fecha de fin del período académico.

Procesos:

- La secretaria académica registrará el período académico.
- La secretaria académica solicita a la aplicación que se guarde el período académico.

- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Período guardado.

Requerimiento 12. Crear Rango de Notas para Aprobación del Año Lectivo

Introducción: El sistema permitirá crear un rango de notas las cuales serán tomadas como referencia para conocer si el estudiante aprobó o reprobó el año lectivo.

Entrada: Período académico, límite inferior, límite superior del rango de nota, especificación de límite inferior, especificación de límite superior, resultado, puntaje requerido para la aprobación del año lectivo.

Procesos:

- La secretaria académica registrará el rango de nota.
- La secretaria académica solicita a la aplicación que se guarde el rango de nota.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Rango de nota guardado.

Requerimiento 13. Modificar Rango de Notas para Aprobación del Año Lectivo.

Introducción: El sistema permitirá modificar los datos de un rango de nota existente.

Entradas: Límite inferior, límite superior del rango de nota, especificación de límite inferior, especificación de límite superior, resultado, puntaje requerido para la aprobación del año lectivo.

Procesos:

- La secretaria académica debe buscar el rango de nota a modificar. (Requerimiento 14).
- La secretaria académica debe registrar nuevos datos para el rango de nota.
- La secretaria académica solicita a la aplicación que se guarden los cambios correspondientes a dicho rango de nota.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Rango de nota actualizado.

Requerimiento 14. Buscar Rango de Notas para Aprobación del Año Lectivo

Introducción: El sistema permitirá buscar los rangos de notas pertenecientes a un año lectivo.

Entrada: Año lectivo.

Procesos:

- El usuario deberá solicitar la búsqueda de los Rangos de notas al sistema.
- El sistema académico desplegará un listado de los rangos de notas pertenecientes a dicho año lectivo.

Salida: Rango de nota o listado de rangos de notas.

3.1.1.3.4.1.5 Docentes

Requerimiento 15. Crear Docente

Introducción: El sistema permitirá crear un docente

Entradas: Cédula, E-Mail, Nombres, Apellidos, Fecha de nacimiento, estado docente, Dirección (Calle principal y Calle secundaria), Barrio o Sector, fecha de ingreso a la escuela, Teléfono, Celular, Estado civil (casado, soltero, divorciado, otros), relación laboral (contratado, por horas), tipo docente (especial, curso).

Procesos:

- El inspector ingresará los datos de un docente.
 - El sistema comprobará que el docente no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos docentes con el mismo número de cédula).
- El inspector solicita a la aplicación que se realice la creación de un docente.
 - El sistema crea una cuenta de usuario con el perfil de docente. La cédula de usuario y contraseña será el número de su cédula.
- El sistema inicializa por defecto los docentes con un estado disponible.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Docente guardado en disco.

Requerimiento 16. Modificar Docente

Introducción: El sistema permitirá modificar los datos de un docente.

Entradas: Cédula, E-Mail, Nombres, Apellidos, Fecha de nacimiento, estado docente, Dirección (Calle principal y Calle secundaria), Barrio o Sector, fecha de ingreso a la escuela, Teléfono, Celular, Estado civil (casado, soltero, divorciado, otros), relación laboral (contratado, por horas), tipo docente (especial, curso).

Procesos:

- El inspector buscará al docente. (Requerimiento 17).
- El inspector actualizará los datos de un docente.
 - El inspector buscará al docente creado, mediante el sistema, para luego proceder a actualizar sus datos.
 - El sistema permitirá inhabilitar un docente, es decir que el docente no estará disponible en los siguientes años lectivos.
- El inspector solicita a la aplicación que se realice los cambios correspondientes a dicho docente.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Docente actualizado.

Requerimiento 17. Buscar Docente

Introducción: El sistema permitirá realizar la búsqueda de los docentes.

Entradas: Cédula del docente.

Procesos:

- El usuario deberá solicitar la búsqueda del docente en el sistema.
 - El usuario debe digitar el número de cédula del docente.
- El sistema desplegará al inspector el docente.
 - En el caso de que el inspector digite (*.*) se deberá desplegar todos los docentes.

Salida: Docente o listado de docentes.

Reporte 2: Listado de Docentes

Introducción: El sistema debe mostrar los datos correspondientes al docente que son: Cédula, E-Mail, Nombres, Apellidos, Fecha de nacimiento, estado

docente, Dirección (Calle principal y Calle secundaria), Barrio o Sector, fecha de ingreso a la escuela, Teléfono, Celular, Estado civil (casado, soltero, divorciado, otros), relación laboral (contratado, por horas), tipo docente (especial, curso), ocupación actual.

Interesados: Inspector.

Entradas: Cédula del docente.

Procesos: El reporte se generará con los docentes con estado disponible.

Salida: Lista completa de todos los docentes registrados del año lectivo en curso.

3.1.1.3.4.1.6 Aulas

Requerimiento 18. Crear Aula

Introducción: El sistema permitirá crear un aula.

Entradas: Nombre del aula, Capacidad, Observación, Estado del aula.

Procesos:

- La secretaria académica ingresará los datos del aula.
 - El sistema comprobará que el aula no haya sido creada con anterioridad, dándolo de alta. (No pueden existir dos aulas con el mismo nombre).
- La secretaria académica solicita a la aplicación que se guarde el aula.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Aula guardada.

Requerimiento 19. Modificar Aula

Introducción: El sistema permitirá modificar los datos que pertenecen a un aula.

Entradas: Nombre del aula, Capacidad, Observación, Estado del aula.

Procesos:

- La secretaria académica buscará el aula en el sistema. (Requerimiento 20).
- La secretaria académica actualizará los datos del aula.
 - La secretaria podrá cambiar el estado de no disponible a disponible e incluso podrá dar al aula de baja.
- La secretaria académica solicitará a la aplicación que se guarden los cambios.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Aula actualizada.

Requerimiento 20. Buscar Aula

Introducción: El sistema permitirá realizar la búsqueda de aulas.

Entrada: Nombre del aula.

Procesos:

- El usuario deberá solicitar la búsqueda del aula en el sistema.
 - El usuario debe digitar el nombre del aula.
- El sistema desplegará al usuario los datos del aula.
 - En el caso de que el usuario digite (*.*) se deberá desplegar todos las aulas activas.

Salida: Aula o listado de aulas.

3.1.1.3.4.1.7. Organización curricular

Requerimiento 21. Asignación de aulas a cursos

Introducción: El sistema permitirá asignar las aulas a los cursos.

Entradas: Curso, Aula.

Procesos:

- La secretaria académica deberá buscar los cursos activos.
(Requerimiento 10).
 - El sistema deberá desplegar un listado de cursos.
- La secretaria académica debe seleccionar un curso.
- La secretaria académica podrá realizar una búsqueda de las aulas existentes.
- La secretaria académica asignará a un curso un aula.
 - El sistema cambiara el estado del aula a no disponible.
- La secretaria académica solicitará al sistema guardar la asignación.
- El sistema notificará el éxito o el fracaso de la solicitud.

Salida: Asignación de aulas a cursos.

Requerimiento 22. Asignación de recursos a aulas

Introducción: El sistema permitirá asignar uno o más recursos a un aula.

Entradas: Aula, Recursos.

Procesos:

- El colector debe buscar las aulas. (Requerimiento 20).
- El sistema debe desplegar el listado de aulas.
- El colector seleccionará el aula.
- El sistema debe buscar los recursos.
- El sistema debe desplegar el listado de recursos.
- El colector podrá asignar los recursos para el aula.
- El colector solicita al sistema guardar la asignación.

Salida: Asignación de recursos a aulas guardado.

Requerimiento 23. Asignación de Docentes a una materia, curso, paralelo.

Introducción: El sistema permitirá asignar un docente a una materia, curso y paralelo.

Entradas: Año lectivo, Curso, Paralelo, Materia, Docente, Fecha.

Procesos:

- La secretaria académica especificará el año lectivo en curso.
- La secretaria académica especificará el curso y paralelo.
- El sistema desplegará un listado de materias concernientes a dicho curso.
- La secretaria académica asignará a cada materia el docente.
 - El sistema permitirá realizar una búsqueda de los docentes disponible. (Requerimiento 17).
- La secretaria académica solicita guardar la asignación.

Salida: Asignación de docentes a una materia, curso, paralelo guardado.

3.1.1.3.4.1.8 Inscripción

Requerimiento 24. Creación de Padres de Familia

Introducción: El sistema permitirá crear un padre de familia.

Entradas: Cédula del padre de familia, Nombres, Apellidos, Sexo, Dirección (Calle principal y Calle secundaria), E-mail, Teléfono, Celular, Nivel académico (Superior, Secundario, Primario), Instituto (Lugar donde estudio), Ocupación, Lugar de trabajo, Puesto de trabajo, Dirección de trabajo, Teléfono del trabajo, Teléfono de emergencia, Situación económica (Muy buena, Buena, Regular), N° Carga familiar, Casa (Propia, Arrendada, Familiares), Tipo de cuenta (Cta. Ahorros o Cta. Corriente), Estado civil

(casado, soltero, divorciado, otros), Congregación (Nombre de la iglesia a la que asiste), Denominación (La descripción de la religión), Iglesia hogar o Célula (Especifica si pertenece a una célula de una iglesia), Nombre del líder (Especifica en caso de que el padre de familia pertenezca a una célula de una iglesia), Nacido de nuevo (si aceptó a Dios en su vida, SI/NO), Responsabilidad (Describe un cargo en la Iglesia en caso de tenerlo).

Procesos:

- La secretaria del departamento de admisión ingresará los datos de un padre de familia.
 - El sistema comprobará que el padre no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos padres con el mismo número de cédula).
- La secretaria del departamento de admisión solicita a la aplicación que se realice la creación de un padre.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Padre de familia guardado en disco.

Requerimiento 25. Modificar Padres de Familia

Introducción: El sistema permitirá modificar los datos de un padre de familia.

Entradas: Cédula del padre de familia, Nombres, Apellidos, Sexo, Dirección (Calle principal y Calle secundaria), E-mail, Teléfono, Celular, Nivel académico (Superior, Secundario, Primario), Instituto (Lugar donde estudio), Ocupación, Lugar de trabajo, Puesto de trabajo, Dirección de trabajo, Teléfono del trabajo, Teléfono de emergencia, Situación económica (Muy buena, Buena, Regular), N° Carga familiar, Casa (Propia, Arrendada,

Familiares), Tipo de cuenta (Cta. Ahorros o Cta. Corriente), Estado civil (casado, soltero, divorciado, otros), Congregación (Nombre de la iglesia a la que asiste), Denominación (La descripción de la religión), Iglesia hogar o Célula (Especifica si pertenece a una célula de una iglesia), Nombre del líder (Especifica en caso de que el padre de familia pertenezca a una célula de una iglesia), Nacido de nuevo (si aceptó a Dios en su vida, SI/NO), Responsabilidad (Describe un cargo en la Iglesia en caso de tenerlo).

Procesos:

- La secretaria debe buscar al padre de familia. (Requerimiento 26).
- La secretaria del departamento de admisión actualizará los datos de un padre de familia.
 - La secretaria del departamento de admisión buscará al padre de familia creado, mediante el sistema, para luego proceder a actualizar sus datos.
- La secretaria del departamento de admisión solicita a la aplicación que se realice los cambios correspondientes a dicho padre.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Padre de familia actualizado.

Requerimiento 26. Búsqueda de los Padres de Familia

Introducción: El sistema permitirá la búsqueda de los padres de familia.

Entrada: Cédula del padre.

Procesos:

- El usuario proveerá al sistema la cédula del padre.
- El usuario solicitará al sistema buscar al padre de familia.

- El sistema desplegará al padre de familia que sea igual o similar.

Salida: Padre.

Requerimiento 27. Inscribir Estudiante

Introducción: El sistema permitirá crear un estudiante.

Entradas: Número de cédula (No obligatoria) o Número de partida de nacimiento (no obligatorio), Nombres (obligatorios), Apellidos (obligatorios), E-mail (no obligatorios), Lugar de nacimiento (obligatorio), Fecha de nacimiento (obligatorio), Nacionalidad, Dirección (Calle principal y Calle secundaria), Número de casa, Barrio, Teléfono, Vive con, Médico, Escuela anterior (En caso de haberlo), N° Hermanos, sexo (Masculino/Femenino), **CARACTERÍSTICA DEL HOGAR** Fallecidos (Se especifica si alguno de los miembros principales de la familia ha fallecido), Relaciones interpersonales (Específica si sus relaciones interpersonales son buenas, muy buenas o regulares), **VIVIENDA** Tipo de vivienda (Describe si la vivienda es Propia, Arrendada, Familiar u Otra), Número de cuartos, Número de baños, Servicios básicos (Describe si la vivienda tiene los servicios básicos), **DATOS DEL PADRE DE FAMILIA, CABECERA DE FACTURAS** Nombres y Apellidos, CI/RUC, teléfono, Sector, estado (Preinscrito/Inscrito) , fecha inscripción.

Procesos:

- La secretaria del departamento de admisión ingresará los datos de un aspirante.
- La secretaria del departamento de admisión debe ingresar los datos de los padres de familia. (Requerimiento 24).
 - En el caso de que el padre de familia ya este registrado en el sistema se procederá a la búsqueda. (Requerimiento 26).

- La secretaria del departamento de admisión debe solicitar a la aplicación que se realice la creación de un estudiante.
 - El sistema creará al estudiante con un estado de preinscrito.
- La aplicación generará un pedido de cobro de ficha de inscripción.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Estudiante guardado en disco.

Requerimiento 28. Modificar Estudiante Inscrito

Introducción: El sistema permitirá modificar los datos de un estudiante.

Entradas: Número de cédula (No obligatoria) o Número de partida de nacimiento (no obligatorio), Nombres (obligatorios), Apellidos (obligatorios), E-mail (no obligatorios), Lugar de nacimiento (obligatorio), Fecha de nacimiento (obligatorio), Nacionalidad, Dirección (Calle principal y Calle secundaria), Número de casa, Barrio, Teléfono, Vive con, Médico, Escuela anterior (En caso de haberlo), N° Hermanos, sexo (Masculino/Femenino), **CARACTERÍSTICA DEL HOGAR** Fallecidos (Se especifica si alguno de los miembros principales de la familia ha fallecido), Relaciones interpersonales (Específica si sus relaciones interpersonales son buenas, muy buenas o regulares), **VIVIENDA** Tipo de vivienda (Describe si la vivienda es Propia, Arrendada, Familiar u Otra), Número de cuartos, Número de baños, Servicios básicos (Describe si la vivienda tiene los servicios básicos), **DATOS DEL PADRE DE FAMILIA, CABECERA DE FACTURAS** Nombres y Apellidos, CI/RUC, teléfono, Sector.

Procesos:

- La secretaria del departamento de admisión deberá buscar a un estudiante. (Requerimiento 29).
- La secretaria del departamento de admisión debe seleccionar a un estudiante.
- La secretaria del departamento de admisión deberá registrar los nuevos datos del estudiante.
 - Si es necesario también se actualizarán los datos del padre de familia. (Requerimiento 25).
- La secretaria del departamento de admisión solicita a la aplicación que se realice los cambios correspondientes a dicho estudiante.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Estudiante actualizado.

Requerimiento 29. Búsqueda de los estudiantes

Introducción: El sistema permitirá la búsqueda de los estudiantes.

Entradas: Código del estudiante, Nombre del estudiante.

Procesos:

- El usuario solicitará al sistema buscar el estudiante.
 - Los filtros para la búsqueda serán el código del estudiante o el nombre.
- El sistema desplegará el estudiante que sea igual o similar a lo ingresado.

Salida: Estudiante.

Reporte 3: Hoja de datos informativos

Introducción: El sistema debe mostrar un reporte de los datos informativos concernientes a cada estudiante matriculado en la Escuela SION.

Interesados: Secretaria Académica, Secretaria del DOBE.

Entradas: Código de estudiante, Nombre del estudiante.

Procesos: El reporte debe permitir un filtrado por código estudiante, nombre de estudiante.

Salida: Reporte con los datos informativos del estudiante.

3.1.1.3.4.1.9 Matriculación

Requerimiento 30. Proceso de Matriculación

Introducción: El sistema permitirá matricular a los estudiantes.

Entradas: Fecha de matriculación, Año lectivo, Estudiante, Curso, Carnet de seguro, Estado (pre matriculado/matriculado), Foto, **SERVICIOS** Seguro médico, **PRESENTACIÓN DE DOCUMENTOS** Convenio de matrícula, Pase de año, Certificado refrendado otra Institución (certificado de Terminación primaria), Partida de nacimiento, Requerimiento Fotos, Certificado médico, Certificado visual, Certificado auditivo, Certificado de vacunas, Certificado del pastor, Observaciones **DATOS DEL REPRESENTANTE** Cédula, Nombre, Apellido, Dirección, Teléfono, Sector.

Proceso:

- El padre de Familia solicita a la secretaria, se efectuó el proceso de matrícula.
- La secretaria del departamento de admisión debe buscar al estudiante. (Requerimiento 29).

- Si el estado del estudiante no es el de un inscrito, el resultado de la búsqueda será nula.
- La secretaria del departamento de admisión debe seleccionar al estudiante.
- La secretaria del departamento de admisión deberá registrar los documentos presentados por el padre de familia.
- La secretaria registrará los datos del representante.
 - El sistema permitirá copiar los datos de los padres en el caso de que el representante sea uno de los padres.
- La secretaria deberá ingresar el curso y paralelo.
- En el caso de ser necesario la secretaria deberá actualizar los datos del estudiante. (Requerimiento 28).
- La secretaria deberá registrar en el sistema los servicios estudiantiles que solicita el padre.(Requerimiento 31)
 - En el caso del servicio de seguro médico, el estudiante que ya posea el seguro médico de la empresa EMY, la secretaria solamente deberá registrar el código del carnet del seguro en la aplicación.
- La secretaria guardará la matrícula.
 - La matrícula se generará con estado de pre matrícula.
- La aplicación generará un pedido de cobro de matrícula y un pedido de cobro por los servicios de seguro médico y carnetización.

Salida: Matrícula realizada.

Requerimiento 31. Crear solicitud de servicio

Introducción: El sistema permitirá crear una solicitud de los servicios mensuales que brinda la escuela, para un estudiante matriculado en el año lectivo en curso.

Entradas: Número de matrícula, servicio mensual, fecha de inicio de la solicitud, fecha de fin de la solicitud, fecha de la creación de la solicitud, porcentaje de descuento, solicitud (Si/No).

Procesos:

- La secretaria de admisión buscará la matrícula de un estudiante por su nombre o código.
- La secretaria de admisión buscará el servicio mensual a solicitar (Requerimiento 64)
- La secretaria de admisión ingresará los datos correspondientes a la solicitud del servicio y solicitará a la aplicación que se realice la creación de dicha solicitud.
- El sistema comprobará que el estudiante no posea solicitudes activas del mismo servicio, dándolo de alta.
- El sistema inicializa por defecto la fecha de creación de la solicitud con la fecha actual del sistema operativo y las fechas de inicio y fin de la solicitud con las fechas de inicio y fin del período académico.
- El sistema comunicará el éxito o fracaso de su solicitud.

Salida: Solicitud de servicio guardada.

Requerimiento 32. Modificar Solicitud de servicio

Introducción: El sistema permitirá modificar los datos de una solicitud de servicio.

Entradas: Número de matrícula, servicio mensual, fecha de inicio de la solicitud, fecha de fin de la solicitud, fecha de la creación de la solicitud, porcentaje de descuento, solicitud (Si/No).

Procesos:

- La secretaria de admisión ordenará al sistema buscar las solicitudes de servicio que posee un estudiante mediante su nombre o código.
- La secretaria de admisión actualizará los datos de una solicitud de servicio.
 - La secretaria académica podrá desactivar la solicitud del servicio al colocar el campo servicio en no y deberá modificar la fecha de fin de la solicitud.
- La secretaria de admisión solicita a la aplicación que se guarden los cambios correspondientes a dicha solicitud.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Solicitud de Servicio actualizada.

Requerimiento 33. Actualizar Presentación de Documentos

Introducción: El sistema permitirá registrar la presentación de documentos de un estudiante matriculado.

Entradas: Convenio de matrícula, Pase de año, Certificado refrendado otra Institución (certificado de Terminación primaria), Partida de nacimiento, Requerimiento Fotos, Certificado médico, Certificado visual, Certificado auditivo, Certificado de vacunas, Certificado del pastor.

Proceso:

- El representante se acerca a la unidad de admisión y presenta los documentos faltantes.
- La secretaria de admisión debe buscar los estudiantes matriculados.
- La secretaria de admisión debe registrar en el sistema la presentación de los documentos.

Salida: Registro de presentación de documentos actualizado.

Reporte 4: Registro de matrícula

Introducción: El sistema debe mostrar un reporte concerniente al registro de matrícula de un estudiante del año lectivo en curso.

Interesados: Secretaria Académica.

Entradas: Código del estudiante, Nombre del estudiante, código del período académico.

Procesos: El reporte debe permitir un filtrado por código estudiante, nombre del estudiante, período académico.

Salida: Reporte con los datos concernientes a la matrícula de un estudiante de un año lectivo.

Reporte 5: Listado simple de estudiantes matriculados

Introducción: El sistema debe mostrar una lista simple de los estudiantes matriculados por curso concernientes a un año lectivo.

Interesados: Secretaria Académica.

Entrada: Año lectivo.

Procesos: El reporte debe permitir un filtrado por Año lectivo y curso.

Salida: Reporte simple de estudiantes matriculados.

Reporte 6: Lista completa de estudiantes matriculados por curso

Introducción: El sistema debe mostrar un reporte completo concerniente a nombre del estudiante, dirección, teléfono, representante, código de matrícula del estudiante perteneciente a un año lectivo.

Interesados: Secretaria Académica.

Entradas: Nombre del curso y Nombre del paralelo, período académico.

Procesos: El reporte debe permitir un filtrado por curso, paralelo y período académico.

Salida: Lista completa de estudiantes matriculados por curso y paralelo.

Reporte 7: Lista de los estudiantes que tengan seguro médico

Introducción: El sistema debe mostrar una lista de los estudiantes matriculados que hayan solicitado el servicio de seguro médico. Los campos a mostrarse son: curso del estudiante, nombre del estudiante, dirección, teléfono, representante, código de matrícula.

Interesados: Secretaria Académica.

Entradas: Curso y Paralelo, Año lectivo.

Procesos: El reporte debe permitir un filtrado por curso, año lectivo.

Salida: Lista completa de estudiantes que utilizan el servicio de seguro médico.

Reporte 8: Lista de estudiantes matriculados por curso y año lectivo.

Introducción: El sistema debe mostrar una lista de los estudiantes matriculados por curso y por año lectivo. Los campos a mostrarse son: año lectivo, curso y paralelo del estudiante, código de matrícula, Nombre del estudiante.

Interesados: Secretaria Académica.

Entradas: Curso y Paralelo, Año lectivo.

Procesos: El reporte debe permitir un filtrado por curso, año lectivo.

Salida: Lista completa de estudiantes matriculados por curso y año lectivo.

Reporte 9: Lista de estudiantes matriculados por año lectivo.

Introducción: El sistema debe mostrar una lista total de los estudiantes matriculados por año lectivo. Los campos a mostrarse son: año lectivo, código de matrícula, nombre del estudiante.

Interesados: Secretaria Académica.

Entrada: Año lectivo.

Procesos: El reporte debe permitir un filtrado por año lectivo.

Salida: Lista completa de todos los estudiantes matriculados en un determinado año lectivo.

Reporte 10: Listado de estudiantes con un detalle de presentación de documentos

Introducción: El sistema debe mostrar un listado de los estudiantes matriculados en el año lectivo en curso. Los campos a mostrarse son: año lectivo, curso y paralelo, código de matrícula, nombre del estudiante, documentos presentados.

Interesados: Secretaria Académica.

Entradas: Año lectivo, Curso.

Procesos: El reporte debe permitir un filtrado por año lectivo, curso y paralelo.

Salida: Lista completa por curso de todos los estudiantes con sus documentos presentados por año lectivo.

3.1.1.3.4.1.10 Evaluación Estudiantes

Requerimiento 34. Abrir Período de Notas

Introducción: El sistema podrá abrir el período para el ingreso de notas especificando el parcial.

Entradas: Registro de Notas (Abierto/cerrado), Parcial (Primero, Segundo, Tercero, etc.).

Proceso:

- La secretaria académica solicitará al sistema abrir el período de registro de notas.
- El sistema abrirá el período para el registro de notas.

Salida: Período de registro de notas abierto.

Requerimiento 35. Cerrar Período de notas

Introducción: El sistema permitirá cerrar un período de registro de notas.

Entradas: Período de Notas (Abierto/cerrado).

Procesos:

- La secretaria académica solicitará al sistema cerrar el período de registro de notas.
- El sistema cerrará el período para el registro de notas.

Salida: Período de registro de notas cerrado.

Requerimiento 36. Ingreso de notas mensuales

Introducción: El sistema debe permitir el registro de notas.

Entradas: Código de Matrícula, Fecha, Parcial, Nota, Observación.

Proceso:

- El sistema debe desplegar las materias que dicta el docente.
- El docente debe seleccionar alguna materia.

- El sistema debe desplegar el listado de estudiantes que tomen esa materia.
- El sistema dará la opción de registrar las notas.
- El docente solicita guardar el registro de notas.

Salida: Registro de Notas realizado.

Requerimiento 37. Corrección de Notas

Introducción: El sistema permitirá realizar la corrección de notas.

Entradas: Código de matrícula, Curso, Materia, Parcial, Nota, Observación,

Proceso:

- La directora solicita la corrección de notas especificando su curso paralelo, materia y estudiante al cual debe hacer la corrección.
- La secretaria académica deberá buscar al estudiante.
- La secretaria deberá definir el curso, la materia, el parcial y la nota a ser modificada
- La secretaria académica deberá especificar las razones de las correcciones.
- La secretaria académica deberá ingresar la nota.
- La secretaria académica solicitará al sistema guardar la corrección de la nota.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Notas Actualizadas.

Reporte 11: Reporte de notas por parcial para nursery, pre kínder y primero

Introducción: El sistema debe mostrar un reporte de las notas (pertenecientes a un parcial) de los estudiantes matriculados en los niveles de nursery, pre kínder y primero.

Interesados: Secretaria Académica.

Entradas: Nombre Curso, código o nombre del estudiante, período académico.

Procesos: El reporte debe permitir un filtrado por curso, código o nombre del estudiante.

Salida: Reporte de los estudiantes o estudiante de un determinado curso, con su respectiva nota, en un determinado año lectivo.

Reporte 12: Reporte de notas por parcial para segundo en adelante.

Introducción: El sistema debe mostrar un reporte de las notas (pertenecientes a un parcial) de los estudiantes matriculados en los niveles de segundo a décimo curso.

Interesados: Secretaria Académica

Entradas: Nombre curso, código o nombre de estudiante, período académico.

Procesos: El reporte debe permitir un filtrado por curso, código o nombre del estudiante.

Salida: Reporte de los estudiantes o estudiante de un determinado curso, con su respectiva nota, en un determinado año lectivo.

Reporte 13: Reportes quimestrales para nursery, pre kínder y primero

Introducción: El sistema debe mostrar un reporte quimestral de las notas de los estudiantes matriculados en los niveles de nursery, pre kínder y primero.

Interesados: Secretaria Académica.

Entradas: Nombre del curso.

Procesos: El reporte debe permitir un filtrado por curso.

Salida: Reporte de los estudiantes de un determinado curso, con las notas de los parciales, en un determinado año lectivo.

Reporte 14: Reportes quimestrales para estudiante de segundo en adelante

Introducción: El sistema debe mostrar un reporte quimestral de las notas de los estudiantes matriculados en los niveles de segundo a décimo curso.

Interesados: Secretaria Académica.

Entradas: Nombre del curso.

Procesos: El reporte debe permitir un filtrado por curso.

Salida: Reporte de los estudiantes de un determinado curso, con las notas de los parciales y su promedio, en un determinado año lectivo.

Reporte 15: Reportes detallado de las calificaciones por año lectivo, curso y materia.

Introducción: El sistema mostrará un reporte anual en base a las notas quimestrales de los estudiantes matriculados, en los niveles de segundo a décimo curso.

Interesados: Secretaria Académica.

Entradas: Nombre del curso y Paralelo, Año lectivo.

Procesos: El reporte debe permitir un filtrado por año lectivo, curso y paralelo.

Salida: Reporte calificaciones de los estudiantes de un determinado curso y paralelo dentro de un año lectivo.

Reporte 16: Registro de estudiantes promovidos (8-10)

Introducción: El sistema debe mostrar el registro de los estudiantes promovidos por curso en un determinado año lectivo. Los campos a mostrarse son: año lectivo, curso y paralelo, nombre del estudiante.

Interesados: Secretaria Académica.

Entradas: Año lectivo, Curso.

Procesos: El reporte debe permitir un filtrado por año lectivo, curso y paralelo.

Salida: Lista completa de todos los estudiantes promovidos en un determinado año lectivo.

3.1.1.3.4.1.11 Ficha de Seguimiento de un Estudiante

Requerimiento 38. Notificación de estudiantes al DOBE.

Introducción: El sistema dará la posibilidad de que el docente notifique a un estudiante al Departamento de Orientación y Bienestar Estudiantil.

Entradas: Notificar, Observación.

Proceso:

- El docente buscará al estudiante que desee notificar al DOBE. (Requerimiento 29).
- El sistema dará la posibilidad de ofrecer un método para que el docente pueda notificar.

Salida: Notificación al Departamento de Orientación y Bienestar Estudiantil.

Requerimiento 39. Actualizar la Ficha de Seguimiento de un estudiante matriculado.

Introducción: El sistema podrá llevar la bitácora de las actividades psicológicas realizadas a un estudiante.

Entradas: Fecha, Actividad, Observación.

Proceso:

- La Psicóloga del departamento de orientación verifica en el sistema los estudiantes notificados.
- La Psicóloga del departamento de orientación realiza una reunión con los padres y con el estudiante.
- La Psicóloga del departamento de orientación comunica al sistema que se atendido la notificación.
- La Psicóloga del departamento de orientación deberá registrar los datos de la ficha de seguimiento.
- La Psicóloga del departamento de orientación solicitará al sistema guardar la ficha de seguimiento.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Ficha de seguimiento de un estudiante actualizada.

Reporte 17: Ficha acumulativa de seguimiento del DOBE.

Introducción: El sistema debe mostrar los datos correspondientes a la ficha de seguimiento de un estudiante.

Interesados: Psicóloga del Departamento de Orientación y Bienestar Estudiantil.

Entradas: Código estudiante, Nombre del estudiante.

Procesos: El reporte debe permitir un filtrado por código de estudiante, nombre del estudiante.

Salida: Ficha de seguimiento de un estudiante registrado.

Requerimiento 40. Registrar Cumplimiento de Contenido

Introducción: El sistema permitirá al docente registrar el cumplimiento de los objetivos con respecto a una materia.

Entradas: Materia, Evaluación.

Proceso:

- El sistema permitirá que cada docente que pertenece a un curso y paralelo registre el cumplimiento de los objetivos de una materia.

Salida: Cumplimiento de la malla curricular actualizada.

Reporte 18: Reporte de cumplimiento de la malla curricular por materia, curso paralelo.

Introducción: El sistema debe mostrar los datos correspondientes al cumplimiento de la malla curricular perteneciente a un curso.

Interesados: Secretaria Académica.

Entradas: Curso, Paralelo, Materia, Docente, evaluación.

Procesos: El reporte debe permitir un filtrado por curso, paralelo, materia.

Salida: Reporte de cumplimiento de malla curricular por materia, curso, paralelo.

3.1.1.3.4.1.12. Asistencia Docentes

Requerimiento 41. Registrar asistencia docentes

Introducción: El sistema llevara una bitácora de la asistencia de los docentes.

Entradas: Datos de los docentes (códigos), fecha, Detalle de asistencia, Observación.

Proceso:

- El inspector verificará la asistencia de los docentes.

- El inspector registrará la asistencia de los docentes para una determinada fecha.
- El inspector solicitará al sistema guardar la asistencia.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Registro de asistencia de docentes actualizado.

Reporte 19: Hoja de asistencias de los docentes dado un rango de fechas

Introducción: El sistema debe mostrar un listado de la asistencia mensual de los docentes dentro de un rango de fechas. Los campos a mostrarse son: nombre del docente, fecha, asistencia.

Interesados: Inspector, Colector.

Entrada: Rango de fechas.

Procesos: El reporte debe permitir un filtrado por un rango de fechas.

Salida: Lista completa de todos los docentes con sus respectivas asistencias dentro de un rango de fechas.

3.1.1.3.4.1.13. Asistencia Estudiantes

Requerimiento 42. Registrar asistencia estudiantes

Introducción: El sistema permitirá registrar la asistencia de los estudiantes.

Entradas: Datos de los estudiantes (número de matrícula), fecha, Detalle de asistencia, Observación.

Proceso:

- El docente verificará la asistencia de los estudiantes.
- El docente registrará la asistencia de los estudiantes para una determinada fecha en el sistema.
- El docente solicitará al sistema guardar la asistencia.

- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Registro de asistencia de los estudiantes actualizados.

Reporte 20: Hoja de asistencia de los estudiantes por curso dado un rango de fechas

Introducción: El sistema debe mostrar un listado de la asistencia de los estudiantes de un determinado curso y paralelo, período académico, dentro de un rango de fechas. Los campos a mostrarse son: año lectivo, curso y paralelo, nombre de estudiante, fecha, asistencia.

Interesado: Inspector.

Entradas: Período académico, Curso, Rango de fechas.

Procesos: El reporte debe permitir un filtrado por un rango de fechas, curso y paralelo.

Salida: Lista completa de todos los estudiantes con sus respectivas asistencias, en un curso para determinado año lectivo y rango de fechas.

Requerimiento 43. Justificación de faltas

Introducción: El sistema permitirá justificar una falta.

Entrada: Datos del estudiante o docente (códigos), fecha, Detalle de asistencia, Observación.

Procesos:

- El inspector recibirá una petición de justificación.
- El inspector deberá buscar las faltas del estudiante o docente.
- La aplicación detallará las faltas que posee un estudiante o docente.
- El inspector justificará la falta.

Salida: Justificación de faltas.

3.1.1.3.4.1.14. Comunicados

Requerimiento 44. Comunicar eventos.

Introducción: El sistema dará la posibilidad de emitir por e-mail comunicados a los padres de familia. Dichos comunicados pueden ser enviados por la secretaria académica.

Entradas: Comunicado, Email del padre.

Proceso:

- La secretaria académica ingresará el comunicado y dará la orden de enviar los correos.

Salida: Notificación.

3.1.1.3.4.2. Módulo de Administración Logística y Financiera

3.1.1.3.4.2.1 Inventarios

Requerimiento 45. Registrar Adquisición (artículo)

Introducción: El sistema permitirá registrar una adquisición (artículo).

Entradas: Nombre, código físico, Descripción, Valor, Fecha de adquisición, Estado del artículo.

Proceso:

- El colector deberá registrar los datos de los artículos que ingresan al inventario.
- El colector deberá solicitar a la aplicación, que se realice el registro de una adquisición con los datos ya ingresados.
 - El artículo se iniciará con un estado disponible.
- El sistema notificará sobre el éxito o fracaso de su solicitud.

Salida: Adquisición guardada en disco.

Requerimiento 46. Modificar Inventario

Introducción: El sistema permitirá modificar los artículos que se encuentran en el inventario.

Entradas: Nombre, código físico, Descripción, Valor, Fecha de adquisición, Estado del artículo.

Proceso:

- El colector deberá buscar una adquisición en el sistema. (Requerimiento 47).
- El colector deberá seleccionar el artículo a modificar.
- El sistema desplegará el artículo con sus datos actuales.
- El colector deberá modificar los datos que quiera actualizar.
 - El colector podrá cambiar el estado de un artículo no disponible a disponible.
 - El usuario también podrá dar de baja a un artículo colocando su estado ha dado de baja.
- El colector deberá solicitar a la aplicación, que se realice la actualización.
- El sistema notificará sobre el éxito o fracaso de su solicitud.

Salida: Inventario actualizado.

Requerimiento 47. Búsqueda de Inventario

Introducción: El sistema permitirá la búsqueda de un artículo del inventario.

Entradas: Nombre del artículo, código físico.

Proceso:

- El usuario proveerá al sistema, el nombre del artículo.

- El usuario solicitará al sistema realizar la búsqueda de un artículo en el inventario.
- El sistema desplegará las adquisiciones que sean iguales o similares a lo ingresado.
 - Si el nombre ingresado por el colector es igual a “*.*” se desplegarán todas las adquisiciones (artículos).

Salida: Artículo o listado de artículos.

Reporte 21: Inventario de la Escuela.

Introducción: El sistema debe generar un listado de los artículos existentes en el inventario de la Escuela SION.

Interesados: colector.

Entrada:

Procesos: Se muestra un listado de los artículos no dados de baja.

Salida: Imprimir un listado de los artículos por nombre y el total del valor de los artículos que posee la escuela.

3.1.1.3.4.2.2. Transporte

Requerimiento 48. Crear Ruta de Transporte

Introducción: El sistema permitirá crear una ruta de transporte.

Entradas: Descripción, Destino, Número de bus, Nombre de la cooperativa, Número de placa, Año lectivo.

Procesos:

- La secretaria académica ingresará los datos de una ruta.
- El sistema comprobará que no existan buses con más de una ruta en el mismo período académico.

- La secretaria académica solicita a la aplicación que se realice la creación de una ruta de transporte.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Ruta guardada en el disco.

Requerimiento 49. Modificar Ruta de Transporte

Introducción: El sistema permitirá modificar los datos de una ruta.

Entradas: Descripción, Destino, Número de bus, Nombre de la cooperativa, Número de placa, Año lectivo.

Procesos:

- La secretaria académica deberá buscar una ruta en el sistema. (Requerimiento 50).
- La secretaria académica actualizará los datos de una ruta.
 - La secretaria académica buscará la ruta creada, mediante el sistema, para luego proceder a actualizar sus datos.
 - El sistema comprobará que no existan buses con más de una ruta en el mismo período académico.
- La secretaria académica solicita a la aplicación que se realice los cambios correspondientes a dicha ruta.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Ruta actualizada.

Requerimiento 50. Búsqueda de Rutas de transporte

Introducción: El sistema permitirá la búsqueda de las rutas de transporte pertenecientes a un año lectivo.

Entradas: Descripción, Año lectivo.

Procesos:

- El usuario solicitará al sistema realizar la búsqueda de la ruta de transporte.
 - Los filtros de la búsqueda serán la descripción de la ruta.
- El sistema desplegará la ruta de transporte que sea igual o similar a lo ingresado o si es igual a “*.*” se desplegarán todas las líneas registradas.

Salida: Ruta de transporte registrada o listado de las rutas de transporte registradas.

Requerimiento 51. Asignación de estudiantes a cada ruta de transporte

Introducción: El sistema tendrá la opción de asignar las rutas de transporte a cada uno de los estudiantes matriculados que han solicitado el servicio de transporte.

Entradas: Número de matrícula de un estudiante, Ruta.

Proceso:

- La secretaria académica buscará a los estudiantes que hayan solicitado servicio de transporte.
- La aplicación desplegará todos los estudiantes matriculados que tengan el servicio de transporte.
- La secretaria académica asignará por medio de la aplicación, una ruta de transporte a un estudiante.
- La secretaria académica solicitará al sistema guardar la asignación de rutas.

- El sistema informará sobre el éxito o fracaso de la solicitud.

Salidas: Asignación de rutas de transporte.

Reporte 22: Lista de los estudiantes que utilizan el servicio de transporte por bus

Introducción: El sistema debe mostrar una lista de los estudiantes matriculados que hayan solicitado el servicio de transporte. Los campos a mostrarse son: curso del estudiante, nombre del estudiante, dirección, teléfono, representante, código de matrícula, número de bus.

Interesados: Secretaria Académica.

Entrada: Número de bus.

Procesos: El reporte debe permitir un filtrado por número de bus.

Salida: Lista completa de estudiantes que utilizan el servicio de transporte.

Reporte 23: Lista de los estudiantes que utilizan el servicio de transporte por curso

Introducción: El sistema debe mostrar una lista de los estudiantes matriculados que hayan solicitado el servicio de transporte. Los campos a mostrarse son: curso del estudiante, nombre del estudiante, dirección, teléfono, representante, código de matrícula, número de bus.

Interesados: Secretaria Académica.

Entrada: Período académico, curso.

Procesos: El reporte debe permitir un filtrado por curso.

Salida: Lista completa de estudiantes que utilizan el servicio de transporte por curso.

3.1.1.3.4.2.3 Productos

Requerimiento 52. Crear Productos

Introducción: El sistema permitirá crear un producto.

Entradas: Nombre, Descripción, Marca, Valor unitario, Stock producto, Stock mínimo, IVA (Si al producto se le carga el IVA si/no), tipo producto.

Proceso:

- El administrador de almacén deberá ingresar los datos del producto al sistema.
 - El sistema permitirá seleccionar si el producto tiene IVA (si/no).
- El administrador de almacén deberá solicitar a la aplicación, que se realice la creación de un producto con los datos ya ingresados.
 - Por defecto el producto se inicializará con tipo producto.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Producto guardado en el disco.

Requerimiento 53. Modificar Producto

Introducción: El sistema permitirá modificar los datos de un producto.

Entradas: Nombre, Descripción, Marca, Valor unitario, Stock producto, Stock mínimo, IVA (Si al producto se le carga el IVA si/no).

Proceso:

- El administrador de almacén deberá buscar un producto en el sistema. (Requerimiento 54).
- El administrador de almacén deberá seleccionar el producto a modificar.
- El sistema desplegará el producto con sus datos actuales.
- El administrador de almacén deberá modificar los datos que quiera actualizar.

- El sistema permitirá seleccionar si el producto tiene IVA (si/no).
- El administrador de almacén deberá solicitar a la aplicación, que se realice la actualización.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Producto Actualizado.

Requerimiento 54. Búsqueda de Productos

Introducción: El sistema permitirá la búsqueda de productos.

Entrada: Nombre del producto.

Proceso:

- El usuario proveerá al sistema, el nombre del producto.
- El usuario solicitará al sistema buscar el producto.
- El sistema desplegará los productos que sean iguales o similares a lo ingresado.
 - Si el nombre ingresado por el administrador de almacén es igual a “*.*” se desplegarán todos los productos.

Salida: Producto o listado de productos.

3.1.1.3.4.2.4 Proveedores

Requerimiento 55. Crear Proveedor

Introducción: El sistema permitirá crear un proveedor.

Entradas: Nombre proveedor, E-mail, celular, Teléfono, Dirección, Crédito (detalla si el proveedor da la posibilidad de dar a consignación su producto para que esta sea pagada en un plazo posterior).

Proceso:

- El administrador de almacén deberá ingresar los datos del proveedor al sistema

- El sistema validará el ingreso del e-mail
- El sistema deberá dar la posibilidad de seleccionar si el proveedor ofrece o no crédito (si/no).
- El administrador de almacén deberá solicitar a la aplicación, que se realice la creación de un nuevo proveedor con los datos ya ingresados.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Proveedor guardado en el disco.

Requerimiento 56. Modificar Proveedor

Introducción: El sistema permitirá modificar los datos de un proveedor.

Entradas: Nombre proveedor, E-mail, celular, Teléfono, Dirección, Crédito (detalla si el proveedor da la posibilidad de dar a consignación su producto para que esta sea pagada en un plazo posterior).

Proceso:

- El administrador de almacén deberá buscar un proveedor en el sistema. (Requerimiento 52).
- El administrador de almacén deberá seleccionar el proveedor a modificar.
- El administrador de almacén desplegará el proveedor con sus datos actuales.
- El administrador de almacén deberá modificar los datos que quiera actualizar.
 - El sistema validará el ingreso del e-mail.
 - El sistema deberá dar la posibilidad de seleccionar si el proveedor ofrece o no crédito (si/no).
- El administrador de almacén deberá solicitar a la aplicación, que se

realice la actualización.

- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Proveedor Actualizado.

Requerimiento 57. Búsqueda de Proveedor

Introducción: El sistema permitirá la búsqueda de productos.

Entrada: Nombre del proveedor.

Proceso:

- El usuario proveerá al sistema el nombre del proveedor.
- El usuario solicitará al sistema buscar el proveedor.
- El sistema desplegará los proveedores que sean iguales o similares a lo ingresado.
 - Si el nombre de los proveedores ingresados por el usuario es igual a “*.*” se desplegarán todos los proveedores.

Salida: Proveedor o listado de proveedores.

3.1.1.3.4.2.5 Pedido

Requerimiento 58. Registrar Pedido

Introducción: El sistema permitirá que el administrador de almacén pueda generar un pedido.

Entradas: Código del estudiante, Producto, Valor total, Fecha, período académico, estado, detalle, fecha de pago, I.V.A., descuento.

Proceso:

- El administrador de almacén debe buscar en el sistema el estudiante para el cual se solicita el producto. (Requerimiento 29).
- El administrador de almacén debe seleccionar al estudiante.

- El usuario debe seleccionar cada uno de los productos que desee comprar.
- El administrador de almacén solicitará al sistema guardar el pedido.
- El sistema notificará el éxito o fracaso de la solicitud.

Salida: Pedido guardado.

3.1.1.3.4.2.6 Entrada de Producto

Requerimiento 59. Registro Ingreso de Productos

Introducción: El sistema permitirá registrar el ingreso de productos.

Entradas: Nombre del producto, Nombre del proveedor, Unidades compradas, valor unitario del producto, I.V.A., Descuento, Costo, Fecha.

Proceso:

- El administrador de almacén podrá registrar el ingreso de productos cuando lo requiera.
- El administrador de almacén debe solicitar al sistema, guardar el ingreso del producto.
 - El sistema debe actualizar el stock.
- El sistema notificará el éxito o fracaso de la solicitud.

Salida: Registro de ingreso de productos.

Reporte 24: Listado de productos filtrado por stock y producto

Introducción: El sistema debe generar un listado de los productos que posee el almacén.

Interesados: Administrador de almacén.

Entrada: Nombre producto

Procesos: El listado de productos se puede filtrar nombre de producto, stock de producto.

Salida: Imprimir un listado de los productos por nombre y stock existente en el almacén.

Reporte 25: Reporte diario de las entradas por producto

Introducción: El sistema debe generar un listado de las entradas de un producto del almacén.

Interesados: Administrador del almacén.

Entradas: Nombre de producto, fecha de inicio y fecha de fin.

Procesos: El listado de productos se puede filtrar nombre de producto, rango de fechas.

Salida: Imprimir un listado de las entradas de un producto dado su nombre, además el total del valor del producto adquirido dentro de un período de tiempo.

Reporte 26: Reporte diario de las salidas por producto

Introducción: El sistema debe generar un listado de las salidas de productos del almacén.

Interesados: Administrador del almacén.

Entradas: Nombre de producto, fecha de inicio, fecha de fin.

Procesos: El listado de productos se puede filtrar nombre de producto, rango de fechas.

Salida: Imprimir un listado de la salida de productos por nombre y el total del valor de los productos expedidos dentro de un período de tiempo.

Reporte 27: Reporte de entradas y salidas de un producto dentro de un período de tiempo.

Introducción: El sistema debe generar un listado de las entradas y salidas de los productos del almacén.

Interesados: Administrador del almacén.

Entradas: Nombre del producto, fecha de inicio, fecha de fin.

Procesos: El listado de productos se puede filtrar por nombre del producto, rango de fechas.

Salida: Imprimir un listado de las entradas, salidas de los productos por nombre, dentro de un período de tiempo.

3.1.1.3.4.2.7 Presupuesto

Requerimiento 60. Registro Presupuesto Anual

Introducción: El sistema dará la opción de registrar una lista de ingresos y egresos detallando cada uno con los valores mensuales que se espera.

Entradas: Rubro, Tipo (ingreso/egreso), Enero, Febrero, Marzo, Abril, Mayo, Junio, Julio, Agosto, Septiembre, Octubre, Noviembre, Diciembre, Año.

Proceso:

- El contador detalla los ingresos mensuales que se espera percibir durante el año.
 - La proyección de los ingresos por cada mes será realizado mediante la aplicación de la siguiente fórmula: número de niños matriculados el anterior año lectivo por valor del servicio solicitado.
- El contador detalla los gastos mensuales en los cuales se espera incurrir durante el año.

Salida: Presupuesto.

Reporte 28: Reporte Presupuestario

Introducción: El sistema debe generar un reporte concerniente al cálculo anticipado de los ingresos y los gastos anuales.

Interesados: Contador.

Entrada: Año lectivo.

Procesos: Se puede filtrar por Año.

Salida: Imprimir un reporte del cálculo anticipado del presupuesto que posee la Escuela SION, con un detalle de los ingresos y los gastos.

Requerimiento 61. Reajuste Mensual del Presupuesto

Introducción: El sistema dará la opción de actualizar una lista de ingresos y egresos detallados cada uno con los valores reales mensuales generados por la Escuela SION.

Entrada: Rubro, Tipo (ingreso/egreso), Enero, Febrero, Marzo, Abril, Mayo, Junio, Julio, Agosto, Septiembre, Octubre, Noviembre, Diciembre, Año.

Proceso:

- El sistema debe dar la posibilidad de modificar los valores del presupuesto.
- El contador solicita al sistema que guarde las modificaciones.
- El sistema debe notificar el éxito o fracaso de la ejecución.

Salida: Presupuesto actualizado.

3.1.1.3.4.2.8 Facturación Servicios

Requerimiento 62. Crear Servicios

Introducción: El sistema permitirá crear un Servicio.

Entradas: Nombre Servicio, Descripción, Valor del servicio, IVA (SI/NO), tipo de servicio.

Proceso:

- El Colector deberá ingresar los datos del Servicio al sistema.
- El Colector deberá solicitar a la aplicación, que se realice la creación de un Servicio con los datos ya ingresados.

- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Servicio guardado en el disco.

Requerimiento 63. Modificar Servicio

Introducción: El sistema permitirá modificar los datos de un Servicio.

Entradas: Nombre Servicio, Descripción, Valor del servicio, IVA (SI/NO), tipo de servicio.

Proceso:

- El Colector deberá buscar un Servicio en el sistema. (Requerimiento 64).
- El Colector deberá seleccionar el Servicio a modificar.
- El sistema desplegará el Servicio con sus datos actuales.
- El Colector deberá modificar los datos que quiera actualizar.
 - El usuario podrá cambiar el tipo del servicio a dado de baja.
- El Colector deberá solicitar a la aplicación, que se realice la actualización.
- El sistema informará sobre el éxito o fracaso de la solicitud.

Salida: Servicio actualizado.

Requerimiento 64. Búsqueda de Servicios

Introducción: El sistema permitirá la búsqueda de Servicios.

Entradas: Nombre del Servicio.

Proceso:

- El usuario proveerá al sistema, el nombre del Servicio.
- El usuario solicitará al sistema buscar el Servicio.
- El sistema desplegará los Servicios que sean iguales o similares a lo ingresado.

- Si el nombre ingresado por el Colector es igual a “*.*” se desplegarán todos los Servicios.

Salida: Servicio o listado de Servicios.

Requerimiento 65. Cobro Mensual de Servicios

Introducción: El sistema permitirá el cobro mensual de servicios.

Entrada: Código del estudiante.

Proceso:

- El colector deberá buscar a un estudiante.
- El sistema desplegará los datos del estudiante.
- El colector deberá seleccionar el servicio.
- El sistema desplegará el mes por el cual debe cancelar.
- El colector realizará el pago de dicho mes.
- El colector solicitará al sistema que se guarde el cobro.

Salida: Cobro de servicios efectuado.

Reporte 29: Listado de servicios que adeuda un estudiante filtrado por curso.

Introducción: El sistema debe generar un listado de los servicios que adeuda un estudiante, este listado será filtrado por curso.

Interesados: Colector.

Entradas: Nombre del curso, paralelo del curso.

Procesos: El listado de los servicios se filtrara por curso.

Salida: Imprimir un listado de los servicios y meses que adeuda un estudiante de un determinado curso.

3.1.1.3.4.2.9 Facturación Pedidos

Requerimiento 66. Registro de Pago de pedidos

Introducción: El sistema permitirá el pago de pedidos.

Entradas: Código del estudiante o Nombre del estudiante, Fecha, Dirección, Cédula del cliente, Teléfono.

Proceso:

- El colector deberá ingresar el código del estudiante y buscar los pedidos correspondientes al mismo.
- El sistema detallará los pedidos que tiene el estudiante.
- El colector tendrá la opción de registrar el pago de los pedidos.
 - En el caso del pago por la ficha de inscripción el estado del estudiante pasará de preinscrito a inscrito.
 - En el caso del pago por la matrícula el estado del estudiante pasará de pre matriculado a matriculado.

Salida: Factura pagada.

Reporte 30: Recibo por servicios y productos

Introducción: El sistema debe generar un recibo concerniente a los servicios o productos solicitados por el estudiante.

Interesados: Colector.

Entradas: Código de estudiante, Nombre del estudiante.

Procesos: El recibo se puede filtrar por código estudiante, nombre de estudiante.

Salida: Recibo con un detalle de los datos pertenecientes a los servicio o productos solicitados por el estudiante.

Reporte 31: Factura por servicios

Introducción: El sistema debe generar una factura concerniente al cobro de los servicios solicitados por el estudiante.

Interesado: Colector.

Entradas: Código de estudiante, Nombre del estudiante.

Procesos: El recibo se puede filtrar por código de estudiante, nombre del estudiante.

Salida: Imprimir una factura con un detalle de los datos pertenecientes a los servicios y el total a pagar por el estudiante.

Reporte 32: Factura por producto

Introducción: El sistema debe generar una factura concerniente al cobro de los productos pedidos por el estudiante.

Interesado: Colector.

Entradas: Código o Nombre del estudiante.

Procesos: El recibo se puede filtrar por código o nombre del estudiante.

Salida: Imprimir una factura con un detalle de los productos pertenecientes a un pedido y el total a pagar por el estudiante.

3.1.1.3.4.3 Módulo de Seguridad

3.1.1.3.4.3.1. Usuarios

Requerimiento 67. Crear Usuario

Introducción: El sistema permitirá crear un usuario y especificar sus privilegios dentro del sistema.

Entradas: Cédula de usuario, Contraseña, Tipo de usuario.

Procesos:

- El administrador del sistema ingresará los datos de un usuario.

- El sistema comprobará que el usuario no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos usuarios con el mismo número de cédula, contraseña, perfil).
- El administrador del sistema solicita a la aplicación que se realice la creación de un usuario.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Usuario guardado en disco.

Requerimiento 68. Modificar Usuario

Introducción: El sistema permitirá modificar los datos de un usuario.

Entrada: Cédula del usuario.

Procesos:

- El Administrador del sistema actualizará los datos de un usuario.
 - El Administrador del sistema buscará al usuario creado, mediante el sistema, para luego proceder a actualizar sus datos.
- El Administrador del sistema solicita a la aplicación que se realice los cambios correspondientes a dicho usuario.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Usuario actualizado.

Requerimiento 69. Eliminar Usuario

Introducción: El sistema permitirá eliminar un usuario es decir que ya no tendrá acceso al sistema.

Entrada: Cédula del usuario.

Procesos:

- El Administrador del Sistema eliminará un usuario.
 - El Administrador del Sistema buscará al usuario creado, mediante el sistema, para luego proceder a eliminarlo.
- El Administrador del Sistema solicita a la aplicación que se realice la eliminación de un usuario.
- El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.

Salida: Usuario eliminado.

Requerimiento 70. Cambiar contraseña

Introducción: El sistema permitirá que cualquier usuario pueda cambiar de contraseña.

Entrada: Cédula del usuario.

Procesos:

- El usuario deberá solicitar al sistema el cambio de contraseña.
- El usuario debe ingresar su nueva contraseña y su confirmación.
- El usuario debe solicitar a la aplicación que se guarde la contraseña nueva.

Salida: Contraseña modificada

Requerimiento 71. Validar Usuario

Introducción: El sistema validará a los usuarios que ingresan al sistema.

Entrada: cédula del usuario.

Procesos:

- El usuario registrará su cédula, contraseña y tipo de usuario.
- El sistema debe validar el usuario.

- Si el usuario está registrado podrá ingresar en el sistema.

Salida: Usuario Validado.

3.1.1.3.4.4. Parámetros

Requerimiento 72. Actualizar Parámetro

Introducción: El sistema permitirá actualizar los parámetros.

Entrada: IVA, Registro nota, Parcial, E-mail institución, Contraseña correo.

Procesos:

- La secretaria deberá actualizar cada uno de los parámetros.
- La secretaria debe confirmar la actualización.

Salida: Parámetro guardado en el disco.

3.1.1.3.4.5. Suposiciones y dependencias

Suposiciones:

Se asume que los requisitos obtenidos previamente son estables una vez que han sido aprobados por los clientes o involucrados.

Dependencias:

La presente versión del sistema no posee dependencias.

3.1.1.3.4.6. Requisitos de usuarios y tecnológicos

Requisitos de usuario:

Los usuarios serán las personas encargadas del área administrativa y académica de la Escuela SION. Las interfaces deberán ser simples y amigables con el usuario, por lo que con una breve capacitación los profesores y administrativos sean capaces de utilizarla.

Requisitos de tecnológicos:

En vista de que existen múltiples usuarios que necesitan acceder a la información de manera regular, se ha optado por un entorno web. La

aplicación se ejecutará sobre un esquema tres capas, con la interfaz de usuario ejecutándose en un browser y éstos solicitando requerimientos al servidor que cumple su proceso y los datos resultantes se almacenarán en la base de datos. El sistema operativo de los clientes puede ser Windows, Linux. El sistema operativo del servidor es Linux y el Sistema Gestor de Base de Datos es PostgreSQL, estos requisitos fueron fijados por la configuración actual de red y el software disponible por el cliente.

3.1.1.3.4.7. Requisitos de interfaces externas

Interfaces de usuario:

La interfaz de usuario debe ser orientada a formularios Web.

Interfaces Hardware:

El usuario poseerá un teclado y ratón comunes.

Interfaces software:

La interfaz de Gestión Administrativa y Académica tendrán una correlación, ya que se integraran en un mismo sistema.

3.1.1.3.4.8. Requisitos de rendimiento

El tiempo estimado de respuesta del sistema a una solicitud del usuario no debe superar los 5 segundos. El tiempo de respuesta en la generación de los reportes está ligado a la cantidad de datos que existan para dicho reporte.

3.1.1.3.4.9. Requisitos de desarrollo

Una vez aprobados los requerimientos del sistema por parte de los usuarios, no se podrán realizar cambios radicales durante el proceso de desarrollo.

3.1.1.3.4.10. Restricciones de diseño

Ajuste a estándares:

Los estándares de facturación serán de acuerdo a como los defina la ESCUELA SION.

Datos:

El ingreso de los datos será exclusivamente responsabilidad de los usuarios. La aplicación no garantiza la coherencia de los datos ingresados. Cabe recalcar que los datos son vitales para el óptimo funcionamiento de la aplicación.

Seguridad:

La seguridad de los datos se trabajará mediante claves de acceso.

Política de Respaldo:

La aplicación no se responsabilizará por la creación de respaldos, ni el mantenimiento de la base de datos.

Base de Datos:

El Sistema Gestor de Base de Datos será relacional. La aplicación se conectará a la base de datos usando el framework JPA (Java Persistence Api).

Política de Borrado:

Una vez guardado los datos no se podrán borrar, ya que la aplicación mantendrá un registro de todos los años lectivos. El borrado de datos solamente se podrá realizar desde la base de datos con la responsabilidad del Administrador de la base de datos.

3.2 Definición de Actores

Tabla 3. 1 Definición de actores

	Docentes	Secretaria Académica	Inspector	Secretaria del DOBE	Secretaria Admisión	Colector	Administrador del almacén	contador	Administrador del Sistema
MÓDULO DE ADMINISTRACIÓN ACADÉMICA									
1. Currículo									
Materias									
• Creación de Materia		X							
• Modificar Materia		X							
• Búsqueda de las Materias de un determinado grado.	X	X							
Contenido									
• Crear contenido		X							
• Modificar contenido		X							
• Eliminar contenido		X							
• Buscar contenido por Materia	X	X							
Curso									
• Creación de Cursos		X							
• Modificar Curso		X							
• Buscar Curso		X	X						
Períodos									
• Crear Período Académico		X							

• Crear Rango de Notas		X							
• Modificar Rango de Notas		X							
Docentes									
• Crear Docente			X						
• Modificar Docente			X						
• Buscar Docente		X	X						
2. Organización curricular									
• Asignación de aulas a cursos		X							
• Asignación de recursos a aulas		X							
• Asignación de Docentes a una materia, grado, paralelo.		X							
3. Admisión									
Inscripción									
• Creación de Padres de Familia					X				
• Modificar Padres de Familia					X				
• Búsqueda de los Padres de Familia					X				
• Inscribir Estudiante					X				
• Modificar Estudiante Inscrito					X				
• Búsqueda de los Estudiantes Inscritos					X				
Matriculación									
• Matricular Estudiante					X				
• Actualizar Presentación de documentos					X				
• Actualizar Solicitud de Servicios					X				

4. Evaluación									
Período de Notas									
• Abrir Período de Registro de Notas		X							
• Cerrar Período de Registro de Notas		X							
Notas									
• Ingreso de notas mensuales	X								
• Corrección de Notas		X							
• Ficha de Seguimiento				X					
• Notificación de alumnos al DOBE.	X								
• Actualización la Ficha de Seguimiento				X					
• Registrar Cumplimiento de Contenido	X								
5. Asistencia									
• Control asistencia de docentes			X						
• Control asistencia estudiantes	X								
• Justificación de faltas			X						
• 6. Comunicar eventos.		X							
• Crear Noticia		X							
• Modificar Noticia		X							
MÓDULO DE ADMINISTRACIÓN LOGÍSTICA Y FINANCIERA									
Logística									
1. Inventarios									

• Registrar Adquisición						X			
• Modificar Inventario						X			
• Búsqueda Inventario						X			
2. Transporte									
• Crear Ruta de Transporte		X							
• Modificar Ruta de Transporte		X							
• Búsqueda de Rutas de transporte		X							
• Asignación de estudiantes a cada ruta		X							
3. Almacén									
Producto									
• Crear Productos							X		
• Modificar Producto							X		
• Búsqueda de Productos							X		
Proveedor									
Crear Proveedor							X		
• Modificar Proveedor							X		
• Búsqueda de Proveedor							X		
Registrar Pedido									
• Registrar Pedido							X		
Entrada de productos									
• Registro Ingreso de Productos							X		
Financiera									
1. Presupuesto									
• Presupuesto Anual								X	

• Reajuste Mensual del Presupuesto								X	
• 2. Facturación									
Servicios									
• Crear Servicio						X			
• Modificar Servicio						X			
• Búsqueda de servicio						X			
• Cobro mensual de Servicios						X			
Producto									
• Registro de Pago de Pedido						X			
MÓDULO DE SEGURIDAD									
Usuarios									
• Crear Usuario									X
• Modificar Usuario									X
• Eliminar Usuario									X
• Cambiar contraseña	X	X	X	X	X	X	X	X	X
• Validar Usuario	X	X	X	X	X	X	X	X	X
Parámetros									
• Actualizar Parámetros		X							

3.3 Modelo de Casos de Uso y Descripción de casos de uso

3.3.1 Plan de estudios

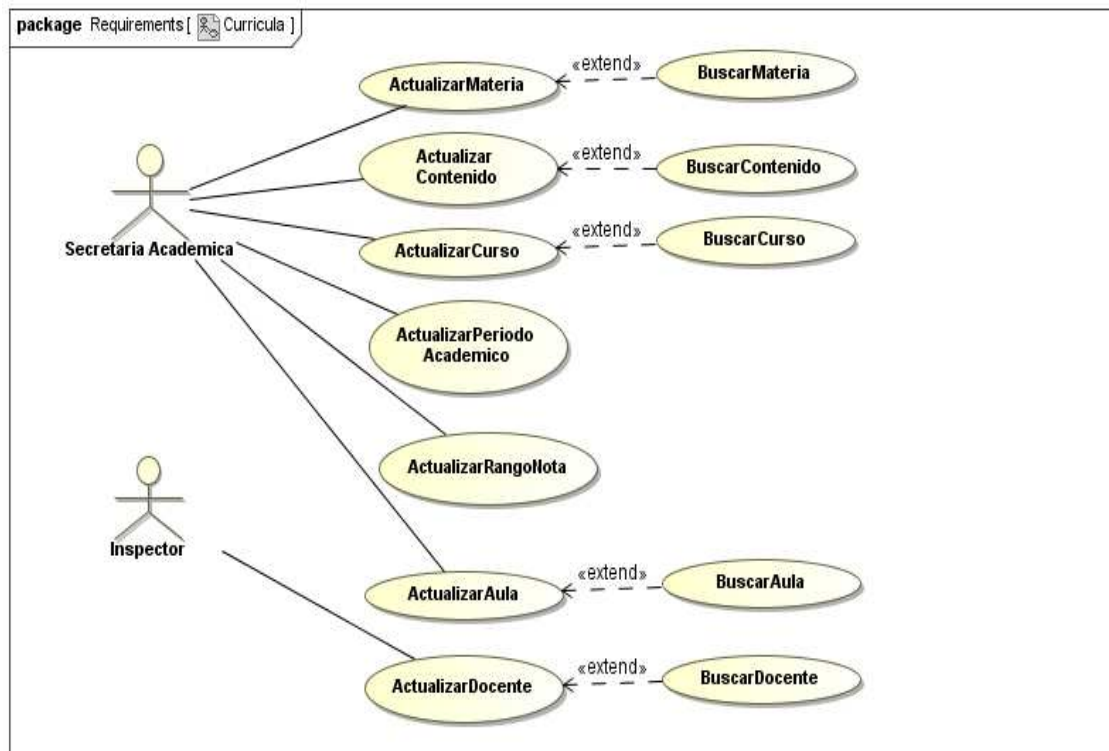


Figura 3. 1 : (Caso de Uso – Currícula)

Tabla 3. 2. : Descripción casos de uso Actualizar Materia (ActualizarMateria)

CASO DE USO:	ActualizarMateria
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear una materia de acuerdo a las disposiciones emitidas por el Ministerio de Educación, los cuales determinarán que materias son las obligatorias para cada curso.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria Académica, la cual debe validarse con su propia cédula de usuario, contraseña, tipo.
6. Datos de Entrada	Nombre de la materia, Curso, Tipo de materia, Objetivo de la materia, Estado de

	materia, Observación.
7. Secuencia	Pasos.
7.1 Crear Materia	<ol style="list-style-type: none"> 1. La secretaria académica ingresará los datos de una materia. 2. La secretaria académica debe solicitar a la aplicación que se realice la creación de una materia. <ul style="list-style-type: none"> ○ El sistema comprobará que la materia no esté creada, dándola de alta. (No pueden existir dos materias con el mismo código o nombre.) 3. El sistema inicializa por defecto las materias con un estado disponible. 4. El sistema notificará el éxito o el fracaso de su solicitud.
7.1.1 Pos condiciones	Materia guardada en el disco.
7.1.2 Excepciones	No se podrá crear una materia sin haber ingresado al menos el nombre de la materia y el curso al que pertenece.
7.2 Modificar Materia	<ol style="list-style-type: none"> 1. La secretaria académica debe buscar la materia, mediante el sistema. 2. La secretaria académica registrará nuevos datos para una materia. <ul style="list-style-type: none"> ○ El sistema permitirá inhabilitar una materia. Cuando una materia es inhabilitada no estará disponible en los siguientes años lectivos. La inhabilitación de las materias lo podrá hacer la secretaria académica previa orden de rectorado. 3. La secretaria académica deberá solicitar a la aplicación que se realice los cambios correspondientes a dicha materia. 4. El sistema comunicará el éxito o el fracaso de la solicitud.
7.2.1 Pos condiciones	Materia actualizada
7.2.2 Excepciones	No se podrá modificar una materia sin haber ingresado su código de materia, nombre y

	curso.
8. Limitaciones	El sistema no podrá eliminar una materia, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 3. : Descripción casos de uso Buscar Materia (BuscarMateria)

CASO DE USO:	BuscarMateria
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá la búsqueda de las materias pertenecientes a un determinado curso.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Nombre del Curso.
7. Secuencia	Pasos.
7.1 Buscar Materia de un determinado curso	<ol style="list-style-type: none"> 1. El usuario solicitará al sistema buscar las materias existentes en un curso. <ul style="list-style-type: none"> ○ Los filtros para la búsqueda serán el nombre curso. 2. El sistema desplegará las materias que correspondan al curso ingresado.
7.1.1 Pos condiciones	Listado de materias por curso.
7.1.2 Excepciones	No se podrá buscar una materia que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 4. : Descripción casos de uso Actualizar Contenido
(ActualizarContenido)

CASO DE USO:	ActualizarContenido
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar cada uno de los contenidos que tiene una materia
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria Académica la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Materia, Contenido, estado del contenido.
7. Secuencia	Pasos.
7.1 Crear contenido	<ol style="list-style-type: none"> 1. La secretaria académica debe buscar las materias activas de un curso. 2. La secretaria académica debe seleccionar la materia a la cual desea agregar contenido. 3. La secretaria académica debe agregar los contenidos que sean necesarios. 4. La secretaria académica solicita a la aplicación que se guarde el contenido. 5. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.
7.1.1 Pos condiciones	Contenido creado.
7.1.2 Excepciones	No se podrá crear un contenido sin haber ingresado al menos el nombre de la materia y el curso al que pertenece.
7.2 Modificar Contenido	<ol style="list-style-type: none"> 1. La secretaria académica debe buscar los contenidos de una materia. 2. La secretaria académica debe registrar nuevos datos para el contenido. 3. La secretaria académica solicita a la aplicación que se guarden los cambios realizados. 4. El sistema notificará el éxito o fracaso de su solicitud.

7.2.1 Pos condiciones	Contenido Actualizado.
7.2.2 Excepciones	No se podrá modificar un contenido sin haber ingresado su código de materia.
7.3 Dar de baja Contenido	<ol style="list-style-type: none"> 1. La secretaria académica debe buscar los contenidos de una materia. 2. La secretaria académica debe dar de baja los contenidos que desea. 3. El sistema notificará el éxito o fracaso de su solicitud.
7.3.1 Pos condiciones	Contenido dado de baja.
7.3.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 5. : Descripción casos de uso Buscar Contenido (BuscarContenido)

CASO DE USO:	BuscarContenido
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá buscar el contenido que posee una materia.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario, contraseña y tipo.
6. Datos de Entrada	Curso, Materia.
7. Secuencia	Pasos.
7.1 Buscar Contenido por Materia	<ol style="list-style-type: none"> 1. El usuario buscará por medio del sistema la materia que se imparte en un curso. 2. El usuario deberá seleccionar la materia. 3. El sistema desplegará todos los contenidos de esa materia.
7.1.1 Pos	Listado de contenidos por materia.

condiciones	
7.1.2 Excepciones	No se podrá buscar un contenido que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 6. : Descripción casos de uso Actualizar Curso (ActualizarCurso)

CASO DE USO:	ActualizarCurso
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear un curso con su respectivo paralelo.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Descripción curso, Descripción paralelo, Estado, tipo de curso.
7. Secuencia	Pasos.
7.1 Crear Curso	<ol style="list-style-type: none"> 1. La secretaria académica debe ingresar los datos de un curso. 2. La secretaria académica solicita a la aplicación que se realice la creación de un curso. <ul style="list-style-type: none"> ○ El sistema comprobará que el curso y el paralelo no estén creados, dándolo de alta. (No pueden haber dos cursos con los paralelos iguales). 3. El sistema inicializa por defecto los cursos con un estado disponible. 4. El sistema comunicará el éxito o fracaso de su solicitud.
7.1.1 Pos condiciones	Cursos guardados en disco.

7.1.2 Excepciones	No se podrá crear un curso sin haber ingresado al menos el nombre del curso y el paralelo existente para dicho curso.
7.2 Modificar Curso	<ol style="list-style-type: none"> 1. La secretaria académica debe buscar los cursos. 2. La secretaria académica debe registrar nuevos datos para el curso. <ul style="list-style-type: none"> o El sistema permitirá inhabilitar un curso con su respectivo paralelo, es decir que este curso y paralelo no estará disponible en el siguiente año lectivo. 3. La secretaria académica solicita a la aplicación que se guarden los cambios correspondientes a dicho curso. 4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.
7.2.1 Pos condiciones	Curso actualizado.
7.2.2 Excepciones	Solo se podrá modificar el estado del curso.
8. Limitaciones	El sistema no podrá eliminar un curso, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 7. : Descripción casos de uso Buscar Curso. (BuscarCurso)

CASO DE USO:	BuscarCurso
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá buscar los cursos.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Nombre curso.

7. Secuencia	Pasos.
7.1 Buscar Curso	<ol style="list-style-type: none"> 1. El usuario deberá solicitar la búsqueda de los cursos al sistema. <ul style="list-style-type: none"> o El usuario deberá especificar el nombre del curso. 2. El sistema académico desplegará un listado de cursos
7.1.1 Pos condiciones	Curso o Listado de cursos.
7.1.2 Excepciones	No se podrá buscar un curso que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 8. : Descripción casos de uso Actualizar Período Académico (ActualizarPeríodoAcadémico)

CASO DE USO:	ActualizarPeríodoAcadémico
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear un período académico.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria Académica. La cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Año lectivo, Fecha de inicio y fin del período académico.
7. Secuencia	Pasos.
7.1 Crear Período Académico	<ol style="list-style-type: none"> 1. La secretaria académica registrará el período académico. 2. La secretaria académica solicita a la aplicación que se guarde el período académico. 3. El sistema comunicará si su solicitud ha

	sido efectuada, o si ha sido fallida.
7.1.1 Pos condiciones	Periodo Guardado.
7.1.2 Excepciones	No se podrá crear un período si la fecha de inicio es mayor a la fecha de fin de período.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 9. : Descripción casos de uso ActualizarRangoNota

CASO DE USO:	ActualizarRangoNota
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear un rango de notas para la aprobación o reprobación de un año lectivo.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria Académica, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Período académico, límite inferior, límite superior del rango de nota, especificación de límite inferior, especificación de límite superior, resultado, puntaje requerido para la aprobación del año lectivo.
7. Secuencia	Pasos.
7.1 Crear Rango Nota	<ol style="list-style-type: none"> 1. La secretaria académica ingresará los datos de un rango de nota. 2. La secretaria académica debe solicitar a la aplicación que se realice la creación de un rango de nota. <ul style="list-style-type: none"> ○ El sistema comprobará que el rango no existe y que sus límites no se intersecan con los de otro rango, dándolo de alta. 4. El sistema notificará el éxito o el fracaso de

	su solicitud.
7.1.1 Pos condiciones	Rango nota guardado en disco.
7.1.2 Excepciones	No se podrán crear rangos con los mismos límites.
7.2 Modificar Rango Nota	<ol style="list-style-type: none"> 1. La secretaria académica debe buscar el rango de nota, mediante el sistema. 2. La secretaria académica registrará nuevos datos para dicho rango de nota. 3. La secretaria académica deberá solicitar a la aplicación que se realice los cambios correspondientes a dicho rango de nota. 4. El sistema comunicará el éxito o el fracaso de la solicitud.
7.2.1 Pos condiciones	Rango nota actualizada
7.2.2 Excepciones	Los límites del rango no se deben intersecar con los de otro rango.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 10. : Descripción casos de uso Actualizar Aula (Actualizar aula)

CASO DE USO:	Actualizar aula
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear un aula.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es Secretaria Académica, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Nombre aula, Capacidad, Observaciones, Estado.

7. Secuencia	Pasos.
7.1 Crear Aula	<p>1. La secretaria académica ingresará los datos del aula.</p> <ul style="list-style-type: none"> o El sistema comprobará que el aula no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos aulas con la misma descripción). <p>2. La secretaria académica solicita a la aplicación que se guarde el aula.</p> <p>3. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.1.1 Pos condiciones	Aula guardada.
7.1.2 Excepciones	No se podrá crear un aula sin haber ingresado al menos el nombre.
7.2 Modificar Aula	<p>1. La secretaria académica buscará el aula en el sistema.</p> <p>2. La secretaria académica actualizará los datos del aula.</p> <p>3. La secretaria académica solicitará a la aplicación que se guarden los cambios.</p> <p>4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.2.1 Pos condiciones	Aula actualizada.
7.2.2 Excepciones	
8. Limitaciones	El sistema no podrá eliminar un aula, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 11. : Descripción casos de uso Buscar Aula (Buscaráula)

CASO DE USO:	Buscaráula
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.

3. Descripción	El sistema permitirá realizar la búsqueda de aulas.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Nombre de aula.
7. Secuencia	Pasos.
7.1 Buscar Aula	<ol style="list-style-type: none"> 1. El usuario deberá solicitar la búsqueda del aula en el sistema. <ul style="list-style-type: none"> ○ El usuario debe digitar el nombre del aula. 2. El sistema desplegará al usuario los datos del aula. <ul style="list-style-type: none"> ○ En el caso de que el usuario digite (*.*) se deberá desplegar todos las aulas activas.
7.1.1 Pos condiciones	Aula o listado de aulas.
7.1.2 Excepciones	No se podrá buscar un aula que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 12. : Descripción casos de uso Actualizar Docente (ActualizarDocente)

CASO DE USO:	ActualizarDocente
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permite introducir información sobre los docentes.
4. Actores	Inspector.

5. Precondiciones	El usuario que ingresará a esta opción es el inspector, el cual debe validarse con su propia cédula y contraseña.
6. Datos de Entrada	Cédula, E-Mail, Nombres, Apellidos, Fecha de nacimiento, estado docente, Dirección (Calle principal y Calle secundaria), Barrio o Sector, fecha de ingreso a la escuela, Teléfono, Celular, Estado civil (casado, soltero, divorciado, otros), relación laboral (contratado, por horas), tipo docente (especial, curso).
7. Secuencia	Pasos.
7.1 Crear Docente	<ol style="list-style-type: none"> 1. El inspector ingresará los datos de un docente. <ul style="list-style-type: none"> ○ El sistema comprobará que el docente no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos docentes con el mismo número de cédula). 2. El inspector solicita a la aplicación que se realice la creación de un docente. <ul style="list-style-type: none"> ○ El sistema crea una cuenta de usuario con el tipo docente. La cédula usuario y contraseña serán el número de su cédula. 3. El sistema inicializa por defecto los docentes con un estado disponible. 4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.
7.1.1 Pos condiciones	Docente guardado en disco.
7.1.2 Excepciones	No se podrá crear un docente sin haber ingresado al menos su número de cédula.
7.2 Modificar Docente	<ol style="list-style-type: none"> 1. El inspector buscará al docente. 2. El inspector actualizará los datos de un docente. <ul style="list-style-type: none"> ○ El inspector buscará al docente creado, mediante el sistema, para luego proceder a actualizar sus datos. ○ El sistema permitirá inhabilitar un docente, es decir que el docente no estará

	<p>disponible en los siguientes años lectivos.</p> <p>3. El inspector solicita a la aplicación que se realice los cambios correspondientes a dicho docente.</p> <p>4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.2.1 Pos condiciones	Docente actualizado.
7.2.2 Excepciones	No se podrá modificar un docente sin haber ingresado su número de cédula.
8. Limitaciones	El sistema no podrá eliminar un docente, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 13. : Descripción casos de uso Buscar docente (BuscarDocente)

CASO DE USO:	BuscarDocente
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá realizar la búsqueda de los docentes.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Cédula del docente.
7. Secuencia	Pasos.
7.1 Buscar Docente	<p>1. El usuario deberá solicitar la búsqueda del docente en el sistema.</p> <ul style="list-style-type: none"> ○ El usuario debe digitar el número de cédula del docente. <p>2. El sistema desplegará al usuario los datos del docente.</p> <ul style="list-style-type: none"> ○ En el caso de que el usuario digite (*.*) se

	deberá desplegar todos los docentes.
7.1.1 Pos condiciones	Docente o listado de docentes.
7.1.2 Excepciones	No se podrá buscar un docente que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.2 Asignación académica

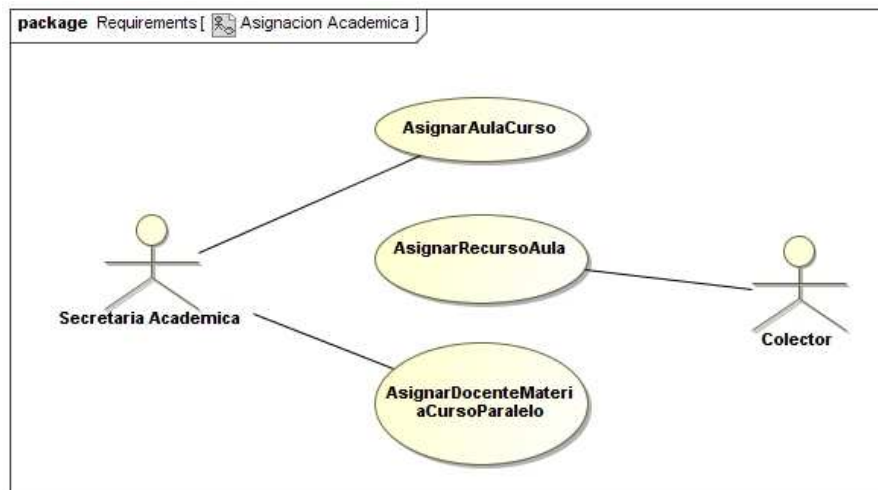


Figura 3. 2: (Caso de Uso – Asignación académica)

Tabla 3. 14. : Descripción casos de uso Asignación de aulas a cursos (AsignarAulaCurso)

CASO DE USO:	AsignarAulaCurso
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá asignar las aulas a los cursos.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria académica. La cual debe validarse con su propia cédula de usuario y

	contraseña.
6. Datos de Entrada	Curso, Aula.
7. Secuencia	Pasos.
7.1 Asignar aulas a cursos	<p>1. La secretaria académica deberá buscar los cursos activos.</p> <ul style="list-style-type: none"> ○ El sistema deberá desplegar un listado de cursos. <p>2. La secretaria académica asignará a un curso un aula.</p> <p>3. La secretaria académica solicitará al sistema guardar la asignación.</p> <p>4. El sistema notificará el éxito o el fracaso de la solicitud.</p>
7.1.1 Pos condiciones	Asignación de aulas a cursos.
7.1.2 Excepciones	Solo se le asignará un aula a un curso con estado disponible.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 15. : Descripción casos de uso Asignación de recursos a aulas (AsignarRecursoAula).

CASO DE USO:	AsignarRecursoAula
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá asignar uno o más recursos a un aula.
4. Actores	Colector.
5. Precondiciones	El usuario que ingresará a esta opción es el colector el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Aula, Recursos (Artículo).

7. Secuencia	Pasos.
7.1 Asignar recursos a aulas	1. El colector debe buscar las aulas. <ul style="list-style-type: none"> ○ El sistema debe desplegar el listado de aulas. 2. El colector seleccionará el aula. 3. El sistema debe buscar los recursos. <ul style="list-style-type: none"> ○ El sistema debe desplegar el listado de recursos. 4. El colector podrá asignar los recursos para el aula. 5. El colector solicita al sistema guardar la asignación.
7.1.1 Pos condiciones	Asignación guardada.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 16. : Descripción casos de uso Asignación de Docentes a materias, cursos y paralelos (AsignarDocenteMateriaCursoParalelo).

CASO DE USO:	Asignación de Docentes a materias, cursos y paralelos
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá asignar un docente a una materia, curso y paralelo.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria académica, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Año lectivo, Curso, Paralelo, Materia, Docente, Fecha.

7. Secuencia	Pasos.
7.1 Asignar Docente a materia, curso y paralelo	<ol style="list-style-type: none"> 1. La secretaria académica especificará el año lectivo. 2. La secretaria académica especificará el curso y paralelo. 3. El sistema desplegará un listado de materias concernientes a dicho curso. 4. La secretaria académica asignará a cada materia el docente. <ul style="list-style-type: none"> ○ El sistema permitirá realizar una búsqueda de los docentes disponibles. 5. La secretaria académica solicita guardar la asignación.
7.1.1 Pos condiciones	Asignación de docentes a una materia, curso, paralelo guardado.
7.1.2 Excepciones	No se podrá asignar un docente no disponible.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.3. Admisión

3.3.3.1 Inscripción

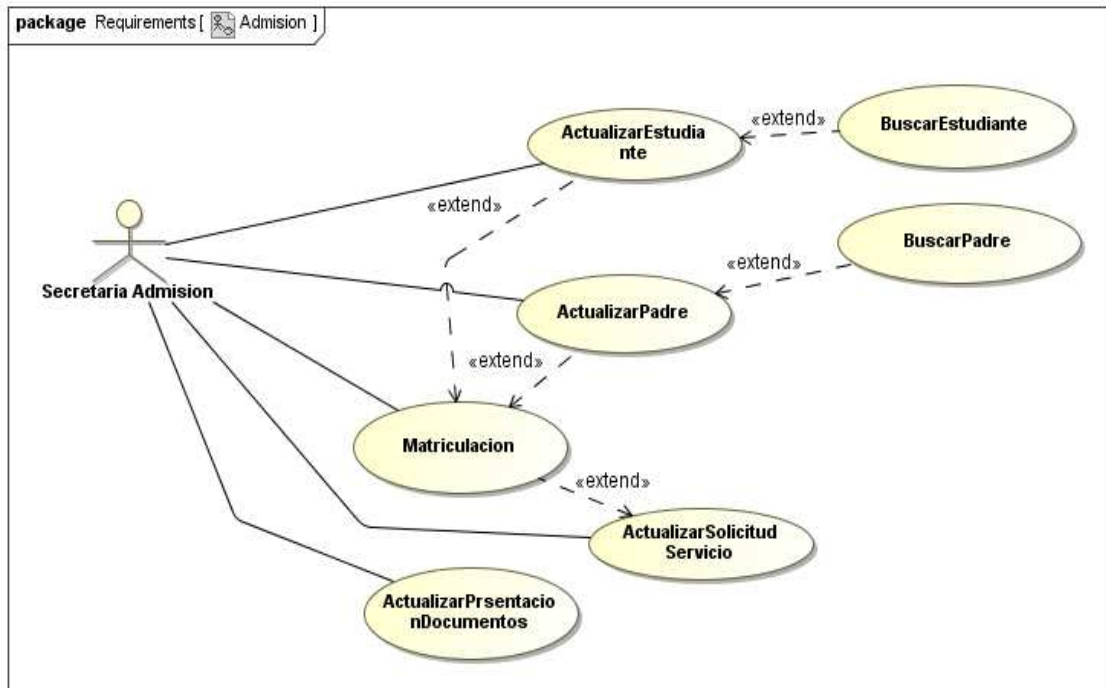


Figura 3. 3: (Caso de Uso – Admisión)

Tabla 3. 17. : Descripción casos de uso Actualizar Padre de Familia

CASO DE USO:	Actualizar Padre de Familia
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permite introducir información sobre los padres de familia que posean hijos inscritos en la Escuela SION o modificar los existentes.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria de admisión la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Cédula del padre de familia, Nombres, Apellidos, Sexo, Dirección (Calle principal y Calle secundaria), E-mail, Teléfono, Celular, Nivel académico (Superior, Secundario, Primario), Instituto (Lugar donde estudio),

	Ocupación, Lugar de trabajo, Puesto de trabajo, Dirección de trabajo, Teléfono del trabajo, Teléfono de emergencia, Situación económica (Muy buena, Buena, Regular), N° Carga familiar, Casa (Propia, Arrendada, Familiares), Tipo de cuenta (Cta. Ahorros o Cta. Corriente), Estado civil (casado, soltero, divorciado, otros), Congregación (Nombre de la iglesia a la que asiste), Denominación (La descripción de la religión), Iglesia hogar o Célula (Especifica si pertenece a una célula de una iglesia), Nombre del líder (Especifica en caso de que el padre de familia pertenezca a una célula de una iglesia), Nacido de nuevo (si aceptó a Dios en su vida, SI/NO), Responsabilidad (Describe un cargo en la Iglesia en caso de tenerlo).
7. Secuencia	Pasos.
7.1 Creación de Padres de Familia	<p>1. La secretaria del departamento de admisión ingresará los datos de un padre de familia.</p> <ul style="list-style-type: none"> ○ El sistema comprobará que el padre no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos padres con el mismo número de cédula). <p>2. La secretaria del departamento de admisión solicita a la aplicación que se realice la creación de un padre.</p> <p>3. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.1.1 Pos condiciones	Padre de familia guardado en disco.
7.1.2 Excepciones	No se podrá crear un padre sin haber ingresado al menos la cédula.
7.2 Modificar Padres de Familia	<p>1. La secretaria de admisión debe buscar el padre de familia.</p> <p>2. La secretaria del departamento de admisión actualizará los datos de un padre de familia.</p> <ul style="list-style-type: none"> ○ La secretaria del departamento de admisión buscará al padre de familia

	<p>creado mediante el sistema, para luego proceder a actualizar sus datos.</p> <p>3. La secretaria del departamento de admisión solicita a la aplicación que se realice los cambios correspondientes a dicho padre.</p> <p>4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.2.1 Pos condiciones	Padre de familia actualizado.
7.2.2 Excepciones	No se podrá modificar un padre sin haber ingresado su cédula.
8. Limitaciones	El sistema no podrá eliminar un padre de familia, solo podrá ser modificado, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 18. : Descripción casos de uso Búsqueda de los Padres de Familia.

CASO DE USO:	Búsqueda de los Padres de Familia
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá la búsqueda de los padres de familia pertenecientes a un determinado estudiante.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Cédula del padre.
7. Secuencia	Pasos.
7.1 Buscar Padres de Familia	<p>1. El usuario proveerá al sistema la cédula del padre.</p> <p>2. El usuario solicitará al sistema buscar el padre de familia.</p> <p>3. El sistema desplegará el padre de familia.</p>

7.1.1 Pos condiciones	Padre.
7.1.2 Excepciones	No se podrá buscar un padre que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 19. : Descripción casos de uso Actualizar estudiante inscrito.

CASO DE USO:	Actualizar estudiante inscrito
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permite introducir información sobre los estudiantes.
4. Actores	Secretaria Admisión.
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria de admisión, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Número de cédula (No obligatoria) o Número de partida de nacimiento (no obligatorio), Nombres (obligatorios), Apellidos (obligatorios), E-mail (no obligatorios), Lugar de nacimiento (obligatorio), Fecha de nacimiento (obligatorio), Nacionalidad, Dirección (Calle principal y Calle secundaria), Número de casa, Barrio, Teléfono, Vive con, Médico, Escuela anterior (En caso de haberlo), N° Hermanos, sexo (Masculino/Femenino), CARACTERÍSTICA DEL HOGAR Fallecidos (Se especifica si alguno de los miembros principales de la familia ha fallecido), Relaciones interpersonales (Especifica si sus relaciones interpersonales son buenas, muy buenas o regulares), VIVIENDA Tipo de vivienda (Describe si la vivienda es Propia, Arrendada,

	Familiar u Otra), Número de cuartos, Número de baños, Servicios básicos (Describe si la vivienda tiene los servicios básicos), DATOS DEL PADRE DE FAMILIA, CABECERA DE FACTURAS Nombres y Apellidos, CI/RUC, teléfono, Sector, estado (Preinscrito/Inscrito) , fecha inscripción.
7. Secuencia	Pasos.
7.1 Inscribir Estudiante	<ol style="list-style-type: none"> 1. La secretaria del departamento de admisión ingresará los datos de un estudiante. 2. La secretaria del departamento de admisión debe ingresar la cédula del padre y la madre de familia. <ul style="list-style-type: none"> ○ El sistema permitirá la búsqueda de los padres de familia. 3. La encargada académica debe solicitar a la aplicación que se realice la creación de un estudiante. <ul style="list-style-type: none"> ○ El sistema creará al estudiante con un estado de preinscrito. 3. El sistema realizará un pedido de cobro de ficha de inscripción. 4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.
7.1.1 Pos condiciones	Estudiante guardado en disco.
7.1.2 Excepciones	
7.2 Modificar Estudiante Inscrito	<ol style="list-style-type: none"> 1. La secretaria del departamento de admisión deberá buscar a un estudiante. 2. La secretaria del departamento de admisión debe seleccionar a un estudiante. 3. La secretaria del departamento de admisión deberá registrar los nuevos datos del estudiante. <ul style="list-style-type: none"> ○ Si es necesario también se actualizarán los datos del padre de familia. 4. La secretaria del departamento de

	<p>admisión solicita a la aplicación que se realice los cambios correspondientes a dicho estudiante.</p> <p>5. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.2.1 Pos condiciones	Estudiante actualizado.
7.2.2 Excepciones	No se podrá modificar un estudiante sin haber sido buscado por su código o nombre de estudiante.
8. Limitaciones	El sistema no podrá eliminar un estudiante inscrito, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 20. : Descripción casos de uso Búsqueda de los estudiantes.

CASO DE USO:	Búsqueda de los estudiantes
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá la búsqueda de los estudiantes.
4. Actores	Usuario.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Código del estudiante, Nombre del estudiante.
7. Secuencia	Pasos.
7.1 Buscar estudiante	<p>1. El usuario solicitará al sistema buscar el estudiante.</p> <ul style="list-style-type: none"> ○ Los filtros para la búsqueda serán el código del estudiante o el nombre. <p>2. El sistema desplegará el estudiante.</p>
7.1.1 Pos condiciones	Estudiante inscrito o listado de estudiantes inscritos.

7.1.2 Excepciones	No se podrá buscar un estudiante que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 21. : Descripción casos de uso Matriculación.

CASO DE USO:	Matriculación
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá matricular a los estudiantes.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria Académica, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Fecha de matriculación, Año lectivo, Estudiante, Curso, Carnet de seguro, Estado (pre matriculado/matriculado), Foto, SERVICIOS Seguro médico, PRESENTACIÓN DE DOCUMENTOS Convenio de matrícula, Pase de año, Certificado refrendado otra Institución (certificado de Terminación primaria), Partida de nacimiento, Requerimiento Fotos, Certificado médico, Certificado visual, Certificado auditivo, Certificado de vacunas, Certificado del pastor, Observaciones DATOS DEL REPRESENTANTE Cédula, Nombre, Apellido, Dirección, Teléfono, Sector.
7. Secuencia	Pasos.
7.1 Matricular Estudiante	1. El padre de Familia solicita a la secretaria de admisión, se efectuó el proceso de matrícula.

	<p>2. La secretaria del departamento de admisión debe buscar al estudiante.</p> <ul style="list-style-type: none"> ○ Si el estado del estudiante no es el de un inscrito, el resultado de la búsqueda será nula. <p>3. La secretaria del departamento de admisión debe seleccionar al estudiante.</p> <p>4. La secretaria del departamento de admisión deberá registrar los documentos presentados por el padre de familia.</p> <p>5. La secretaria registrará los datos del representante.</p> <ul style="list-style-type: none"> ○ El sistema permitirá copiar los datos de los padres en el caso de que el representante sea uno de los padres. <p>6. La secretaria deberá ingresar el curso y paralelo.</p> <p>8. En el caso de ser necesario la secretaria deberá actualizar los datos del estudiante.</p> <p>9. La secretaria deberá registrar en el sistema los servicios estudiantiles que solicita el padre.</p> <ul style="list-style-type: none"> ○ En el caso del servicio de seguro médico, el estudiante que ya posea el seguro médico de la empresa EMY, la secretaria solamente deberá registrar el código del carnet del seguro en la aplicación. <p>10. La secretaria guardará la matrícula.</p> <ul style="list-style-type: none"> ○ La matrícula se generará con estado de pre matrícula. <p>11. El sistema realizará un pedido de cobro de matrícula y un pedido de cobro por los servicios de seguro médico y carnetización.</p>
7.1.1 Pos condiciones	Matrícula realizada.
7.1.2 Excepciones	No se podrá crear un registro de matrícula sin que el estudiante haya sido previamente inscrito.
8. Limitaciones	El sistema no podrá eliminar un registro de matrícula, solo podrá modificarla. Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera

	garantizará la coherencia de los datos ingresados.
--	--

Tabla 3. 22. : Descripción casos de uso Actualizar Solicitud Servicio

CASO DE USO:	ActualizarSolicitudServicio
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear una solicitud de servicio para un estudiante matriculado
4. Actores	Secretaria de Admisión.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria de Admisión, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Número de matrícula, servicio mensual, fecha de inicio de la solicitud, fecha de fin de la solicitud, fecha de la creación de la solicitud, porcentaje de descuento, solicitud (Si/No).
7. Secuencia	Pasos.
7.1 Crear Solicitud Servicio	<ol style="list-style-type: none"> 1. La Secretaria de Admisión ingresará los datos de una solicitud de servicio. 2. La Secretaria de Admisión debe solicitar a la aplicación que se realice la creación de una solicitud de servicio. <ul style="list-style-type: none"> ○ El sistema comprobará que no exista una solicitud activa de dicho servicio para el estudiante seleccionado. 4. El sistema notificará el éxito o el fracaso de su solicitud.
7.1.1 Pos condiciones	Solicitud Servicio guardada en disco.
7.1.2 Excepciones	No se podrán crear nuevas solicitudes del mismo servicio si las anteriores no fueron desactivadas.
7.2 Modificar Solicitud Servicio	<ol style="list-style-type: none"> 1. La Secretaria de Admisión debe buscar la solicitud de servicio de un estudiante, mediante el sistema.

	<p>2. La Secretaria de Admisión registrará nuevos datos para dicha solicitud de servicio.</p> <p>3. Secretaria de Admisión deberá solicitar a la aplicación que se realice los cambios correspondientes a dicha solicitud de servicio.</p> <p>4. El sistema comunicará el éxito o el fracaso de la solicitud.</p>
7.2.1 Pos condiciones	Solicitud de servicio actualizada
7.2.2 Excepciones	No se podrá modificar una solicitud de servicio sin haberla seleccionado previamente
8. Limitaciones	El sistema no podrá eliminar una solicitud de servicio solo podrá poner su solicitud en No, los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 23. : Descripción casos de uso Actualizar Presentación de Documentos

CASO DE USO:	Actualizar Presentación de Documentos
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar la presentación de documentos de un estudiante matriculado.
4. Actores	Secretaria de admisión.
5. Precondiciones	El usuario que ingresará a esta opción es la Secretaria de admisión, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Convenio de matrícula, Pase de año, Certificado refrendado por otra Institución (Certificado de terminación primaria), Partida de nacimiento, Requerimiento Fotos, Certificado médico, Certificado visual, Certificado auditivo, Certificado de vacunas, Certificado del pastor.

7. Secuencia	Pasos.
7.1 Actualizar Presentación de Documentos	<ol style="list-style-type: none"> 1. El representante se acerca a la unidad de admisión y presenta los documentos faltantes. 2. La secretaria de admisión debe buscar los estudiantes matriculados. 3. La secretaria de admisión debe registrar en el sistema la presentación de los documentos.
7.1.1 Pos condiciones	Registro de presentación de documentos actualizado.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.4. Evaluación

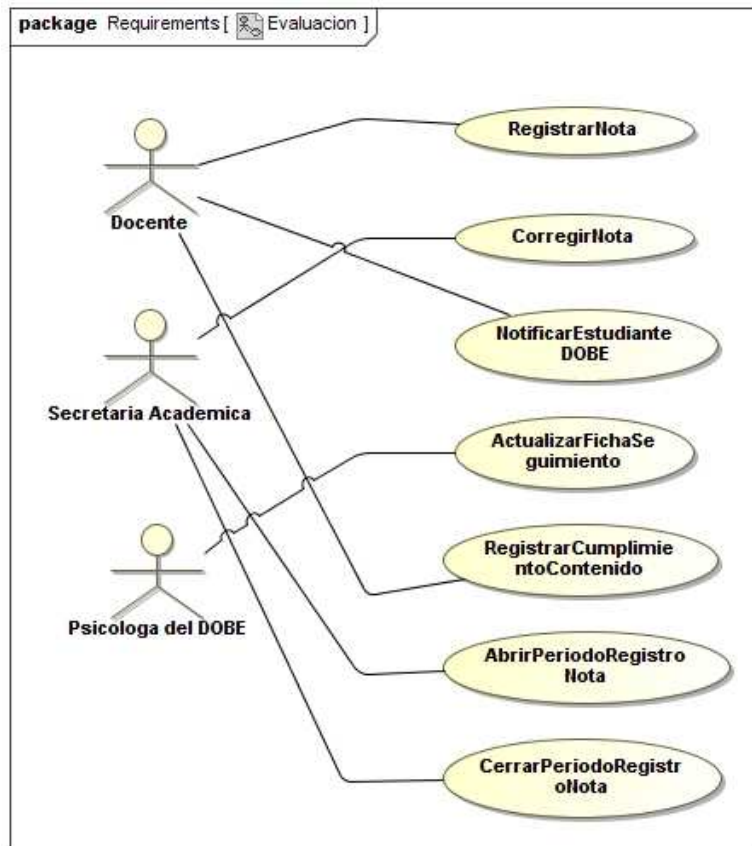


Figura 3. 4: (Caso de Uso – Evaluación)

Tabla 3. 24. : Descripción casos de uso Abrir Período para el registro de Notas(AbrirPeríodoRegistroNota)

CASO DE USO:	AbrirPeríodoRegistroNota
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012
3. Descripción	El sistema dará la posibilidad de abrir el período para registro de notas
4. Actores	Secretaria académica
5. Precondiciones	El usuario debe estar validado con el perfil de secretaria académica.
6. Datos de Entrada	Período registro notas (Abierto/Cerrado), Parcial
7. Secuencia	Pasos
7.1 Abrir Periodo para el registro de Notas	1.-La secretaria académica solicitará al sistema abrir el período de registro de notas para un determinado parcial.
7.1.1 Pos condiciones	Creación de un período de notas.
7.1.2 Excepciones	
8. Limitaciones	

Tabla 3. 25. : Descripción casos de uso CerrarPeríodo para el Registro de Notas. (CerrarPeríodoRegistroNota)

CASO DE USO:	CerrarPeríodoRegistroNota
9. Autor	Alexis Alcocer Paúl Cuichán
10.Fecha Creación	01/08/2012
11.Descripción	El sistema dará la posibilidad de cerrar el período para registro de notas
12.Actores	Secretaria académica
13.Precondiciones	El usuario debe estar validado con el perfil de secretaria académica.
14.Datos de Entrada	Período registro notas (Abierto/Cerrado)

15. Secuencia	Pasos
15.1 Cerrar Periodo para el registro de Notas	1. La secretaria académica solicitará al sistema cerrar el período de registro de notas. 2. El sistema cerrará el período para el registro de notas.
5.1.1 Pos condiciones	Creación de un período de notas.
5.1.2 Excepciones	
16. Limitaciones	

Tabla 3. 26. Descripción casos de uso Registrar Nota (RegistrarNota)

CASO DE USO:	RegistrarNota
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012
3. Descripción	El sistema debe permitir el registro de notas.
4. Actores	Docente
5. Precondiciones	El usuario que ingresará a esta opción es el docente el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Curso, Código de Estudiante, materia, Parcial, Nota, Observación.
7. Secuencia	Pasos
7.1 Registrar nota	1.- El sistema debe desplegar las materias que dicta el docente en un determinado curso. 2.- El docente debe seleccionar alguna de las materias. 3. El sistema debe desplegar el listado de estudiantes. 4. El sistema dará la opción de registrar las notas. 5. El docente solicita guardar el registro de notas.
7.1.1 Pos	Registro de Notas realizado.

condiciones	
7.1.2 Excepciones	
8. Limitaciones	El sistema no podrá eliminar una nota. Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 27. : Descripción casos de uso CorregirNota

CASO DE USO:	CorregirNota
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012
3. Descripción	El sistema permitirá realizar la corrección de notas.
4. Actores	La secretaria académica
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria académica, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Código de Estudiante, Curso, Materia, Parcial, Nota, Observación.
7. Secuencia	Pasos
7.1 Corrección de Notas	<ol style="list-style-type: none"> 1. La directora solicita la corrección de notas especificando su curso paralelo, materia y estudiante al cual debe hacer la corrección. 2. La secretaria académica deberá buscar al estudiante. 3. La secretaria deberá definir el curso, la materia, el parcial. 4. La secretaria académica deberá especificar las razones de las correcciones. 5. La secretaria académica deberá ingresar la nota. 6. La secretaria académica solicitará al sistema guardar la corrección de la nota. 7. El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos	Notas Actualizadas

condiciones	
7.1.2 Excepciones	
8. Limitaciones	El sistema no podrá eliminar una nota. Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 28. : Descripción casos de uso Notificación de estudiantes al DOBE.

CASO DE USO:	Notificación de estudiantes al DOBE.
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema dará la posibilidad de que el docente notifique a un estudiante al Departamento de Orientación y Bienestar Estudiantil.
4. Actores	Docente.
5. Precondiciones	El usuario que ingresará a esta opción es el docente, el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Notificar, Observación.
7. Secuencia	Pasos.
7.1 Notificar estudiante al DOBE	1. El docente buscará al estudiante que desee notificar al DOBE. 2. El sistema dará la posibilidad de ofrecer un método para que el docente pueda notificar.
7.1.1 Pos condiciones	Notificación al Departamento de Orientación y Bienestar Estudiantil.
7.1.2 Excepciones	El profesor solo podrá notificar a los estudiantes pertenecientes a los cursos en los cuales da clase.
8. Limitaciones	El sistema no podrá eliminar una notificación. Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 29. Descripción casos de uso Actualizar la Ficha de Seguimiento de un estudiante matriculado (ActualizarFichaSeguimiento).

CASO DE USO:	ActualizarFichaSeguimiento
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema podrá llevar la bitácora de las actividades psicológicas con respecto a un estudiante.
4. Actores	La Psicóloga del departamento de orientación.
5. Precondiciones	El usuario que ingresará a esta opción es la Psicóloga del DOBE, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Fecha, Actividad, Observación.
7. Secuencia	Pasos.
7.1 Actualizar Ficha de Seguimiento de un estudiante matriculado.	<ol style="list-style-type: none"> 1. La Psicóloga del departamento de orientación verifica en el sistema los estudiantes notificados. 2. La Psicóloga del departamento de orientación realiza una reunión con los padres y con el estudiante. 3. La Psicóloga del departamento de orientación comunica al sistema que se ha atendido la notificación. 4. La Psicóloga del departamento de orientación deberá registrar los datos de la ficha de seguimiento. 5. La Psicóloga del departamento de orientación solicitará al sistema guardar la ficha de seguimiento. 6. El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos condiciones	Ficha de seguimiento de un estudiante actualizada.
7.1.2 Excepciones	No se podrá actualizar una ficha de seguimiento, si el estudiante no fue previamente notificado al DOBE.
8. Limitaciones	El sistema no podrá eliminar una ficha de

	<p>seguimiento.</p> <p>Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.</p>
--	---

Tabla 3. 30. Descripción casos de uso Registrar cumplimiento contenido.
(RegistrarCumplimientoContenido)

CASO DE USO:	RegistrarCumplimientoContenido
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá al docente registrar el cumplimiento de los objetivos con respecto a una materia.
4. Actores	Docente.
5. Precondiciones	El usuario que ingresará a esta opción es el docente, el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Materia, contenido, evaluación.
7. Secuencia	Pasos.
7.1 Registrar cumplimiento de la malla curricular	1. El sistema permitirá que cada docente que pertenece a un curso y paralelo registre el cumplimiento de los contenidos de una materia.
7.1.1 Pos condiciones	Cumplimiento de malla curricular actualizada.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.5 Asistencia

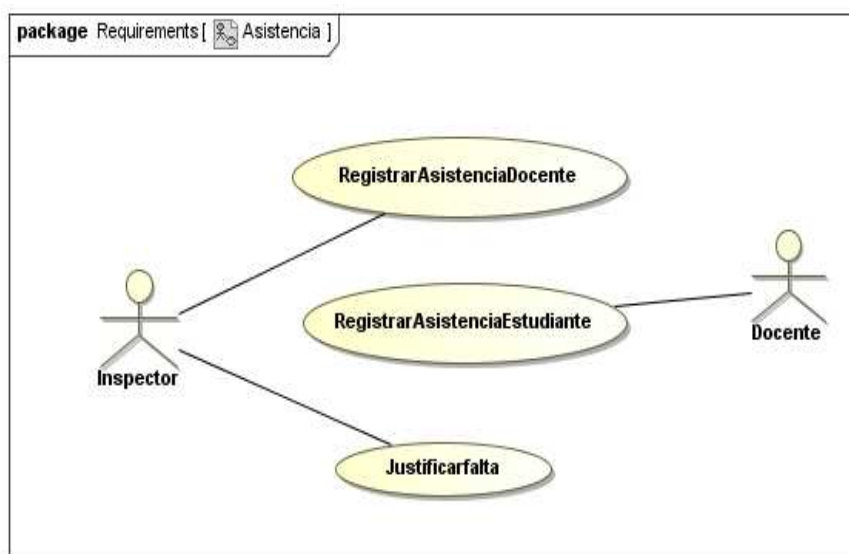


Figura 3. 5: (Caso de Uso – Asistencia)

Tabla 3. 31. : Descripción casos de uso Registrar Asistencia Docente(RegistrarAsistenciaDocente).

CASO DE USO:	RegistrarAsistenciaDocente
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema llevará una bitácora de la asistencia de los docentes.
4. Actores	Inspector.
5. Precondiciones	El usuario que ingresará a esta opción es el inspector, el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Datos del docente (códigos), fecha, Detalle de asistencia, Observación.
7. Secuencia	Pasos.
7.1 Registrar asistencia Docente	<ol style="list-style-type: none"> 1. El inspector verificará la asistencia de los docentes. 2. El inspector registrará la asistencia de los docentes para una determinada fecha. 3. El inspector solicitará al sistema guardar la asistencia. 4. El sistema informará sobre el éxito o fracaso de la solicitud.

7.1.1 Pos condiciones	Registro de Asistencia de Docentes Actualizado.
7.1.2 Excepciones	
8. Limitaciones	El sistema no podrá eliminar una falta solo podrá justificar las faltas. Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 32. : Descripción casos de uso Registrar Asistencia Estudiante (RegistrarAsistenciaEstudiante).

CASO DE USO:	RegistrarAsistenciaEstudiante
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar la asistencia de los estudiantes.
4. Actores	Docente.
5. Precondiciones	El usuario que ingresará a esta opción es el docente, el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Datos del estudiante (códigos), fecha, Detalle de asistencia, Observación.
7. Secuencia	Pasos.
7.1 Registrar asistencia Estudiante	<ol style="list-style-type: none"> 1. El docente verificará la asistencia de los estudiantes. 2. El docente registrará la asistencia de los estudiantes para una determinada fecha en el sistema. 3. El docente solicitará al sistema guardar la asistencia. 4. El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos condiciones	Registro de asistencia de los estudiantes actualizados.
7.1.2 Excepciones	
8. Limitaciones	El sistema no podrá eliminar una falta solo

	podrá justificar las faltas. Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.
--	---

Tabla 3. 33. : Descripción casos de uso Justificar Falta (JustificarFalta)

CASO DE USO:	JustificarFalta
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá justificar una falta.
4. Actores	Inspector.
5. Precondiciones	El usuario que ingresará a esta opción es el inspector, el cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Datos del estudiante o docente (códigos), fecha, Detalle de asistencia, Observación.
7. Secuencia	Pasos.
7.1 Justificar Falta/Atraso	<ol style="list-style-type: none"> 1. El inspector recibirá una petición de justificación. 2. El inspector deberá buscar las faltas del estudiante o docente. 3. La aplicación detallará las faltas que posee un estudiante o docente. 4. El inspector justificará la falta.
7.1.1 Pos condiciones	Justificación de faltas.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.6 Comunicar Eventos

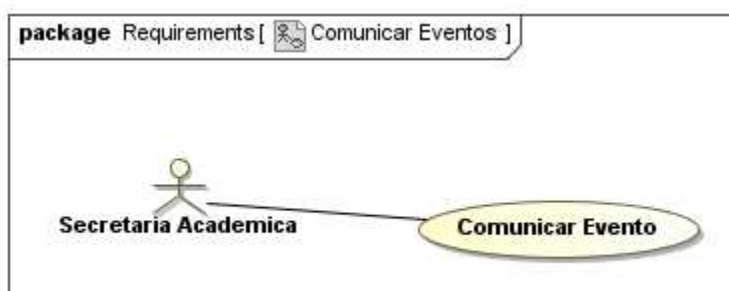


Figura 3. 6: (Caso de Uso – Comunicar Evento)

Tabla 3. 34. Descripción casos de uso Comunicar evento.

CASO DE USO:	ComunicarEvento
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012
3. Descripción	El sistema dará la posibilidad de emitir por e-mail comunicados a los padres de familia. Dichos comunicados pueden ser enviados por la secretaria.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Comunicado, Email del padre.
7. Secuencia	Pasos.
7.1 Comunicar eventos	1. El interesado ingresará el comunicado y dará la orden de enviar los correos.
7.1.1 Pos condiciones	Notificación.
7.1.2 Excepciones	No se podrá enviar un comunicado si el destinatario no posee email.
8. Limitaciones	El sistema no podrá eliminar un comunicado, ni afirmar si el comunicado llego a su destinatario.

3.3.7 Logística

3.3.7.1 Inventario



Figura 3. 7: (Caso de Uso – Inventario)

Tabla 3. 35. : Descripción casos de uso Actualizar Inventario (ActualizarInventario)

CASO DE USO:	ActualizarInventario
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar las entradas de un artículo.
4. Actores	Colector.
5. Precondiciones	El usuario debe estar validado con el perfil de Colector.
6. Datos de Entrada	Nombre, código físico, Descripción, Valor, Fecha de adquisición, Estado de artículo.
7. Secuencia	Pasos.
7.1 Registrar Adquisición	1.- El colector deberá ingresar los datos de los artículos que ingresan al inventario. 2.- El colector deberá solicitar a la aplicación, que se realice el registro de una adquisición con los datos ya ingresados. <ul style="list-style-type: none">o El artículo se iniciará con un estado disponible. 3.- El sistema notificará el éxito o fracaso de su solicitud.

7.1.1 Pos condiciones	Entrada de un nuevo artículo guardado.
7.1.2 Excepciones	Si al menos no se ha detallado el nombre y el código físico de un artículo, el sistema no podrá registrar el ingreso del mismo.
7.2 Modificar Inventario	<p>1.- El colector deberá buscar una adquisición en el sistema.</p> <p>2.- El colector deberá seleccionar el artículo a modificar.</p> <p>3.- El sistema desplegará el artículo con sus datos actuales.</p> <p>4.- El colector deberá modificar los datos que quiera actualizar.</p> <ul style="list-style-type: none"> o El colector podrá cambiar el estado de un artículo no disponible a disponible. o El usuario también podrá dar de baja a un artículo colocando su estado ha dado de baja. <p>5.- El colector deberá solicitar a la aplicación, que se realice la actualización.</p> <p>6.- El sistema notificará sobre el éxito o fracaso de su solicitud.</p>
7.2.1 Pos condiciones	Datos de los artículos del inventario actualizados.
7.2.2 Excepciones	Si al menos no se ha detallado el nombre y el código físico de un artículo el sistema no podrá registrar el ingreso del artículo.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 36. : Descripción casos de uso Buscar Inventario (BuscarInventario)

CASO DE USO:	BuscarInventario
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.

3. Descripción	El sistema permitirá la búsqueda de un artículo del inventario.
4. Actores	Colector.
5. Precondiciones	El usuario debe estar logeado con el perfil de Colector.
6. Datos de Entrada	Nombre del artículo, Código físico.
7. Secuencia	Pasos.
7.1 Buscar Inventario	<ol style="list-style-type: none"> 1.- El usuario proveerá al sistema, el nombre del artículo. 2.- El usuario solicitará al sistema realizar la búsqueda de un artículo en el inventario. 3.- El sistema desplegará las adquisiciones que sean iguales a lo ingresado. <ul style="list-style-type: none"> o Si el nombre ingresado por el colector es igual a “*.*” se desplegarán todas las adquisiciones.
7.1.1 Pos condiciones	Artículo o listado de artículos.
7.1.2 Excepciones	Si no hay ningún artículo con dichos parámetros, la búsqueda será nula.
8. Limitaciones	

3.3.7.2 Transporte

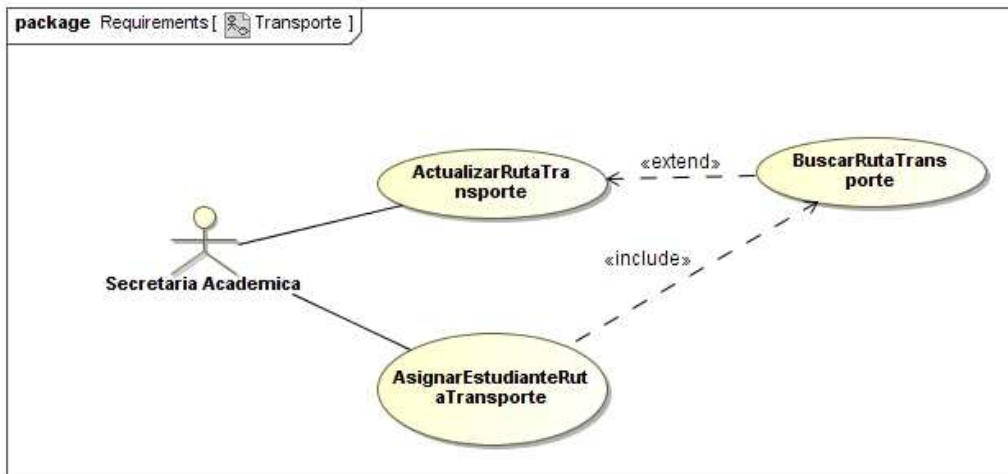


Figura 3. 8: (Caso de Uso –Transporte)

Tabla 3. 37. : Descripción casos de uso Actualizar Ruta de Transporte (ActualizarRutaTransporte).

CASO DE USO:	ActualizarRutaTransporte
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permite introducir información sobre las rutas de transporte.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es secretaria académica, la cual debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Descripción, Destino, Número de bus, Nombre de cooperativa, Número de placa, Año lectivo.
7. Secuencia	Pasos.
7.1 Crear Ruta de Transporte	<ol style="list-style-type: none"> 1. La secretaria académica ingresará los datos de una ruta. 2. El sistema comprobará que no existan dos buses con más de una ruta en el mismo período académico. 3. La secretaria académica solicita a la aplicación que se realice la creación de una ruta de transporte.

	4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.
7.1.1 Pos condiciones	Ruta guardada en disco.
7.1.2 Excepciones	No se podrá crear una ruta sin haber ingresado al menos el número de bus.
7.2 Modificar Ruta de Transporte	<ol style="list-style-type: none"> 1. La secretaria académica deberá buscar una ruta en el sistema. 2. La secretaria académica actualizará los datos de una ruta. <ul style="list-style-type: none"> ○ La secretaria académica buscará la ruta creada, mediante el sistema, para luego proceder a actualizar sus datos. ○ El sistema comprobará que no existan dos buses con más de una ruta en el mismo período académico. 3. La secretaria académica solicita a la aplicación que se realice los cambios correspondientes a dicha ruta. 4. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.
7.2.1 Pos condiciones	Ruta actualizada.
7.2.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 38. : Descripción casos de uso Búsqueda de rutas de transporte (BuscarRutaTransporte)

CASO DE USO:	BuscarRutaTransporte
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá la búsqueda de las rutas de transporte pertenecientes a un año lectivo.
4. Actores	Usuario.

5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Descripción, Año lectivo.
7. Secuencia	Pasos.
7.1 Buscar la ruta de transporte	<p>1. El usuario solicitará al sistema realizar la búsqueda de la ruta de transporte.</p> <ul style="list-style-type: none"> ○ Los filtros de la búsqueda serán la descripción de la ruta, año lectivo. <p>2. El sistema desplegará la ruta de transporte que sea igual a lo ingresado o si lo ingresado por el usuario es igual a “*.*” se desplegarán todas las rutas registradas.</p>
7.1.1 Pos condiciones	Ruta registrada o listado de las rutas de transporte registradas durante un año lectivo.
7.1.2 Excepciones	No se podrá buscar una ruta que no cumpla con los parámetros necesarios para su localización.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 39. : Descripción casos de uso Asignación de ruta de transporte a estudiante. (AsignarEstudianteRutaTransporte)

CASO DE USO:	AsignarEstudianteRutaTransporte
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema tendrá la opción de asignar las rutas de transporte a cada uno de los estudiantes matriculados que han solicitado el servicio de transporte.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario que ingresará a esta opción es la secretaria académica, la cual debe poseer su propia cédula de usuario y contraseña.

6. Datos de Entrada	Estudiante, Ruta.
7. Secuencia	Pasos.
7.1 Asignar estudiante a ruta de transporte	<ol style="list-style-type: none"> 1. La secretaria académica buscará a los estudiantes que hayan solicitado el servicio de transporte. 2. La aplicación desplegará todos los estudiantes matriculados que tengan el servicio de transporte. 3. La secretaria académica asignará por medio de la aplicación, una ruta de transporte a un estudiante. 4. La secretaria académica solicitará al sistema guardar la asignación de rutas. 5. El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos condiciones	Asignación de rutas de transporte.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario, el sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.7.3 Almacén

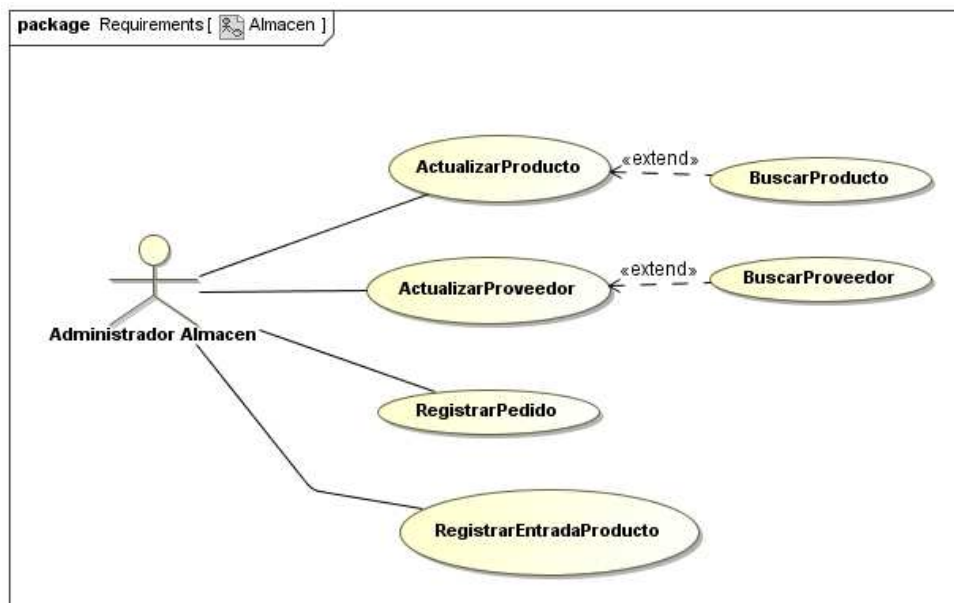


Figura 3. 9: (Caso de Uso –Almacén)

Tabla 3. 40. : Descripción casos de uso Actualizar Producto
(ActualizarProducto)

CASO DE USO:	ActualizarProducto
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permite introducir información sobre los productos.
4. Actores	Administrador de Almacén.
5. Precondiciones	El usuario debe estar logeado con el perfil de Administrador de Almacén.
6. Datos de Entrada	Nombre, Descripción, Marca, Valor unitario, Stock Producto, Stock mínimo, IVA (Si al producto se le carga el IVA si/no), tipo.
7. Secuencia	Pasos.
7.1 Crear Productos	<ol style="list-style-type: none"> 1. El administrador del almacén deberá ingresar los datos del producto al sistema. <ul style="list-style-type: none"> ○ El sistema permitirá seleccionar si el producto tiene IVA (si/no). 2. El administrador del almacén deberá solicitar a la aplicación que se realice la creación de un producto con los datos ya ingresados, por defecto el tipo será producto. 3. El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos condiciones	Producto guardado en disco.
7.1.2 Excepciones	No se efectuara la creación de un producto si al menos no se detalla el nombre del producto.
7.2 Modificar Producto	<ol style="list-style-type: none"> 1. El administrador del almacén deberá buscar un producto en el sistema. 2. El administrador del almacén deberá seleccionar el producto a modificar. 3. El sistema desplegará el producto con sus datos actuales. 4. El administrador del almacén deberá modificar los datos que quiera actualizar. <ul style="list-style-type: none"> ○ El sistema permitirá seleccionar si el producto tiene IVA (si/no).

	<p>5. El administrador del almacén deberá solicitar a la aplicación, que se realice la actualización.</p> <p>6. El sistema informará sobre el éxito o fracaso de la solicitud.</p>
7.2.1 Pos condiciones	Producto Actualizado.
7.2.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 41. : Descripción casos de uso: Buscar Producto

CASO DE USO:	Buscar Producto
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá la búsqueda de productos.
4. Actores	Usuarios del sistema.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Nombre del Producto.
7. Secuencia	Pasos.
7.1 Buscar Producto	<p>1. El usuario proveerá al sistema, el nombre del producto.</p> <p>2. El usuario solicitará al sistema buscar el producto.</p> <p>3. El sistema desplegará los productos que sean iguales a lo ingresado.</p> <p>○ Si el nombre ingresado es igual a “*.*” se desplegarán todos los productos.</p>
7.1.1 Pos	Producto o listado de productos.

condiciones	
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 42. : Descripción casos de uso Actualizar Proveedor (ActualizarProveedor)

CASO DE USO:	ActualizarProveedor
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema deberá dar la posibilidad de crear un proveedor o modificar sus datos.
4. Actores	Administrador del Almacén.
5. Precondiciones	El usuario debe estar logeado con el perfil de Administrador del Almacén.
6. Datos de Entrada	Nombre del proveedor, E-mail, Teléfono, celular, Dirección, Crédito (detalla si el proveedor otorga la posibilidad de dar a consignación su producto para que este sea pagado en un plazo posterior).
7. Secuencia	Pasos.
7.1 Crear Proveedor	<ol style="list-style-type: none"> 1. El administrador de almacén deberá ingresar los datos del proveedor al sistema. <ul style="list-style-type: none"> ○ El sistema validará el ingreso del e-mail. ○ El sistema deberá dar la posibilidad de seleccionar si el proveedor ofrece o no crédito (si/no). 2. El administrador de almacén deberá solicitar a la aplicación, que se realice la creación de un nuevo proveedor con los datos ya ingresados. 3. El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos	Proveedor guardado en disco.

condiciones	
7.1.2 Excepciones	
7.2 Modificar Proveedor	<ol style="list-style-type: none"> 1. El administrador de almacén deberá buscar un proveedor en el sistema. 2. El administrador de almacén deberá seleccionar el proveedor a modificar. 3. El administrador de almacén desplegará el proveedor con sus datos actuales. 4. El administrador de almacén deberá modificar los datos que quiera actualizar. <ul style="list-style-type: none"> o El sistema validará el ingreso del e-mail. o El sistema deberá dar la posibilidad de seleccionar si el proveedor ofrece o no crédito (si/no). 5. El administrador de almacén deberá solicitar a la aplicación, que se realice la actualización. 6. El sistema informará sobre el éxito o fracaso de la solicitud.
7.2.1 Pos condiciones	Proveedor Actualizado.
7.2.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 43. : Descripción casos de uso: Buscar Proveedor

CASO DE USO:	Buscar Proveedor
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá la búsqueda de productos.
4. Actores	Usuarios del Sistema.
5. Precondiciones	El usuario que ingresará a esta opción debe validarse con su propia cédula de usuario y contraseña.
6. Datos de Entrada	Nombre del proveedor.

7. Secuencia	Pasos.
7.1 Buscar Proveedor	<ol style="list-style-type: none"> 1. El usuario proveerá al sistema el nombre del proveedor. 2. El usuario solicitará al sistema buscar el proveedor. 3. El sistema desplegará los proveedores que sean iguales o similares a lo ingresado. <ul style="list-style-type: none"> ○ Si el nombre de los proveedores ingresados por el usuario es igual a “*.*” se desplegarán todos los proveedores.
7.1.1 Pos condiciones	Proveedor o listado de proveedores.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 44. : Descripción casos de uso Registrar Pedido (RegistrarPedido)

CASO DE USO:	RegistrarPedido
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá que el administrador del almacén pueda generar un pedido.
4. Actores	Administrador de Almacén.
5. Precondiciones	El usuario debe estar logeado con el perfil de Administrador de Almacén.
6. Datos de Entrada	Código del estudiante, Producto, Valor total, Fecha, período académico, estado, detalle, fecha de pago, I.V.A., descuento.
7. Secuencia	Pasos.
7.1 Registrar Pedido	<ol style="list-style-type: none"> 1. El administrador de almacén debe buscar en el sistema el estudiante para el que se solicita el producto. 2. El administrador de almacén debe seleccionar al estudiante.

	<p>3. El usuario debe seleccionar cada uno de los productos que desee comprar.</p> <p>4. El administrador de almacén solicitará al sistema guardar el pedido.</p> <p>5. El sistema notificará el éxito o fracaso de la solicitud.</p>
7.1.1 Pos condiciones	Pedido Guardado.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 45. : Descripción casos de uso Registrar Entrada de Productos (RegistrarEntradaProducto)

CASO DE USO:	RegistrarEntradaProducto
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar el ingreso de productos.
4. Actores	Administrador de Almacén.
5. Precondiciones	El usuario debe estar logeado con el perfil de Administrador de Almacén.
6. Datos de Entrada	Nombre del producto, Unidades compradas, valor unitario del producto, Proveedor, I.V.A., Descuento, Costo, Fecha.
7. Secuencia	Pasos.
7.1 Registrar Ingreso de Producto	<p>1. El administrador de almacén podrá registrar el ingreso de productos cuando lo requiera.</p> <p>2. El administrador de almacén debe solicitar al sistema, guardar el ingreso del producto.</p> <ul style="list-style-type: none"> ○ El sistema debe actualizar el stock. <p>3. El sistema notificará el éxito o fracaso de la solicitud.</p>
7.1.1 Pos	Registro de ingreso de productos.

condiciones	
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.8 Financiera

3.3.8.1 Presupuesto

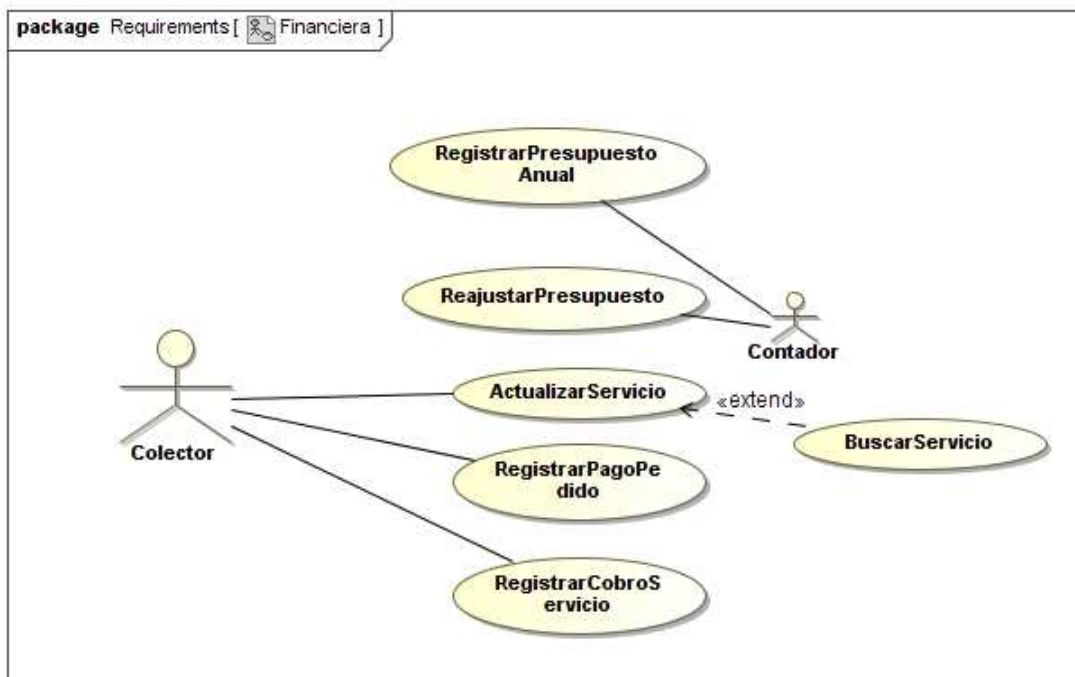


Figura 3. 10: (Caso de Uso – Presupuesto)

Tabla 3. 46. : Descripción casos de uso Registrar Presupuesto Anual (RegistrarPresupuestoAnual)

CASO DE USO:	RegistrarPresupuestoAnual.
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema dará la opción registrar el presupuesto anual y ajustarlo conforme lo requiera el contador.
4. Actores	Contador.

5. Precondiciones	El usuario debe estar logeado con el perfil de Contador.
6. Datos de Entrada	Rubro, Tipo (ingreso/egreso) Enero, Febrero, Marzo, Abril, Mayo, Junio, Julio, Agosto, Septiembre, Octubre, Noviembre, Diciembre, Año.
7. Secuencia	Pasos.
7.1 Registrar Presupuesto Anual	<p>1.- El contador detalla los ingresos mensuales que se espera percibir durante el año.</p> <p>2.- El contador debe detallar los ingresos y gastos mensuales en los cuales se espera incurrir durante el año.</p> <ul style="list-style-type: none"> o El sistema dará la opción de calcular la proyección de los ingresos por cada mes. La fórmula a aplicarse es: número de niños matriculados el anterior año lectivo por valor del servicio solicitado. <p>3.- El contador solicitará guardar el presupuesto.</p>
7.1.1 Pos condiciones	Presupuesto guardado.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 47. : Descripción casos de uso Reajustar Presupuesto.
(ReajustarPresupuesto)

CASO DE USO:	ReajustarPresupuesto.
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema dará la opción de ajustar el presupuesto conforme lo requiera el contador.

4. Actores	Contador.
5. Precondiciones	El usuario debe estar logeado con el perfil de Contador.
6. Datos de Entrada	Rubro, Tipo (ingreso/egreso) Enero, Febrero, Marzo, Abril, Mayo, Junio, Julio, Agosto, Septiembre, Octubre, Noviembre, Diciembre, Año.
7. Secuencia	Pasos.
7.1 Reajustar Presupuesto	<p>1.-El sistema debe dar la posibilidad de modificar los valores del presupuesto.</p> <p>2.-El contador solicita al sistema que guarde las modificaciones.</p> <p>3.-El sistema debe notificar el éxito o fracaso de la ejecución.</p> <p>4.-El sistema comunicará, si su solicitud ha sido ejecutada o en el caso de que el sistema no logre ejecutar su petición también deberá notificarlo.</p>
7.1.1 Pos condiciones	Presupuesto actualizado.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.8.2 Facturación

Tabla 3. 48. : Descripción casos de uso Actualizar Servicio. (ActualizarServicio)

CASO DE USO:	ActualizarServicio
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema dará la opción de actualizar el catálogo de servicios.

4. Actores	Colector.
5. Precondiciones	El usuario debe estar logeado con el perfil de Colector.
6. Datos de Entrada	Nombre Servicio, Descripción, Valor del servicio, IVA (SI/NO), tipo de servicio.
7. Secuencia	Pasos.
7.1 Crear Servicios	<ol style="list-style-type: none"> 1.- El Colector deberá ingresar los datos del Servicio al sistema. 2.- El Colector deberá solicitar a la aplicación, que se realice la creación de un Servicio con los datos ya ingresados. 3.- El sistema informará sobre el éxito o fracaso de la solicitud.
7.1.1 Pos condiciones	Servicio Creado.
7.1.2 Excepciones	El servicio se creará únicamente si todos los datos de entrada están detallados.
7.2 Modificar Servicios	<ol style="list-style-type: none"> 1.- El Colector deberá buscar un Servicio en el sistema. 2.- El Colector deberá seleccionar el Servicio a modificar. 3.- El sistema desplegará el Servicio con sus datos actuales. 4.- El Colector deberá modificar los datos que quiera actualizar. <ul style="list-style-type: none"> o El usuario podrá cambiar el tipo de servicio a dado de baja. 5.- El Colector deberá solicitar a la aplicación, que se realice la actualización. 6.- El sistema informará sobre el éxito o fracaso de la solicitud.
7.2.1 Pos condiciones	Servicio Modificado.
7.2.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de

	ninguna manera garantizará la coherencia de los datos ingresados.
--	---

Tabla 3. 49. : Descripción casos de uso Buscar Servicio (BuscarServicio)

CASO DE USO:	BuscarServicio
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema dará la opción de buscar un servicio.
4. Actores	Colector.
5. Precondiciones	El usuario debe estar logeado con el perfil de Colector.
6. Datos de Entrada	Descripción del Servicio.
7. Secuencia	Pasos.
7.1 Buscar Servicios	<p>1.- El usuario proveerá al sistema, la descripción del Servicio.</p> <p>2.- El usuario solicitará al sistema buscar el Servicio.</p> <p>3.- El sistema desplegará los Servicios que sean iguales a lo ingresado.</p> <ul style="list-style-type: none"> ○ Si el nombre ingresado es igual a “*.*” se desplegarán todos los Servicios.
7.1.1 Pos condiciones	Servicio o listado de Servicios.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 50. : Descripción casos de uso Registro de Pago de Pedidos (RegistrarPagoPedido).

CASO DE USO:	RegistrarPagoPedido
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar el pago de los pedidos que se generan al inscribirse, matricularse o al solicitar en el almacén algún producto.
4. Actores	Colector.
5. Precondiciones	El usuario debe estar logeado con el perfil de Colector.
6. Datos de Entrada	Código del estudiante o Nombre del estudiante, Fecha, Dirección, Cédula del cliente, Teléfono.
7. Secuencia	Pasos.
7.1 Facturar Servicio	<ol style="list-style-type: none"> 1. El colector deberá ingresar el código del estudiante y buscar los pedidos correspondientes al estudiante. 2. El sistema detallará los pedidos que tiene el estudiante. 3. El colector tendrá la opción de registrar el pago de los pedidos. 4. En el caso del pago por la ficha de inscripción, el estado del estudiante pasará de preinscrito a inscrito. 5. En el caso del pago por la matrícula, el estado del estudiante pasará de pre matriculado a matriculado.
7.1.1 Pos condiciones	Pago de pedido registrado.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de

	ninguna manera garantizará la coherencia de los datos ingresados.
--	---

Tabla 3. 51. : Descripción casos de uso Registrar Cobro de Servicio (RegistrarCobroServicio)

CASO DE USO:	RegistrarCobroServicio
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá registrar el cobro de servicio.
4. Actores	Colector.
5. Precondiciones	El usuario debe estar logeado con el perfil de Colector.
6. Datos de Entrada	Código del estudiante.
7. Secuencia	Pasos.
7.1 Facturar Servicio	1.- El contador deberá buscar a un estudiante. 2.- El sistema desplegará los datos del estudiante. 3.- El contador deberá seleccionar el servicio. 4.- El sistema desplegará el mes por el cual debe cancelar. 5.- El colector realizará el pago de dicho mes 6.- El contador solicitará al sistema que se guarde el cobro.
7.1.1 Pos condiciones	Cobro de servicio guardado.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.9 Seguridad

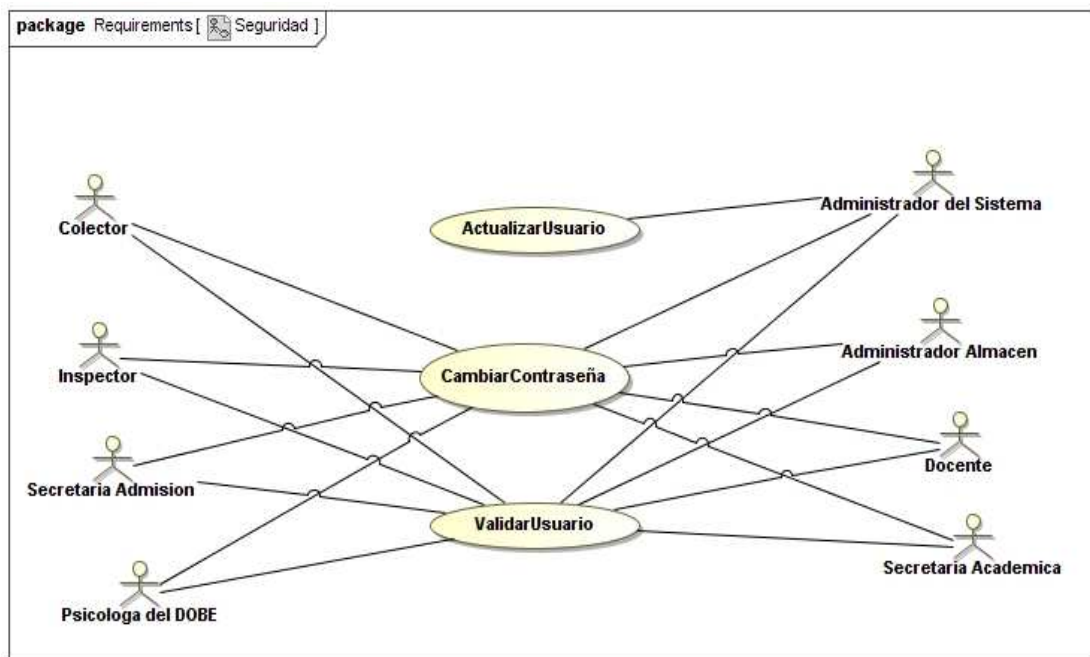


Figura 3. 11: (Caso de Uso - Seguridad)

Tabla 3. 52. : Descripción casos de uso: Actualizar Usuario

CASO DE USO:	Actualizar Usuario
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá crear un usuario y especificar sus privilegios dentro del sistema.
4. Actores	Administrador del Sistema.
5. Precondiciones	El usuario debe estar logeado con el perfil de administrador del sistema.
6. Datos de Entrada	Cédula de usuario, Contraseña, Tipo de usuario.
7. Secuencia	Pasos.
7.1 Crear Usuario	<ol style="list-style-type: none"> 1. El administrador del sistema ingresará los datos de un usuario. <ul style="list-style-type: none"> ○ El sistema comprobará que el usuario no haya sido creado con anterioridad, dándolo de alta. (No pueden existir dos usuarios con el mismo número de

	<p>cédula, contraseña, tipo).</p> <p>2. El Administrador del sistema solicita a la aplicación que se realice la creación de un usuario.</p> <p>3. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.1.1 Pos condiciones	Usuario guardado en disco.
7.1.2 Excepciones	
7.2 Modificar Usuario	<p>1. El Administrador del sistema actualizará los datos de un usuario.</p> <ul style="list-style-type: none"> ○ El Administrador del sistema buscará al usuario creado, mediante el sistema, para luego proceder a actualizar sus datos. <p>2. El Administrador del sistema solicita a la aplicación que se realice los cambios correspondientes a dicho usuario.</p> <p>3. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.2.1 Pos condiciones	Usuario actualizado.
7.2.2 Excepciones	
7.3 Eliminar Usuario	<p>1. El Administrador del sistema eliminará un usuario.</p> <ul style="list-style-type: none"> ○ El Administrador del sistema buscará al usuario creado, mediante el sistema, para luego proceder a eliminarlo. <p>2. El Administrador del sistema solicita a la aplicación que se realice la eliminación de un usuario.</p> <p>3. El sistema comunicará si su solicitud ha sido efectuada, o si ha sido fallida.</p>
7.3.1 Pos condiciones	Usuario Eliminado.
7.3.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de

	ninguna manera garantizará la coherencia de los datos ingresados.
--	---

Tabla 3. 53. : Descripción casos de uso: Cambiar Contraseña

CASO DE USO:	Cambiar Contraseña
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá que cualquier usuario pueda cambiar de contraseña.
4. Actores	Contador, Secretaria, Director, Administrador del sistema, DOBE.
5. Precondiciones	El usuario debe estar logeado con cualquier perfil válido.
6. Datos de Entrada	Nombre de usuario.
7. Secuencia	Pasos.
7.1 Cambiar Contraseña	<ol style="list-style-type: none"> 1. El usuario deberá solicitar al sistema el cambio de contraseña. 2. El usuario debe ingresar su contraseña nueva y su verificación. 3. El usuario debe solicitar a la aplicación que se guarde la contraseña nueva.
7.1.1 Pos condiciones	Contraseña Modificada.
7.1.2 Excepciones	
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

Tabla 3. 54. : Descripción casos de uso: Validar Usuario

CASO DE USO:	Validar Usuario
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.

3. Descripción	El sistema validará a los usuarios que ingresan al sistema.
4. Actores	Usuario PC.
5. Precondiciones	
6. Datos de Entrada	Nombre de usuario.
7. Secuencia	Pasos.
7.1 Validar Usuario	<ol style="list-style-type: none"> 1. El usuario registrará su usuario y contraseña. 2. El sistema debe validar el usuario. <ul style="list-style-type: none"> o Si el usuario está registrado podrá ingresar en el sistema.
7.1.1 Pos condiciones	Usuario Validado.
7.1.2 Excepciones	Si el usuario, la contraseña y el tipo no son correctos, el sistema le impedirá el acceso.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.3.10 Parámetros

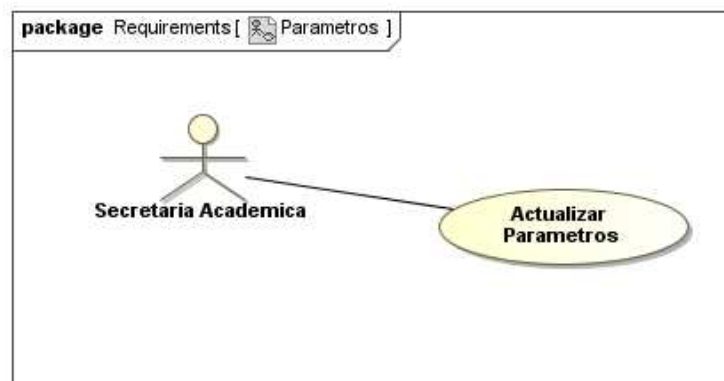


Figura 3. 12: (Caso de Uso - Parámetros)

Tabla 3. 55. : Descripción casos de uso: Actualizar Parámetro

CASO DE USO:	Actualizar Parámetro
1. Autor	Alexis Alcocer Paúl Cuichán
2. Fecha Creación	01/08/2012.
3. Descripción	El sistema permitirá actualizar los parámetros.
4. Actores	Secretaria Académica.
5. Precondiciones	El usuario debe estar logeado con el perfil de Secretaria Académica.
6. Datos de Entrada	IVA, Registro nota, Parcial, E-mail institución, Contraseña correo
7. Secuencia	Pasos.
7.1 Modificar Parámetro	1.- El usuario deberá actualizar cada uno de los parámetros. 2.- El usuario debe confirmar la actualización.
7.1.1 Pos condiciones	Parámetro guardado en disco.
7.1.2 Excepciones	Si el usuario o la contraseña no son correctos, el sistema le impedirá el acceso.
8. Limitaciones	Los datos ingresados serán únicamente responsabilidad del usuario. El sistema de ninguna manera garantizará la coherencia de los datos ingresados.

3.4 Diagrama de Clases

Los diagramas de clases se encuentran en el Tomo II Anexo A

3.5 Diagramas de Secuencia

Los diagramas de secuencia se encuentran en el Tomo II Anexo A

3.6 Diagramas de Estado

Los diagramas de secuencia se encuentran en el Tomo II Anexo A

3.7 Diagrama de Despliegue

Los diagramas de secuencia se encuentran en el Tomo II Anexo A

3.8 Diagramas Navegacionales

Los diagramas navegacionales se encuentran en el Tomo II Anexo A

3.9 Diagramas de Presentación

Los diagramas de presentación se encuentran en el Tomo II Anexo A

3.10 Diagramas de Actividades

Los diagramas de actividades se encuentran en el Tomo II Anexo A

3.11 Diagramas Conceptual de la Base de Datos

El diagrama conceptual de la base de datos se encuentran en el Tomo II Anexo A

3.12 Diagramas Físico de la Base de Datos

El diagrama físico de la base de datos se encuentran en el Tomo II Anexo A

DESARROLLO

4.1 Arquitectura a Implementarse



Figura 4. 1: (Arquitectura a Implementarse)

La siguiente figura muestra como se realiza el flujo de información entre el cliente y el servidor en nuestra aplicación. Primero empieza en la base de datos, luego pasa al procesamiento de la información a través de los componentes y finaliza en la presentación de los resultados en un navegador web.

4.2 Estándares de Implementación

4.2.1 Estándares de codificación

4.2.1.1 Estructura del archivo

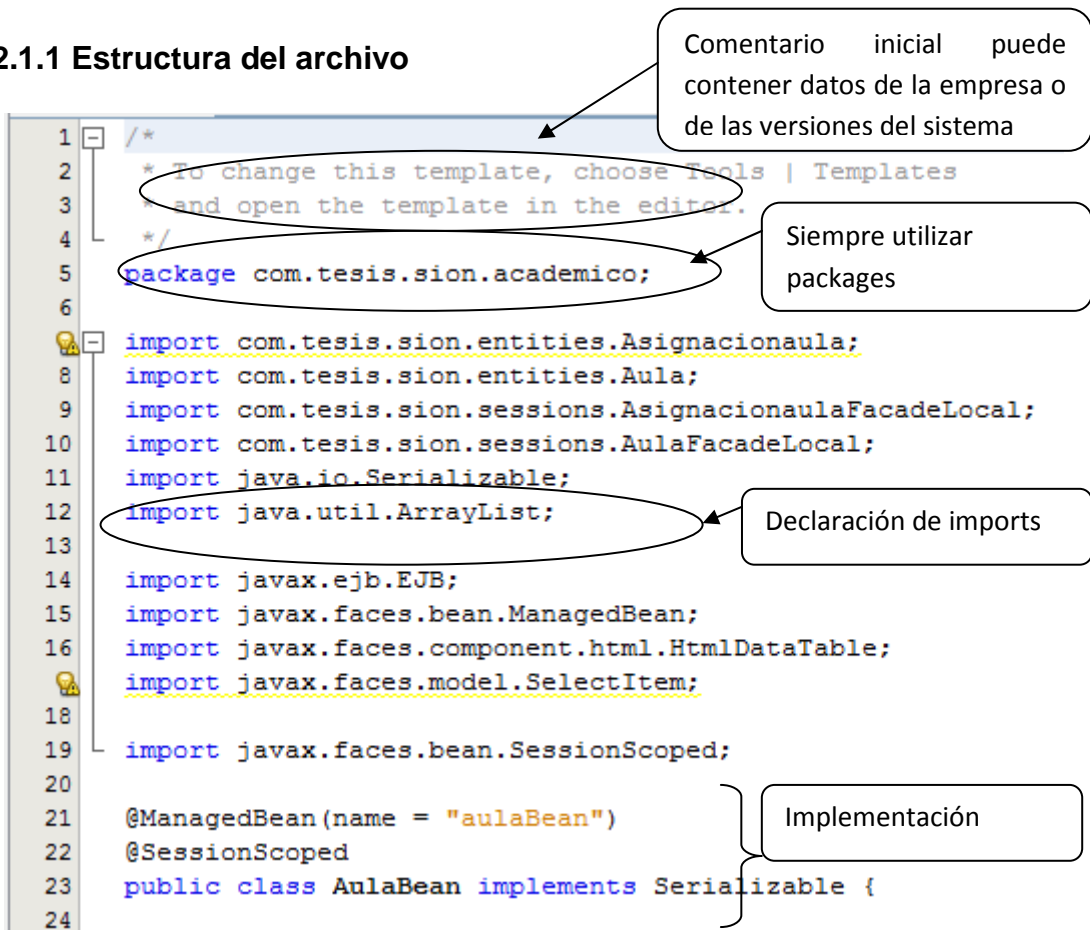


Figura 4. 2. :(Estructura del archivo Bean)

4.2.1.2 Nomenclatura de nombres

Nombres de paquete: Completamente en minúsculas. Ejemplo: com.tesis.sion.academico

Nombres de clase e interfaces: La primera letra de cada palabra debe ser mayúscula. Ejemplo: AulaBean

Nombres de métodos y variables: La primera palabra en minúsculas, y el resto comenzando en mayúsculas. Ejemplo: recuperarTodosAulas()

Nombres de constantes: Deben estar completamente en mayúsculas, las palabras se separan con _ (guion bajo). Ejemplo: USER_KEY

4.2.1.3 Declaración de variables.

```
public class AulaBean implements Serializable {  
  
    ///////////  
    // VARIABLES//  
    ///////////  
    @EJB  
    private AulaFacadeLocal aulaFacade = null;  
    @EJB  
    private AsignacionaulaFacadeLocal asignacionaulaFacade = null;  
    private ArrayList<Aula> listadoAula = new ArrayList<Aula>();  
    private Aula aula = null;  
    private Aula aulaInsertar = null;  
    //parametro busqueda  
    private String parametroBusqueda = ".*.*";  
    //Tabla  
    private HtmlDataTable tabla = null;  
    //Popup  
    private boolean visibilidadPopup = false;  
    private String tituloPopup = null;  
    private String contenidoPopup = null;  
    private String iconoPopup = null;  
}
```

Hacer una declaración por línea

Dar un valor inicial a la variable

Utilizar nombres descriptivos

Figura 4. 3. :(Declaración de variables)

4.2.1.4 Sentencias

```
public String guardarAula() {  
    Aula art = null;  
    if (aulaInsertar.getNombreAula() == null || aulaInsertar.getNombreAula()  
        visibilidadPopup = true;  
        abrirPopup("Los campos obligatorios no se han completado");  
    }  
    return contenidoPopup;  
}  
try {  
    art = aulaFacade.buscarNombreAula(aulaInsertar.getNombreAula()).get(  
} catch (Exception e) {  
    art = null;  
}  
if (aulaInsertar.getNombreAula() != null);  
{  
    if (art != null) {  
        abrirPopup("Ya existe un aula con el mismo nombre", "Error", "");//  
        return contenidoPopup;  
    }  
}
```

Utilizar siempre corchetes y paréntesis

Las sentencias try evitan excepciones generadas por la ausencia de valores en la base de datos

Figura 4. 4. :(Sentencias)

4.3 Desarrollo de los módulos del sistema

Durante la fase de desarrollo de la aplicación se han utilizado las siguientes herramientas:

- NetBeans IDE 7.1
- JDK 1.6

- PostgreSQL 9.0

Con la ayuda del IDE de desarrollo NetBeans se ha logrado generar el respectivo código fuente de la aplicación, además con PostgreSQL se ha llevado un manejo eficiente de la información, así como una correcta generación del script de la base de datos.

Para un correcto funcionamiento de la aplicación es necesario realizar pruebas, las mismas que se detallan a continuación:

4.4 Integración del Código

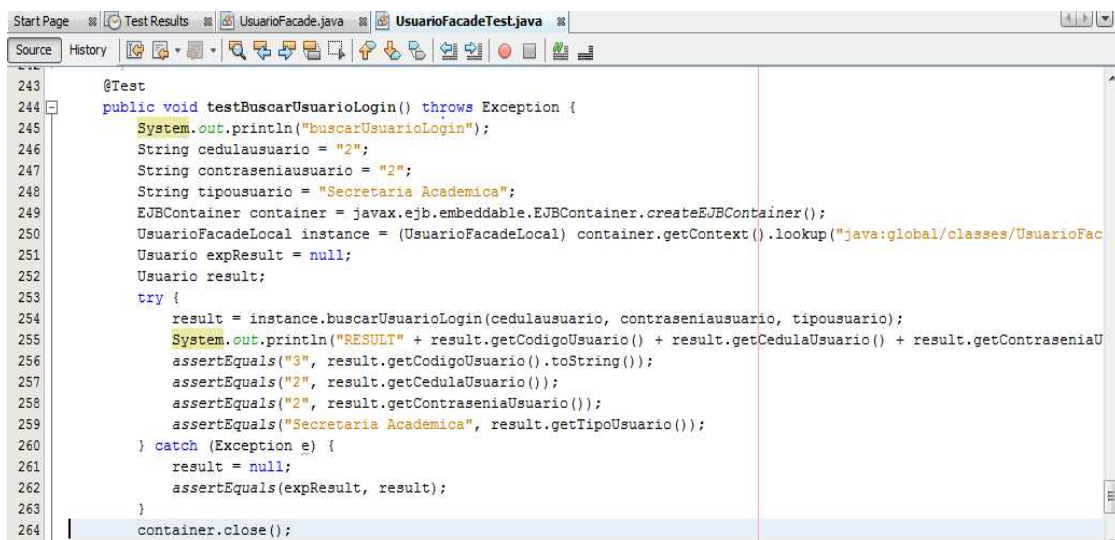
La integración del código se lo realizó de manera manual, con la inspección de cada uno de los testistas. El código fuente generado esta en el Anexo Digital.

4.5 Pruebas

4.5.1 Pruebas de Caja Blanca

Las pruebas de caja blanca están ligadas al código fuente de la aplicación, es decir comprueban el flujo de datos a través de las distintas funciones del sistema.

4.5.1.1 Buscar usuario para logearse



```
243 @Test
244 public void testBuscarUsuarioLogin() throws Exception {
245     System.out.println("buscarUsuarioLogin");
246     String cedulausuario = "2";
247     String contraseniausuario = "2";
248     String tipousuario = "Secretaria Academica";
249     EJBContainer container = javax.ejb.embeddable.EJBContainer.createEJBContainer();
250     UsuarioFacadeLocal instance = (UsuarioFacadeLocal) container.getContext().lookup("java:global/classes/UsuarioFac
251     Usuario expResult = null;
252     Usuario result;
253     try {
254         result = instance.buscarUsuarioLogin(cedulausuario, contraseniausuario, tipousuario);
255         System.out.println("RESULT" + result.getCodigoUsuario() + result.getCedulaUsuario() + result.getContraseniaU
256         assertEquals("3", result.getCodigoUsuario().toString());
257         assertEquals("2", result.getCedulaUsuario());
258         assertEquals("2", result.getContraseniaUsuario());
259         assertEquals("Secretaria Academica", result.getTipoUsuario());
260     } catch (Exception e) {
261         result = null;
262         assertEquals(expResult, result);
263     }
264     container.close();
}
```

Figura 4. 5. : (Código Junit testbuscarUsuarioLogin)

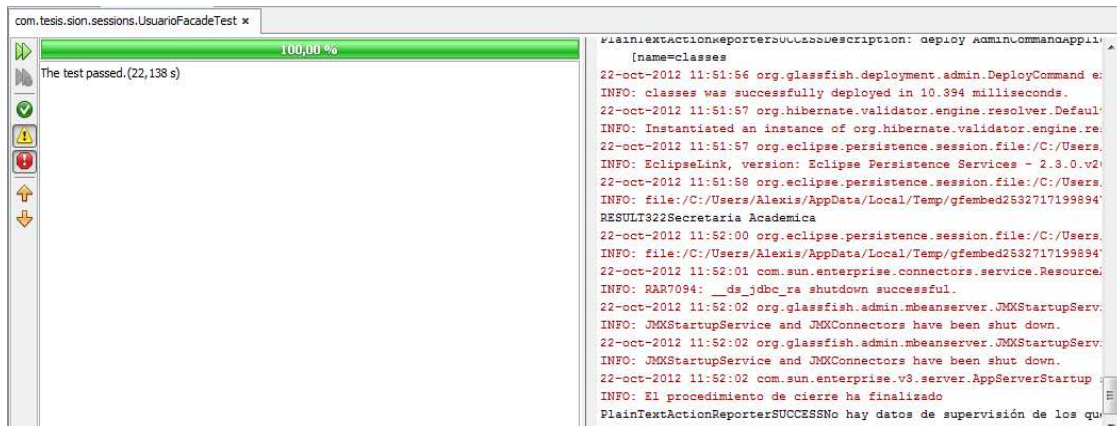


Figura 4. 6. : (Resultado Junit testbuscarUsuarioLogin)

Tabla 4. 1(Pueba de caja blanca - Buscar usuario para logearse)

Caso de Prueba	Resultado esperado	Resultado obtenido
Usuario existe	CodigoUsuario = "3", CedulaUsuario = "2", ContraseniaUsuario ="2", TipoUsuario = "Secretaria Academica";	CodigoUsuario = "3", CedulaUsuario = "2", ContraseniaUsuario ="2", TipoUsuario = "Secretaria Academica";
Usuario no existe	null	null

4.5.1.2 Buscar el código del último estudiante inscrito

```

235 // }
236 /**
237  * Test of retornaUltimoCodigoEstudiante method, of class EstudianteFacade.
238  */
239 @Test
240 public void testRetornaUltimoCodigoEstudiante() throws Exception {
241     System.out.println("retornaUltimoCodigoEstudiante");
242     EJBContainer container = javax.ejb.embeddable.EJBContainer.createEJBContainer();
243     EstudianteFacadeLocal instance = (EstudianteFacadeLocal) container.getContext().lookup("java:global/classes/Estu
244     Integer expectedResult = null;
245     Integer result;
246     try {
247         result = instance.retornaUltimoCodigoEstudiante();
248         assertEquals("2", result.toString());
249     } catch (Exception e) {
250         result = null;
251         assertEquals(expResult, result);
252     }
253
254     container.close();
255
256

```

Figura 4. 7. : (Código Junit testRetornaUltimoCodigoEstudiante)

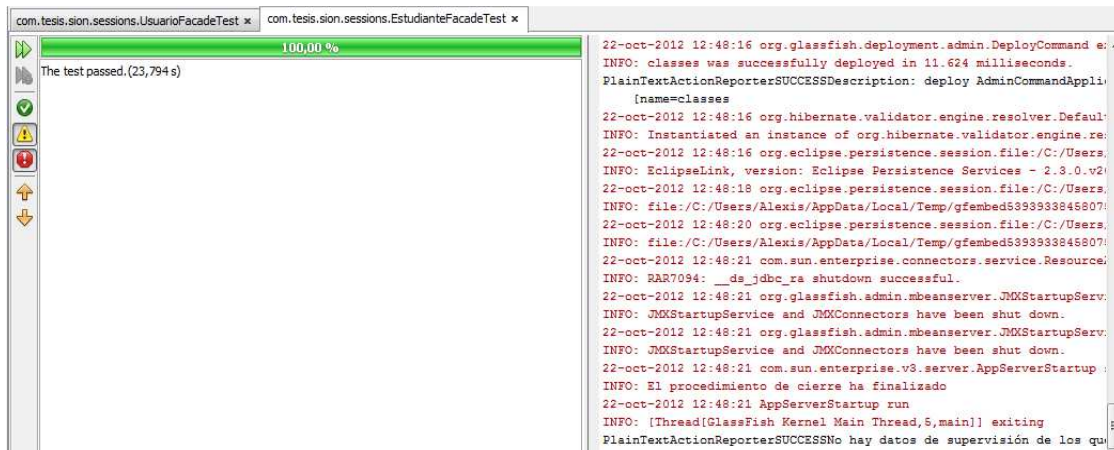


Figura 4. 8. : (Resultado Junit testRetornaUltimoCodigoEstudiante)

Tabla 4. 2. (Prueba de caja blanca - Buscar el código del último estudiante inscrito)

Caso de Prueba	Resultado esperado	Resultado obtenido
Existen estudiantes	2	2
No existen estudiantes	null	null

4.5.1.3 Buscar matrícula de un estudiante en un período académico específico

```

364 // }
365 /**
366  * Test of existeMatricula method, of class MatriculaFacade.
367  */
368 @Test
369 public void testExisteMatricula() throws Exception {
370     System.out.println("existeMatricula");
371     Integer codigoestudiante = 1;
372     Integer periodo = 1;
373     EJBContainer container = javax.ejb.embeddable.EJBContainer.createEJBContainer();
374     MatriculaFacadeLocal instance = (MatriculaFacadeLocal) container.getContext().lookup("java:global/classes/MatriculaFacadeLocal");
375     boolean expectedResult = true;
376     boolean result;
377
378     result = instance.existeMatricula(codigoestudiante, periodo);
379     assertEquals(expectedResult, result);
380
381     container.close();
382
383 }

```

Figura 4. 9. : (Código Junit testExisteMatrícula)

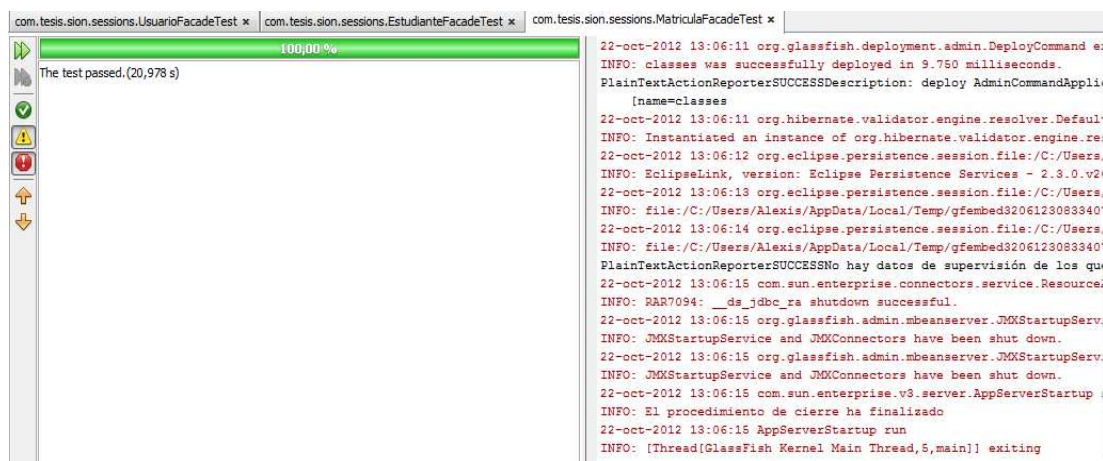


Figura 4. 10. : (Resultado Junit testExisteMatrícula)

Tabla 4. 3. (Prueba de caja blanca - Buscar matrícula de un estudiante en un período académico específico)

Caso de Prueba	Resultado esperado	Resultado obtenido
Existe matrícula	true	true
No existe matrícula	false	false

4.5.1.4 Buscar si existe una solicitud de transporte para una matrícula del período lectivo en curso

```

245  * test of existeSolicitudtransporte method, or class Solicitudserviciofacade.
246  */
247  @Test
248  public void testExisteSolicitudtransporte() throws Exception {
249      System.out.println("existeSolicitudtransporte");
250      Integer numeromatrícula = 1;
251      EJBContainer container = javax.ejb.embeddable.EJBContainer.createEJBContainer();
252      SolicitudservicioFacadeLocal instance = (SolicitudservicioFacadeLocal) container.getContext().lookup("java:global
253      Boolean expResult = true;
254      Boolean result = instance.existeSolicitudtransporte(numeromatrícula);
255      assertEquals(expResult, result);
256      container.close();
257  }
258
259  }
260

```

Figura 4. 11. : (Código Junit testExisteSolicitudtransporte)

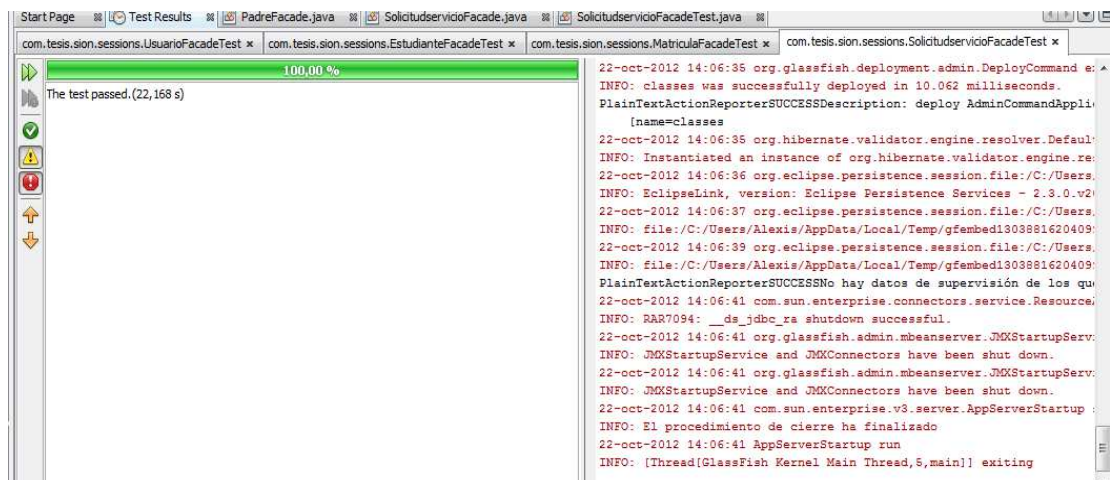


Figura 4. 12. : (Resultado Junit testExisteSolicitudtransporte)

Tabla 4. 4. (Prueba de caja blanca - Buscar si existe una solicitud de transporte para una matrícula del período lectivo en curso)

Caso de Prueba	Resultado esperado	Resultado obtenido
Existe solicitud de transporte	true	true
No existe solicitud de transporte	false	false

4.5.1.5 Buscar el código del último pedido generado

```

219 * Test of retornaUltimoCodigoPedido method, of class PedidoFacade.
220 */
221 @Test
222 public void testRetornaUltimoCodigoPedido() throws Exception {
223     System.out.println("retornaUltimoCodigoPedido");
224     EJBContainer container = javax.ejb.embeddable.EJBContainer.createEJBContainer();
225     PedidoFacadeLocal instance = (PedidoFacadeLocal) container.getContext().lookup("java:global/classes/PedidoFacade");
226     Integer expectedResult = null;
227     Integer result;
228     try {
229         result = instance.retornaUltimoCodigoPedido();
230         assertEquals("4", result.toString());
231     } catch (Exception e) {
232         result = null;
233         assertEquals(expectedResult, result);
234     }
235
236     container.close();
237
238 }
239 }

```

Figura 4. 13. : (Código Junit testRetornaUltimoCodigoPedido)

100,00 %

The test passed. (25,584s)

```

PlainTextActionReporterSUCCESSDescription: deploy AdminCommandAppli
[name=classes
22-oct-2012 14:50:58 org.glassfish.deployment.admin.DeployCommand e
INFO: classes was successfully deployed in 10.858 milliseconds.
22-oct-2012 14:50:58 org.hibernate.validator.engine.resolver.Default
INFO: Instantiated an instance of org.hibernate.validator.engine.re
22-oct-2012 14:50:58 org.eclipse.persistence.session.file:/C:/Users
INFO: EclipseLink, version: Eclipse Persistence Services - 2.3.0.v2
22-oct-2012 14:50:59 org.eclipse.persistence.session.file:/C:/Users
INFO: file:/C:/Users/Alexis/AppData/Local/Temp/gfembed8744961943565
22-oct-2012 14:51:04 org.eclipse.persistence.session.file:/C:/Users
INFO: file:/C:/Users/Alexis/AppData/Local/Temp/gfembed8744961943565
22-oct-2012 14:51:06 com.sun.enterprise.connectors.service.Resource
INFO: RAR7094: __ds_jdbc_ra shutdown successful.
22-oct-2012 14:51:06 org.glassfish.admin.mbeanserver.JMXStartupServ
INFO: JMXStartupService and JMXConnectors have been shut down.
22-oct-2012 14:51:06 org.glassfish.admin.mbeanserver.JMXStartupServ
INFO: JMXStartupService and JMXConnectors have been shut down.
22-oct-2012 14:51:06 com.sun.enterprise.v3.server.AppServerStartup
INFO: El procedimiento de cierre ha finalizado
22-oct-2012 14:51:06 AppServerStartup run
INFO: [Thread[GlassFish Kernel Main Thread,5,main]] exiting
PlainTextActionReporterSUCCESSNo hay datos de supervisión de los qu

```

Figura 4. 14. : (Resultado Junit testRetornaUltimoCodigoPedido)

Tabla 4. 5. (Prueba de caja blanca - Buscar el código del último pedido generado)

Caso de Prueba	Resultado esperado	Resultado obtenido
Existen pedidos	4	4
No existen pedidos	null	null

4.5.1.6 Verificar que la fecha no posea registros de asistencias de docentes

```

167  /**
168   * Test of aceptaFechaAsistenciadocente method, of class
169   * AsistenciadocenteFacade.
170   */
171  @Test
172  public void testAceptaFechaAsistenciadocente() throws Exception {
173      System.out.println(" aceptaFechaAsistenciadocente");
174      Date fechaasistenciadocente;
175      SimpleDateFormat formatoDelTexto = new SimpleDateFormat("yyyy-MM-dd");
176      String strFecha = "2012-10-22";
177      fechaasistenciadocente = formatoDelTexto.parse(strFecha);
178      EJBContainer container = javax.ejb.embeddable.EJBContainer.createEJBContainer();
179      AsistenciadocenteFacadeLocal instance = (AsistenciadocenteFacadeLocal) container.getContext().lookup("java:globa
180      Boolean expResult = true;
181      Boolean result = instance.aceptaFechaAsistenciadocente(fechaasistenciadocente);
182      assertEquals(expResult, result);
183      container.close();
184  }
185  }

```

Figura 4. 15. : (Código Junit testAceptaFechaAsistenciadocente)

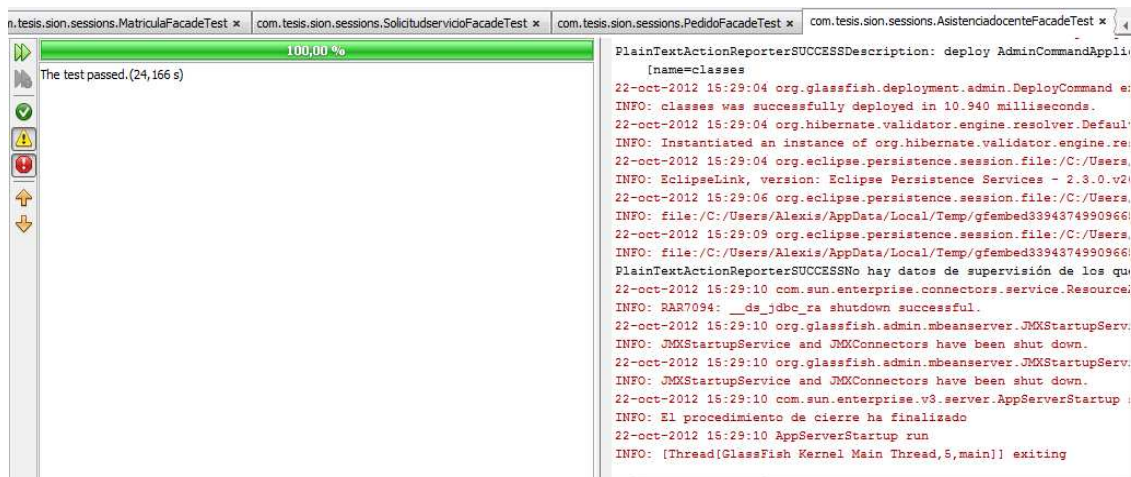


Figura 4. 16. : (Resultado Junit testAceptaFechaAsistenciadocente)

Tabla 4. 6. (Prueba de caja blanca - Verificar que la fecha no posea registros de asistencias de docentes)

Caso de Prueba	Resultado esperado	Resultado obtenido
Existen asistencias de docentes	true	true
No existen asistencias de docentes	false	false

4.5.2 Pruebas de caja negra

Las pruebas de caja negra verifican que cuando se ingresan los datos correctos se obtienen los resultados esperados. Se enfoca en los requisitos del sistema, previniendo errores en las funciones, errores en la interfaz de usuario, errores en acceso a la base de datos, errores al inicializar o finalizar la aplicación, etc.

Se proveen unos datos de entrada y se estudian las salidas que generan para ver si concuerdan con los resultados esperados.

4.5.2.1 Guardar Docente

Tabla 4. 7. : (Pruebas de caja negra-Guardar docente)

Caso de Prueba	Resultado esperado	Resultado obtenido
cedulaDocente= "1717655490" nombreDocente= "Ibeth Yomara" apellidoDocente= "Montalvo Salas" fechanacimientoDocente=13/10/1982 direccionDocente= "Av. Tnte. Hugo Ortiz Cusumaza Oe3-131" sectorDocente= "Cdla. La Internacional" telefonoDocente= "2627742" celularDocente= "0992842099" emailDocente= "i.bethlibra82@hormail.com" estadocivilDocente= "Soltero" fechaingresoDocente=12/12/2005 relacionlaboralDocente= "Contratado" tipoDocente= "Curso"	Docente guardado con éxito	Docente guardado con éxito

estadoDocente= "Disponible"		
-----------------------------	--	--

4.5.2.2 Modificar Contraseña

Tabla 4. 8. : (Pruebas de caja negra-Modificar Contraseña)

Caso de Prueba	Resultado esperado	Resultado obtenido
codigoUsuario=3 cedulaUsuario= "1719939710" contraseniaUsuario="2" tipoUsuario="Docente"	Contraseña Modificada con éxito	Contraseña Modificada con éxito

4.5.2.3 Guardar Curso

Tabla 4. 9. : (Pruebas de caja negra-Guardar Curso)

Caso de Prueba	Resultado esperado	Resultado obtenido
nombreCurso= "Nursery" paraleloCurso="A" estadoCurso="Disponible" tipoCurso="Basica"	Curso Guardado con éxito	Curso Guardado con éxito

4.5.2.4 Buscar Aula

Tabla 4. 10. : (Pruebas de caja negra-Buscar Aula)

Caso de Prueba	Resultado esperado	Resultado obtenido
nombreAula="A1"	codigoAula=1	codigoAula=1

	nombreAula="A1" capacidadAula=15 observacionAula="Piso 1" estadoAula="Disponible"	nombreAula="A1" capacidadAula=15 observacionAula="Piso 1" estadoAula="Disponible"
--	--	--

4.5.2.5 Guardar Producto

Tabla 4. 11. : (Pruebas de caja negra-Guardar Producto)

Caso de Prueba	Resultado esperado	Resultado obtenido
nombreProducto= "Medias" descripcionProducto= "Talla S" marcaProducto= "Umbro" valorunitarioProducto= 1.5 stockProducto= 30 stockminimoProducto= 8 descripcionivaProducto= "Si" tipoProducto= "Producto"	Producto Guardado con éxito	Producto Guardado con éxito

4.5.2.6 Guardar Inscripción

Tabla 4. 12. : (Pruebas de caja negra- Guardar Inscripción)

Caso de Prueba	Resultado esperado	Resultado obtenido
<p>cedulaEstudiante="1718552610" nombreEstudiante="Fatima Sarahi" apellidoEstudiante="Barrionuevo Zapata" emailEstudiante="" lugarnacimientoEstudiante="Quito" fechanacimientoEstudiante=21/05/2007 nacionalidadEstudiante="Ecuatoriano" direccionEstudiante="El canelo S11" numerocasaEstudiante=234 barrioEstudiante="Santa Anita" telefonoEstudiante="3102546" viveconEstudiante="Padres" medicoEstudiante="" escuelaanteriorEstudiante="Las Abejitas" numerohermanosEstudiante=0 sexoEstudiante="Femenino" fechainscripcionEstudiante=18/10/2012 nombredetallefacturaEstudiante="Silvana" apellidodetallefacturaEstudiante="Zapata" ceduladetallefacturaEstudiante="1718552609" sectordetallefacturaEstudiante="El canelo S11" telefonodetallefacturaEstudiante="84589696" codigoPadre=3 codigoMadre=4</p>	<p>Estudiante guardado con éxito</p>	<p>Estudiante guardado con éxito</p>

4.5.3 Pruebas de Stress

Las pruebas de stress buscan validar el correcto funcionamiento del sistema bajo las condiciones normales de trabajo, para obtener resultados sobre el tiempo de respuesta del sistema.

Para la realización de dicha prueba se utilizará la herramienta libre Jmeter.

Jmeter es una herramienta que permite ejecutar pruebas de rendimiento, funcionales y pruebas de carga excesiva sobre aplicaciones Web. Es una aplicación que permite hacer simulaciones sobre cualquier recurso del sistema desarrollado.

Las pruebas de stress determinan la capacidad de respuesta del sistema a las solicitudes de los usuarios, para la prueba se han configurado en Jmeter los siguientes valores:

- Número de Usuarios concurrentes con peticiones de solicitud a la aplicación : 30
- Tiempo de espera entre solicitudes 5 segundos
- Número de veces que se repite la operación 1

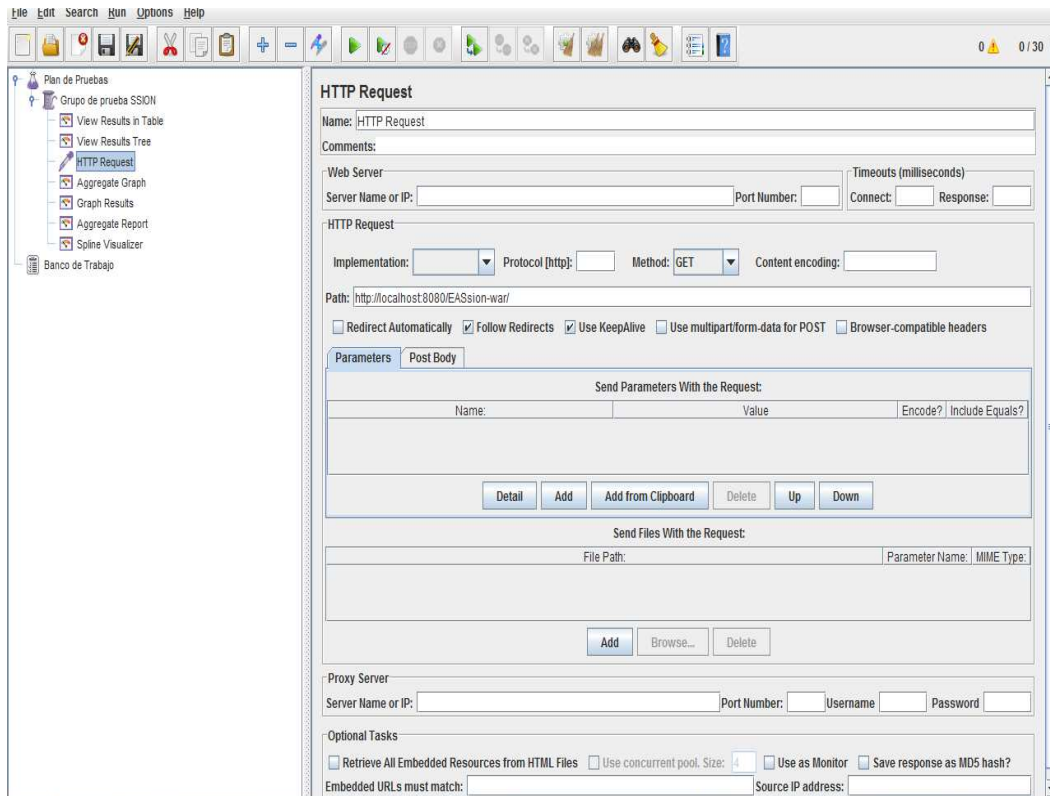


Figura 4. 17. : (Resultado Junit testAceptaFechaAsistenciadocente)

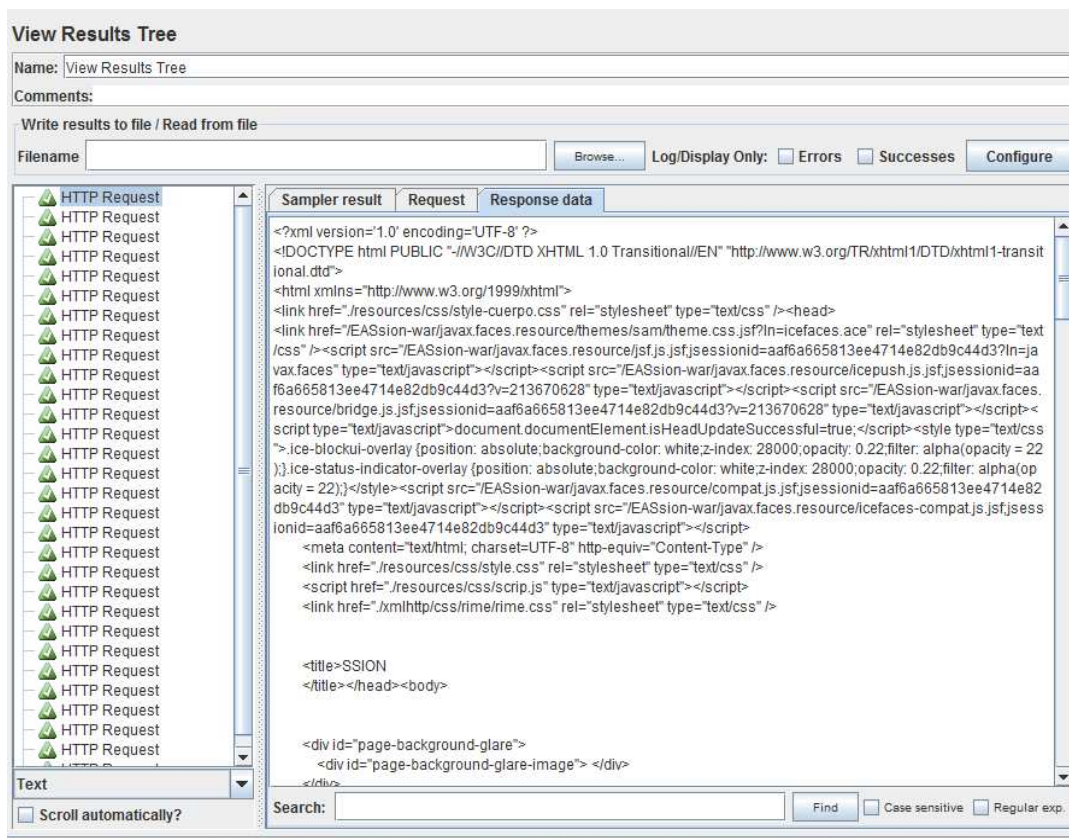


Figura 4. 18. (Pruebas de Stress -Response data de la página login)

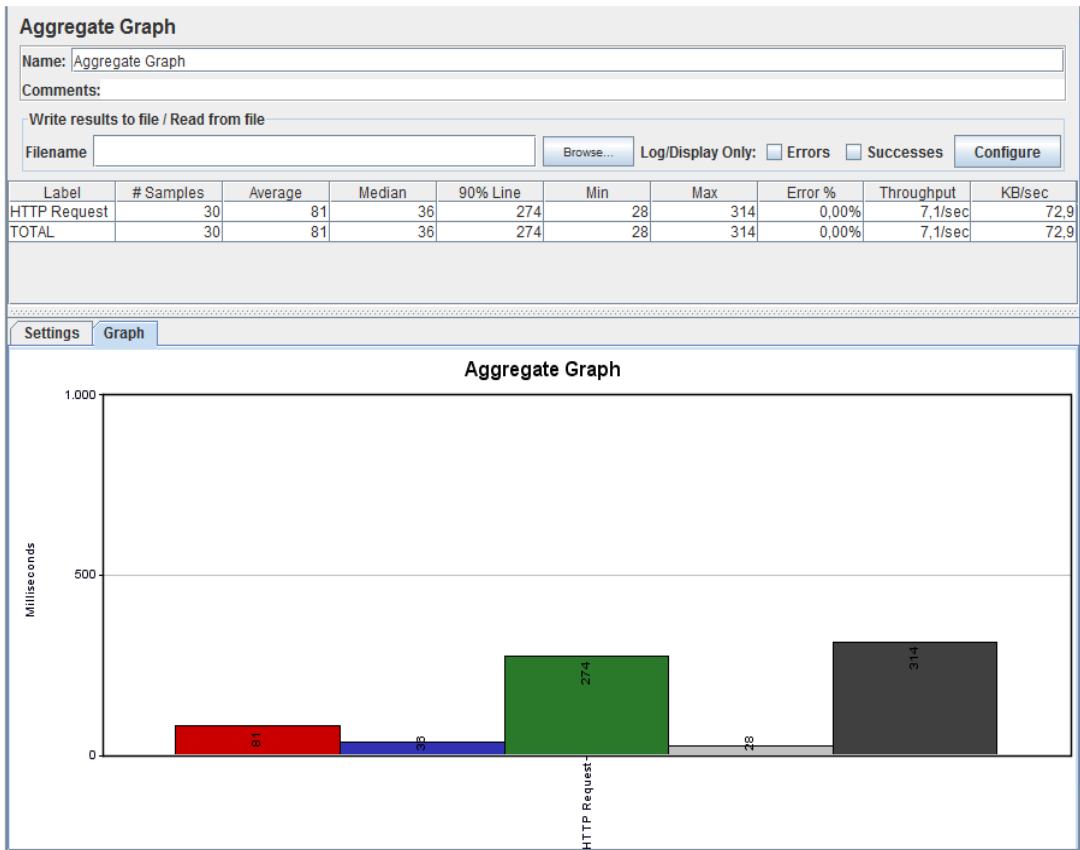


Figura 4. 19. (Pruebas de Stress-Gráfico tiempos de respuesta)

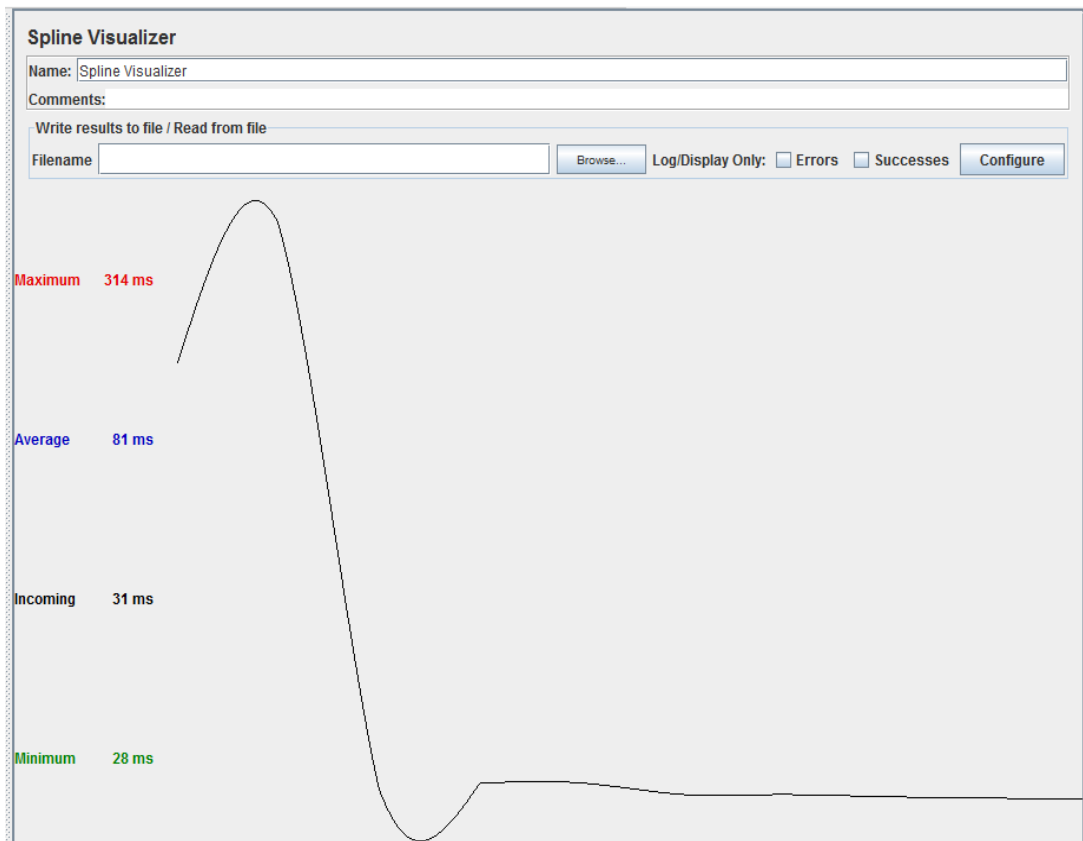


Figura 4. 20. (Pruebas de Stress- Tiempo Máximo y Mínimo de respuesta)

El tiempo medio de respuesta de la aplicación ante una solicitud de servicio de 30 usuarios es de 81 milisegundos. El tiempo máximo es de 314 milisegundos y el mínimo de 28 milisegundos.

ANEXO A: Diagrama de Clases, Diagramas de Secuencia, Diagramas de Estado. Diagrama de Despliegue, Diagramas Navegacionales, Diagramas de Presentación, Diagramas de Actividades, Diagrama Conceptual de la base de datos, Diagrama Físico de la base de datos.

Los diagramas se encuentran en el Tomo II Anexo A

ANEXO B: Manual técnico del sistema SSION

El manual técnico se encuentra en el Tomo II Anexo B

ANEXO C: Manual del usuario del sistema SSION

El manual del usuario se encuentra en el Tomo II Anexo C

ANEXO D: Manual del Programador del Sistema SSION

El manual del programador se encuentra en el Tomo II Anexo D

ANEXO E: Carta de auspicio, Carta de Aceptación

Las cartas de auspicio y aceptación emitidas por la escuela SION International Christian School se encuentran en el Tomo II Anexo E

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- Se desarrolló una aplicación web debido a que los usuarios debían tener fácil acceso al sistema y este debía estar disponible para cualquier persona de la institución.
- El sistema fue desarrollado con herramientas Open Source como son: Netbeans, PostgreSQL, Glassfish, etc. Permitiendo así una reducción significativa en el costo total de producción.
- El sistema ha permitido automatizar el proceso de inscripción y matriculación, reduciendo las actividades que anteriormente realizaban los usuarios, brindando un servicio más rápido a los estudiantes de la Escuela SION.
- A través de la presente tesis se ha logrado reducir significativamente el tiempo en los procesos de recolección de información de los estudiantes que posee la Escuela SION.
- El manejo del sistema ha sido difundido completamente en la escuela y para mejor comprensión se ha facilitado el manual del usuario de la presente tesis.
- Al realizar una aplicación web debido a la existencia de múltiples sistemas operativos y navegadores web, hay la posibilidad de encontrarse con incompatibilidades. Por ejemplo al momento de lanzarse una ventana emergente en Internet Explorer esta no se visualiza correctamente a menos que el navegador haya sido previamente configurado de acuerdo a las características del sistema.

5.2 Recomendaciones

- Es de suma importancia realizar un correcto levantamiento de los requerimientos, puesto que así no se retrasará el proceso de desarrollo del sistema, ya que un proceso no definido, al final de la aplicación puede traer caos y re codificación a los desarrolladores. Para lo cual se recomienda realizar cuestionarios y entrevistas con cada uno de los usuarios del sistema, para conocer como se están realizando los procesos a ser automatizados.
- Se recomienda implementar una adecuada infraestructura para el servidor, puesto que así se asegura el correcto funcionamiento del sistema durante un mayor período de tiempo. Por ejemplo una fuente de energía sustentable en caso de pérdida de la misma.
- Es importante el correcto manejo de la seguridad del sistema. Delimitando así a que páginas va a tener acceso cada usuario, ya que la información académica de los estudiantes es de suma importancia.
- Se recomienda usar tipos de datos seriales como clave primaria en las tablas de la base de datos, para facilitar al administrador el mantenimiento y corrección de errores en la misma. Por ejemplo si una tabla tiene un serial de 1 y dicho secuencial se relaciona con otra tabla, esta clave siempre será 1, manteniendo así la integridad de los datos.
- Es recomendable definir una estructura navegacional, para que así el usuario pueda dirigirse de una manera rápida y fluida a través de las diferentes opciones del sistema.

BIBLIOGRAFÍA Y WEBGRAFÍA

[1] Ciberneta, Historia de las Aplicaciones Web [ONLINE]

http://www.ciberneta.com/manuales/introduccion_aplicaciones_web/3_historia_aplicaciones_web.php

[2] Diccionario de informática Alegsa [ONLINE]

<http://www.alegsa.com.ar/Dic/herramienta%20de%20modelado.php>

[3] De Wikipedia, la enciclopedia libre, Java (lenguaje de programación) [ONLINE]

http://es.wikipedia.org/wiki/Java_%28lenguaje_de_programaci%C3%B3n%29

[4] Sicuma, Tutorial de Java Server Faces [ONLINE]

<http://www.sicuma.uma.es/sicuma/Formacion/documentacion/JSF.pdf>

[5] PostgreSQL, Manual de PostgreSql [ONLINE]

http://www.ecualug.org/2005/11/20/postgresql_robusto_como_un_elefante?device=mobile

[6] De Wikipedia, la enciclopedia libre, GlassFish [ONLINE]

<http://es.wikipedia.org/wiki/GlassFish>

[7] Jorge Sánchez, Diseño conceptual de base de datos [ONLINE]

<http://www.jorgesanchez.net/bd/disenioBD.pdf>

[8] LMU–Ludwig-Maximilians-Universität München Institute for Informatics, Research Unit of Programming and Software Engineering, UWE [ONLINE]

<http://uwe.pst.ifi.lmu.de/teachingTutorialSpanish.html>

[9] Koch, Nora, "Perfil y Metamodelo de UWE", 2008

[10] De Wikipedia, la enciclopedia libre, Lenguaje Unificado de Modelado

[ONLINE] http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado

[11] Según la especificación de UML 2.2 publicada por la Object Management Group (OMG) en febrero de 2009.

[12] Roger S. Presman "Ingeniería de Software un enfoque práctico", 2002, pág. 172