

UNIVERSIDAD NACIONAL DE SAN CRISTOBAL DE HUAMANGA

Facultad de Ingeniería de Minas, Geología y Civil

Escuela de Formación Profesional de Ingeniería de Sistemas



“Aplicación Web para automatizar el proceso de matrícula de la Universidad Nacional de San Cristóbal de Huamanga, 2015”

Tesis presentada por : Bach. Liz Karla Paucar Rondinel

Para optar el título profesional de : Ingeniera de Sistemas

Tipo de investigación : Aplicada

Área de investigación : Ingeniería de Software

Asesora : Ing. Elinar Carrillo Riveros

Ayacucho, Setiembre del 2016

DEDICATORIA

A Dios quien siempre me acompaña en los distintos momentos de mi vida, a mis padres, Guillermo Nicanor Paucar Cusihuamán y Asunción Rondinel Ochante, por el esfuerzo y apoyo incondicional durante mi formación y por ser el ejemplo a seguir, y a mis pequeños hermanos.

AGRADECIMIENTOS

A la Universidad Nacional de San Cristóbal de Huamanga, mi alma máter.

A la Escuela de Formación Profesional de Ingeniería de Sistemas – UNSCH, por haberme albergado en sus aulas durante estos años de formación.

A los docentes de la Universidad Nacional de San Cristóbal de Huamanga, de los distintos departamentos académicos, por su contribución durante mi desarrollo académico y profesional.

A la Ingeniera Elinar Carrillo Riveros, por inculcar los conocimientos básicos e impulsar el desarrollo de la investigación, gracias por todo ese apoyo.

CONTENIDO

	Página
DEDICATORIA -----	i
AGRADECIMIENTOS -----	ii
CONTENIDO -----	iii
RESUMEN -----	v
INTRODUCCIÓN -----	vi

CAPITULO I

PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1	DIAGNOSTICO Y ENUNCIADO DEL PROBLEMA -----	1
1.2	DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN -----	4
1.3	OBJETIVOS DE LA INVESTIGACION -----	4
1.4	JUSTIFICACION Y DELIMITACION DE LA INVESTIGACION -----	4
1.4.1	IMPORTANCIA DE LA INVESTIGACION -----	4
1.4.2	JUSTIFICACION -----	5
1.4.3	DELIMITACION -----	5

CAPITULO II

REVISIÓN DE LITERATURA

2.1	ANTECEDENTES DE LA INVESTIGACIÓN -----	6
2.2	MARCO TEÓRICO -----	7
2.2.1	AUTOMATIZACIÓN-----	7
2.2.1.1	ARTEFACTO-----	7
2.2.1.2	PROTOTIPO-----	8
2.2.2	PROCESO DE MATRICULA-----	8
2.2.2.1	FICHA DE MATRICULA-----	8
2.2.2.2	REGISTRO DE EVALUACION-----	8
2.2.2.3	ACTA DE EVALUACION FINAL-----	9
2.2.2.4	ESQUELA DE PAGOS-----	9
2.2.3	METODOLOGIA ICONIX-----	9
2.2.4	APLICACIÓN WEB-----	17
2.2.5	PROGRAMACIÓN ORIENTADA A OBJETOS -----	18
2.2.6	SISTEMA GESTOR DE BASE DE DATOS-----	19
2.2.7	TECNOLOGIAS DE INTERNET-----	20

CAPITULO III

METODOLOGÍA DE LA INVESTIGACIÓN

3.1	TIPO DE INVESTIGACIÓN-----	21
3.2	DISEÑO DE LA INVESTIGACION -----	21
3.3	POBLACIÓN Y MUESTRA-----	22
3.4	VARIABLES -----	22
3.4.1	DEFINICION CONCEPTUAL DE LAS VARIABLES-----	22
3.4.2	DEFINICION OPERACIONAL DE LAS VARIABLES-----	24
3.5	TÉCNICAS E INSTRUMENTOS-----	24
3.5.1	INSTRUMENTOS PARA RECOLECTAR INFORMACION -----	24
3.5.2	HERRAMIENTAS PARA EL TRATAMIENTO DE DATOS E INFORMACION-----	25
3.5.3	TECNICA PARA APLICAR ICONIX-----	27

CAPITULO IV

ANALISIS Y RESULTADOS DE LA INVESTIGACION

4.1	PROCESO DE MIGRACIÓN DE ARCHIVOS RELACIONADOS DE FOXPRO A UNA BASE DE DATOS ORACLE-----	32
4.2	ANALISIS DE LOS PROCESOS QUE INTERVIENE EN LA INVESTIGACION-----	33
4.3	ARTEFACTOS DEL SOFTWARE APLICANDO LA METODOLOGIA ICONIX-----	38
4.3.1	ANALISIS DE REQUISITOS-----	38
4.3.2	REVISION DE REQUISITOS-----	62
4.3.3	DISEÑO PRELIMINAR-----	72
4.3.4	REVISION DEL DISEÑO PRELIMINAR-----	79
4.3.5	ARQUITECTURA TECNICA-----	79
4.3.6	DISEÑO DETALLADO-----	81
4.3.7	REVISION CRITICA DE DISEÑO-----	88
4.3.8	IMPLEMENTACION-----	89

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1	CONCLUSIONES-----	100
5.2	RECOMENDACIONES-----	101
	BIBLIOGRAFÍA-----	102
	ANEXOS-----	105

RESUMEN

Las Secretarías de las Escuelas de Formación Profesional de la Universidad Nacional de San Cristóbal de Huamanga, usan el sistema de proceso de matrícula denominado SISACAD desarrollado en el año de 1998. Este sistema no cumple con los requerimientos actuales de los procesos académicos que se desean realizar, generando demora en la gestión de sus procesos, especialmente en el proceso de matrícula; lo cual repercute en la incomodidad, tanto de estudiantes, docentes y personal administrativo.

Actualmente, la universidad ha promovido la matrícula virtual con el sistema SIMA, que facilita al estudiante la matrícula de los diferentes cursos de acuerdo a su plan de estudios y la generación de la Esquela de Pago. Pero no abarca todas las actividades que implican un proceso de matrícula, tales como la gestión de Registros de Evaluaciones y Actas de Evaluación Final.

El objetivo de esta investigación es desarrollar una aplicación web para automatizar el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, con la finalidad de lograr un control integral del proceso, de manera que sea confiable, rápido y seguro.

La investigación se realizó tomando como muestra la información de la Escuela de Formación Profesional de Ingeniería de Sistemas de la Universidad Nacional de San Cristóbal de Huamanga.

La investigación es de tipo aplicada a un nivel descriptivo, utilizando el paradigma del diseño orientado a objetos, la metodología Iconix, y como herramienta un sistema gestor de base de datos relacional; Con la técnica de la observación y análisis documental.

Palabras claves: Automatizar, Proceso de matrícula, Metodología Iconix, Aplicación Web.

INTRODUCCION

Los sistemas de información bajo ambiente Web han sido adoptados por diferentes instituciones para permitir mejorar y estar a la vanguardia del desarrollo tecnológico, es así que una de las instituciones que han tomado conciencia de las ventajas que ofrece la web son las universidades y han hecho uso de ella para poder facilitar la gestión de sus procesos académicos.

La motivación para realizar este trabajo de investigación, es presentar una solución tecnológicamente adecuada para apoyar en el proceso de matrícula, el cual aún sigue sin resolver en la Universidad Nacional de San Cristóbal de Huamanga.

Para lo cual, se llevó acabo el desarrollo de una aplicación web que automatice el proceso de matrícula de los estudiantes en la Universidad Nacional de San Cristóbal de Huamanga.

El presente trabajo de investigación se estructura en 5 capítulos; el primer capítulo contiene el planteamiento y formulación del problema en estudio; el segundo capítulo contiene el marco conceptual; el tercer capítulo se refiere a la metodología; el cuarto capítulo se presenta los resultados de la investigación y como quinto capítulo tenemos las conclusiones y recomendaciones.

CAPITULO I

PLANTEAMIENTO DE LA INVESTIGACION

1.1 DIAGNOSTICO Y ENUNCIADO DEL PROBLEMA

Actualmente, las secretarías de las Escuelas de Formación Profesional, usan el sistema de proceso de matrícula denominado SISACAD, que es una evolución del primer sistema desarrollado en 1987 para un miniordenador WANG VS 5 codificado en Cobol ANS77, el cual fue migrado en 1998 para ser operado en microordenadores compatibles codificado en FoxPro 2.6. Este sistema aún sigue vigente con algunas mejoras en su operatividad de acuerdo a necesidades ocasionales de usuarios finales.

A pesar de las mejoras, existe un deficiente proceso de gestión académica, especialmente en el proceso de matrícula, lo cual estuvo reflejado en las largas colas generadas por los estudiantes para poder ser atendidos, haciendo poco eficiente la labor de los secretarios de escuela; generando errores en la matrícula, ya sea al momento de registrar las fichas de matrícula al sistema, las cuales son rellenas por cada estudiante de manera manual, cabiendo la posibilidad de ingresos erróneos por fallas humanas al momento de consignar sus datos.

El sistema SISACAD, permitía que el alumno se matricule un curso a pesar de haberlo llevado anteriormente (duplicidad de notas), matriculas en cursos sin cumplir los requisitos.

Por estos inconvenientes se implementa el sistema SIMA (Sistema de Matrícula) en su versión beta, aplicación web para el proceso de matrícula, para el semestre académico 2015-I, para determinadas escuelas, entre las que se encontraba la E.F.P de Administración de Empresas.

En primera instancia, a pesar de ser una aplicación web, se siguió con los procedimientos tradicionales de hacer colas para solicitar la esuela de pago, además se tuvo

inconvenientes al momento de realizar el proceso de matrícula, porque no se tenía actualizado la base de datos.

La aplicación web SIMA permite al estudiante matricularse en los cursos virtualmente, pero la ficha de matrícula en físico se tiene que entregar a la dirección de escuela para procesar la matrícula en el sistema SISACAD.

Esta aplicación web implementada, tiene falencias, como: No imprimir las boletas de notas de otros semestres académicos, no presenta un módulo específico para que el docente pueda registrar sus notas e imprimir sus registros de evaluaciones, no presenta un módulo para el personal del OGIS para la emisión de las Actas de Evaluación Final.

A su vez, se debe tomar en cuenta que la cantidad de matriculados desde los últimos años tiene una tendencia a aumentar progresivamente, lo cual también dificulta el proceso de matrícula, como se podrá detallar en la siguiente tabla:

Escuela de Formación Profesional	2011-I	2011-II	2012-I	2012-II	2013-I	2013-II	2014-I	2014-II	2015-I	2015-II	2016-I
AGRONOMIA	570	514	566	509	568	534			603	547	593
BIOLOGIA	465	454	511	481	504	492			512	480	507
EDUCACION INICIAL	140	153	163	154	190	187			207	204	191
EDUCACION PRIMARIA	171	173	184	178	205	190			190	182	163
EDUCACION SECUNDARIA	412	420	469	440	458	429			433	413	396
EDUCACION FISICA	132	123	166	151	178	174			193	180	205
ADMINISTRACION DE EMPRESAS	645	621	629	604	640	628			676	636	649
CONTABILIDAD Y AUDITORIA	684	679	698	687	710	717			715	717	680
ECONOMIA	671	641	705	672	703	680			731	684	738
ANTROPOLOGIA SOCIAL	384	336	376	348	375	342			379	348	366
ARQUEOLOGIA E HISTORIA	282	280	286	284	290	281			275	270	292
SERVICIO SOCIAL	359	356	354	354	377	377			390	366	396
DERECHO	698	680	717	696	695	685			717	676	706
ENFERMERIA	433	421	455	430	437	428			459	428	456
INGENIERIA DE MINAS	369	353	365	353	398	392			442	418	489
INGENIERIA CIVIL	547	535	571	563	599	593			666	632	712
INGENIERIA QUIMICA	201	188	236	215	234	222			258	240	250
OBSTETRICIA	463	418	534	488	502	444			514	469	500
ING. EN INDUSTRIAS ALIMENTARIAS	305	274	316	287	326	294			328	305	329
FARMACIA Y BIOQUIMICA	430	433	448	467	467	456			471	451	477
INGENIERIA AGRICOLA	531	519	553	532	578	557			618	581	644
INGENIERIA AGROINDUSTRIAL	275	263	318	305	322	316			321	292	338
CIENCIAS DE LA COMUNICACIÓN	396	398	417	395	441	421			428	396	406
MEDICINA VETERIANARIA	280	269	279	264	279	265			262	250	269
INGENIERIA INFORMATICA	0	0	0	0	0	4			0	0	0
CIENCIAS FISICO - MATEMATICAS	177	150	205	169	200	170			183	151	176
INGENIERIA DE SISTEMAS	466	453	496	475	495	465			513	495	542
INGENIERIA AGROFORESTAL	133	126	179	171	225	210	240	228	215	201	
MEDICINA HUMANA	0	0	36	27	63	60			116	109	159
TOTAL	10619	10230	11232	10699	11459	11013	240	228	11815	11121	11629

Tabla N° 1.1: Matriculados 2011 -2016 por Escuelas de Formación Profesional (Oficina de Informática y Sistemas, 2016)

Teniendo la necesidad de implementar un sistema web, que permita automatizar el proceso de matrícula en la Universidad Nacional de San Cristóbal de Huamanga, para los distintos actores ya sean estudiantes, docentes y personal de la Oficina General de Informática y Sistemas de la universidad.

1.2 DEFINICION DEL PROBLEMA DE INVESTIGACION

PROBLEMA PRINCIPAL

¿Cómo la automatización optimiza el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, 2015?

PROBLEMAS SECUNDARIOS

- a. ¿Cómo diseñar los artefactos para el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga?
- b. ¿Cómo crear un prototipo para el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga?

1.3 OBJETIVOS DE LA INVESTIGACION

OBJETIVO GENERAL

Implementar una aplicación web para optimizar el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, 2015. Mediante la metodología Iconix, un lenguaje de programación orientado a objetos, un sistema gestor de base de datos relacional y tecnologías de Internet, con el propósito de lograr un control integral del proceso de matrícula y con la finalidad de obtener un software que automatice dicho proceso, de manera que sea confiable, rápido y seguro.

OBJETIVOS ESPECIFICOS

- a. Modelar los artefactos adecuadamente para el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga.
- b. Crear un prototipo amigable para el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga.

1.4 JUSTIFICACION Y DELIMITACION DE LA INVESTIGACION

1.4.1. IMPORTANCIA DE LA INVESTIGACION

IMPORTANCIA TECNICA

La aplicación para automatizar el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, se desarrolló utilizando el paradigma de la programación orientada a objetos lo que permite ser reutilizado por otras

universidades que presenten la misma realidad académica, además por ser ejecutado en un entorno web será usado mediante tecnologías de internet que posibilita que su funcionamiento sea continuo.

IMPORTANCIA SOCIAL

Se implementó el aplicativo web con el objetivo de automatizar el proceso de matrícula, debido a que se realiza de una manera deficiente; porque no se ajusta a los requerimientos actuales de la universidad, generando demoras en la matrícula, largas colas por parte de los estudiantes por inconvenientes presentados en el sistema SIMA ; siendo perjudicados directamente.

IMPORTANCIA ECONÓMICA

En el aspecto económico, al implementar este aplicativo web para automatizar el proceso de matrícula reducirá los gastos adicionales generados por un proceso de matrícula deficiente.

1.4.2. JUSTIFICACION

La Universidad Nacional de San Cristóbal de Huamanga requiere contar con una adecuada aplicación web, como herramienta importante para automatizar el proceso de matrícula, y así lograr un control integral de dicho proceso, de manera que sea confiable, rápido y seguro; beneficiando a los estudiantes, docentes y al personal administrativo de la universidad.

1.4.3. DELIMITACION

La presente investigación se realizó para la Escuela de Formación Profesional de Ingeniería de Sistemas de la Universidad Nacional de San Cristóbal de Huamanga, 2015.

CAPITULO II

REVISIÓN DE LITERATURA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

Según Villón (2013), en su tesis titulado “Creación e implementación de un software para la automatización del registro de matrícula de los estudiantes en el Centro de Educación Básica Ignacio Alvarado de la comuna Palmar, Provincia de Santa Elena, período lectivo 2012-2013”, concluye que los recursos tecnológicos son una parte imprescindible en las entidades educativas; ya que la tecnología se ha convertido en un aliado clave para la realización de todo tipo de tareas. Como se puede contrastar con el sistema implementado, el cual permite el control de todos los registros de entrada y salida de la matrícula de cada estudiante de manera eficiente y segura; y brinda la información necesaria a la institución de manera dinámica, rápida, y correcta, mejorando el nivel de atención a los padres de familia de la Institución educativa.

Según Mora y Vega (2013), la finalidad de su investigación fue crear un sistema informático de control de notas y proceso de matriculación para automatizar la información de manera ágil, eficiente y eficaz de los estudiantes de la Unidad Educativa “Wenceslao Ríjavec”, para tal efecto se implementó la metodología del desarrollo de ciclo de vida del modelo incremental. Al realizarse las pruebas de validación, se constató que el sistema optimizó los procesos de control de notas y de matriculación de la UEWR en un 73,11% de tiempo, con referencia a los procesos que se realizaron manualmente. Finalmente la UEWR confirmó la optimización del sistema que le permitió mejorar con eficiencia y eficacia sus procesos educativos.

Según Díaz, Duran y Quijano (2010), en su tesis “Sistema de Información para Gestión Académica”, indican que los planteles educativos como empresas públicas y privadas suelen presentar una gran necesidad de llevar su información de manera ordenada, que les permita obtener datos precisos, concisos y de forma rápida para la toma de decisiones. Y teniendo claro que hoy en día la tecnología nos facilita los procesos, haciéndonos las tareas mucho más fáciles y en menor tiempo posible, han implementado una herramienta

que permite sistematizar los procesos académicos que se lleva a cabo en un plantel educativo de educación básica y media técnica para facilitar las labores de los rectores, coordinadores y docentes que son los principales beneficiarios con el nuevo sistema de información para la gestión académica (SIGA).

Según Lázaro (2006), en su tesis “Sistematización de un proceso de matrícula”, indica que el enfoque sistémico tiene por objetivo representar, de forma comprensiva y objetiva, el medio en el que tiene lugar la toma de decisiones; la cual será mucho más fácil de tomar, si se cuenta con una descripción concreta y objetiva del sistema dentro del cual debe ser tomada. Siguiendo con esta ideología se prosiguió a implementar un software para la sistematización de la matrícula y control académico compatible en todas las facultades de la Universidad Nacional Santiago Antúnez de Mayolo; permitiendo así mejorar la Toma de Decisiones dentro de la institución, dando información requerida de manera oportuna para los usuarios del sistema de información.

2.2. MARCO TEÓRICO

2.2.1. AUTOMATIZACIÓN

Automatización se refiere a mejorar y simplificar procesos, integrando procesos internos para ahorrar tiempo, dinero; acceder y recuperar información (Soares, 2008).

Automatización corresponde a la implementación de una tecnología para apoyar la realización de las actividades de un proceso. Esta puede llevarse a cabo cuando existe un proceso en el que participan varias personas con tareas y actividades bien definidas, que deben ser realizadas en un orden específico y que típicamente se apoyan en documentos/información que ingresa, se crea y/o se transforma (Zapata, 2008).

2.2.1.1. ARTEFACTO

Un artefacto es un término general que se utiliza para describir cualquier pieza de información usada o producida al desarrollar sistemas (Kendall, 2005).

Es una pieza de información que es usada o producida mediante un proceso de desarrollo de software. Un artefacto puede ser un documento, un modelo o un elemento de un modelo (Pressman, 2010).

2.2.1.2. PROTOTIPO

Modelo o maqueta del sistema que se construye para comprender mejor el problema y sus posibles soluciones (Kendall, 2005).

El prototipo es el modelo que se desarrolla de un software para reflejar cómo se comporta un sistema. Estos prototipos se utilizan para comprender cómo funciona el sistema en cuestión (Pressman, 2010).

2.2.2. PROCESO DE MATRÍCULA

“La matrícula es un acto formal, por el cual el estudiante se compromete voluntariamente a llevar determinadas asignaturas en un semestre académico. La matrícula se registra en una ficha única y se realiza en la Secretaría de la Escuela de Formación Profesional correspondiente” (Estatuto Reformulado de la UNSCH, 2015, p. 39).

“La matrícula es un acto académico – administrativo, por el cual el estudiante confirma o renueva en cada semestre académico su condición de tal y se compromete a cumplir las disposiciones legales que rigen la vida universitaria” (Reglamento Interno de la UNSCH, 2011, p. 14).

2.2.2.1. FICHA DE MATRÍCULA

La Ficha de Matrícula como instrumento técnico pedagógico es el único documento que acredita la condición de estudiante. Tiene naturaleza de Declaración Jurada (Estatuto Reformulado de la UNSCH, 2015).

Es el documento donde el estudiante, registra bajo exclusiva responsabilidad las asignaturas, prácticas o actividades cocurriculares que se deseen cursar en el semestre (Reglamento Interno de la UNSCH, 2011).

2.2.2.2. REGISTRO DE EVALUACIÓN

El Registro de Evaluación es un instrumento de trabajo que sirve para controlar y verificar la labor de enseñanza de los docentes (Reglamento Interno de la UNSCH, 2011).

Es una plantilla, que sirve para registrar los resultados obtenidos por el conjunto de alumnos que conforman un grupo, de forma clara, completa, eficaz y lo más fiable posible, respecto a los indicadores de evaluación establecidos para cada una de las unidades didácticas o temas, de un área o materia dentro de un curso académico (García y Medina, 1987).

2.2.2.3. ACTA DE EVALUACIÓN FINAL

La Acta de Evaluación Final es el documento oficial, en que queda constancia de la calificación final obtenida por los estudiantes en un asignatura o curso académico (Estatuto Reformulado de la UNSCH, 2015).

Además, junto a la calificación letrada se consigna la palabra “aprobado” o “desaprobado”. Las actas no deben tener manchas ni enmendaduras. Estas se remiten y distribuyen en los plazos que señala el Calendario de Actividades Académicas de la Universidad, bajo la responsabilidad de los jefes de las respectivas dependencias; y su inscripción es normada por la Oficina de Secretaría General (Reglamento Interno de la UNSCH, 2011).

2.2.2.4. ESQUELA DE PAGOS

Es el documento donde se registra los conceptos de pago de matrícula, de los estudiantes para cada semestre académico (Reglamento Interno de la UNSCH, 2011).

2.2.3. METODOLOGÍA ICONIX

La metodología ICONIX se encuentra entre el Rational Unified Process (RUP) y la Programación Extrema (XP). ICONIX está conducido por casos de uso al igual que el RUP, pero sin la sobrecarga del mismo. Es relativamente pequeño y ligero, igual que XP, pero no descarta el análisis y diseño formal como la XP. ICONIX usa racionalmente el lenguaje unificado de modelado (UML), haciendo referencia a la trazabilidad de los requisitos (Rosenberg et al., 2005).

El proceso ICONIX se divide en flujos de trabajos dinámicos y estáticos, que son altamente repetitivos; usted puede ir a través de una repetición de todo el proceso para una pequeña cantidad de casos de uso (quizás un par de paquetes de valor, que no es una

cantidad enorme teniendo en cuenta que cada caso de uso es sólo un par de párrafos), todo el camino a la fuente de código y pruebas unitarias. Por esta razón, el proceso ICONIX es muy adecuado para proyectos ágiles, donde se necesita información rápida sobre factores tales como los requisitos, el diseño, y las estimaciones (Rosenberg y Stephens, 2007).

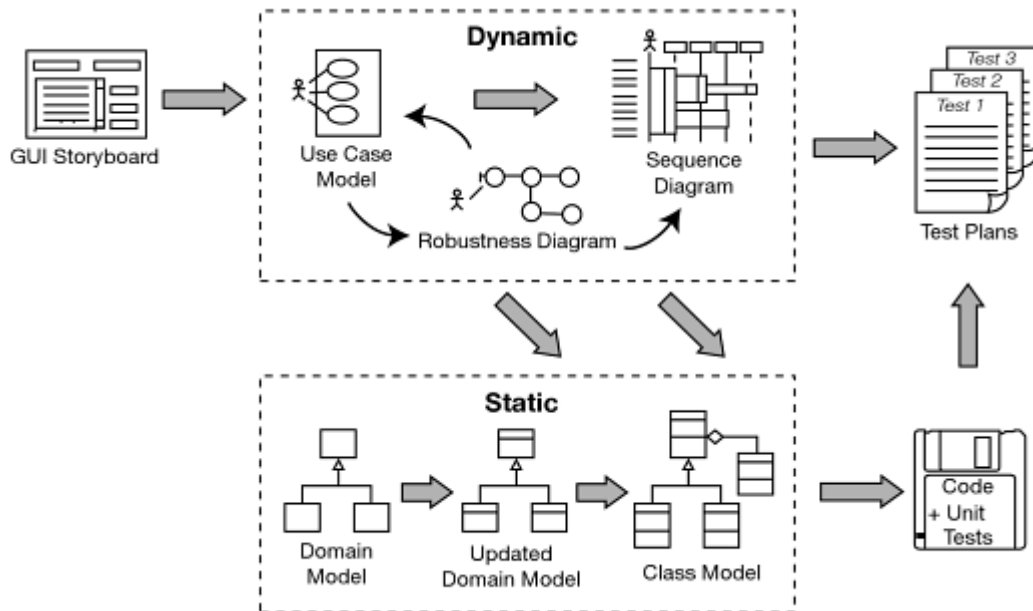


Figura N° 2.1: Flujo de trabajo Iconix (Rosenberg y Stephens, 2007)

2.2.3.1 TÉCNICAS DE LA METODOLOGÍA ICONIX

A. ANÁLISIS DE REQUISITOS

A.1. MODELO DE DOMINIO

El Modelo de Dominio es un artefacto colaborativo vivo. Es refinado y actualizado en cada parte del proyecto, de modo que refleja siempre la comprensión actual del espacio del problema (Rosenberg y Stephens, 2007).

Es la tarea de descubrir objetos (clases), que representan cosas y conceptos del “mundo real”. Según ICONIX, el modelado de dominio inicial abarca palabras externas de los requisitos para construir el modelo estático (Rosenberg et al., 2005).

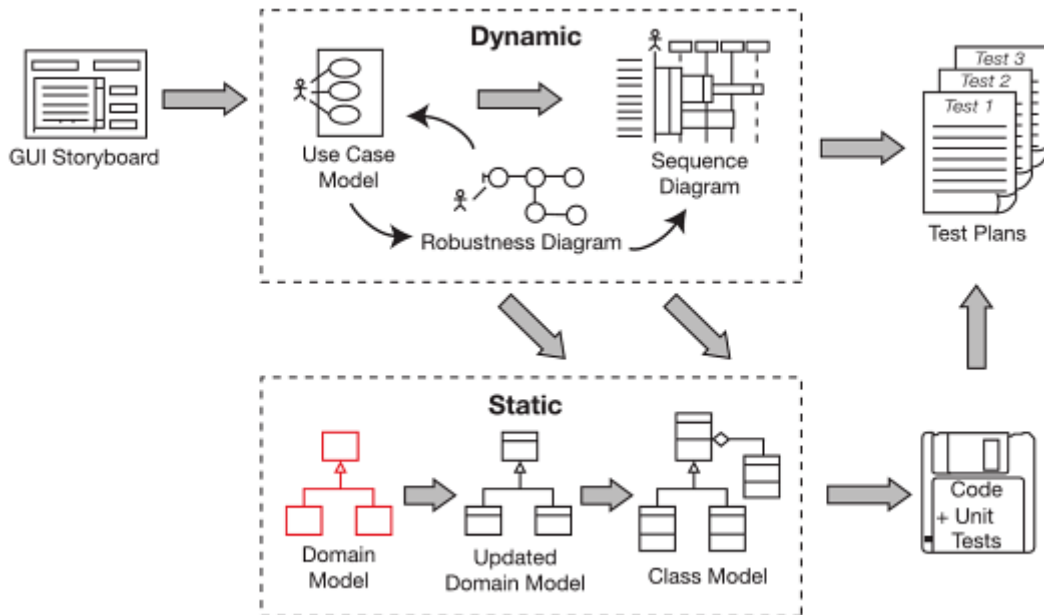


Figura N° 2.2: Modelo de Dominio (Rosenberg y Stephens, 2007)

A.2. MODELO DE CASO DE USOS

Con un primer modelo de dominio inicial en su lugar, es hora de comenzar a escribir los casos de uso. Los casos de uso dan un modo estructurado de capturar los requisitos de comportamiento de un sistema, de modo que puede razonablemente crear un diseño desde ellos. Le ayudan a responder ciertas preguntas fundamentales: ¿Qué están tratando de hacer los usuarios del sistema? ¿Cuál es la experiencia del usuario? Una cantidad sorprendente de lo que su software debe hacer se dicta por el modo en que los usuarios deben interactuar con él. (Rosenberg y Stephens, 2007).

Según Rosenberg et al. (2005), los modelos de casos de uso son desarrollados en cooperación con el modelo de dominio; todos los casos de uso sirven de base para pruebas de funcionalidades durante la fase de implementación; un caso de uso es una secuencia de acciones que un actor realiza en el sistema para alcanzar un objetivo.

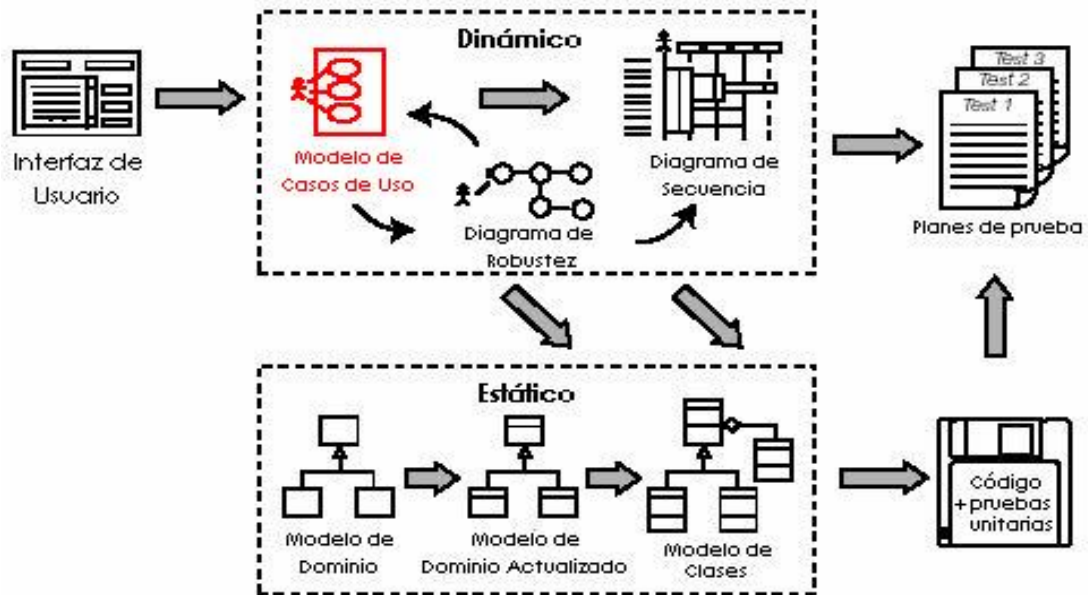


Figura N° 2.3: Modelo de Casos de uso (Rosenberg y Stephens, 2007)

B. REVISIÓN DE REQUISITOS

La revisión de requisitos garantiza que el sistema tal y como se describe coincida con los requisitos. Se trata de un período de sesiones de colaboración que impliquen al representante(s) del cliente, los usuarios finales (es decir, las personas que realmente van a utilizar el sistema, o quien está usando el sistema actual que se sustituirá), y las personas de marketing—básicamente, todos los stakeholders que tienen un interés en asegurar que los requisitos encajen con su punto de vista del sistema (Rosenberg y Stephens, 2007).

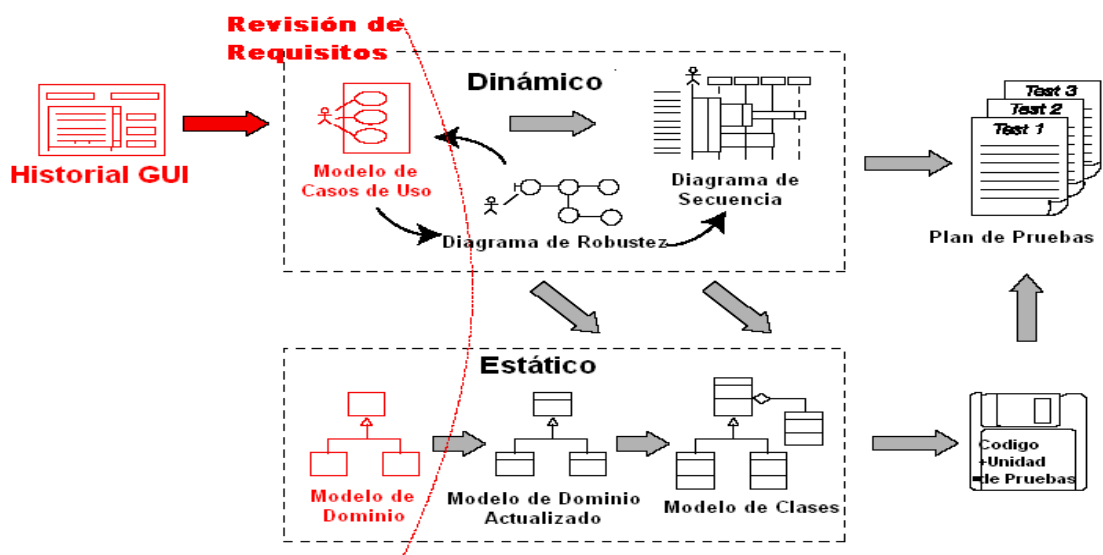


Figura N° 2.4: Revisión de requisitos (Rosenberg y Stephens, 2007)

C. DISEÑO PRELIMINAR

C.1. ANÁLISIS DE ROBUSTEZ

Para obtener a partir de los casos de uso un diseño detallado (y luego el código), lo que se necesita es enlazar los casos de uso a los objetos. El análisis de robustez, le ayuda a superar la brecha que existe del análisis al diseño. Es una manera de analizar sus casos de uso e identificar un primer conjunto de objetos para cada caso de uso. Estos se clasifican en objetos interfaz, objetos entidad, y controladores (que son a menudo más como funciones que como objetos) (Rosenberg y Stephens, 2007).

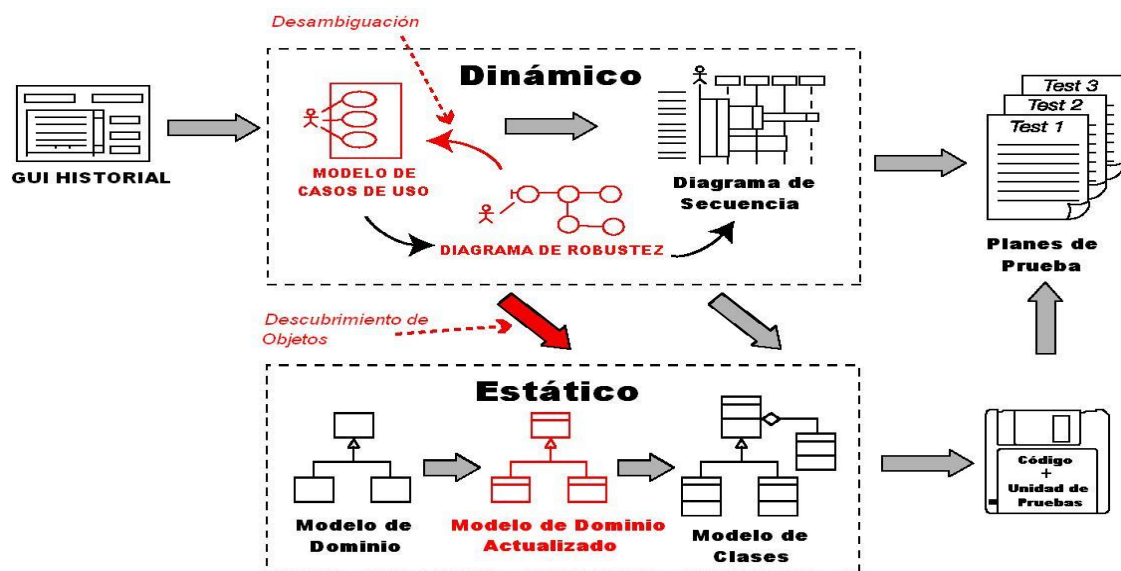


Figura N° 2.5: Análisis de Robustez (Rosenberg y Stephens, 2007)

C.2. MODELO DE DOMINIO ACTUALIZADO

El Modelo de Dominio Actualizado se realiza al hacer el diagrama de robustez, es un modelo de dominio con nuevas clases y atributos (Rosenberg et al., 2005).

D. REVISIÓN DEL DISEÑO PRELIMINAR

La Revisión del Diseño Preliminar (RDP) ayuda a asegurarse que los diagramas de robustez, el modelo de dominio, y la descripción de casos de uso coincidan entre sí. Esta revisión es el "puente" entre el diseño preliminar y en las etapas del diseño detallado, para cada paquete de casos de uso (Rosenberg y Stephens, 2007).

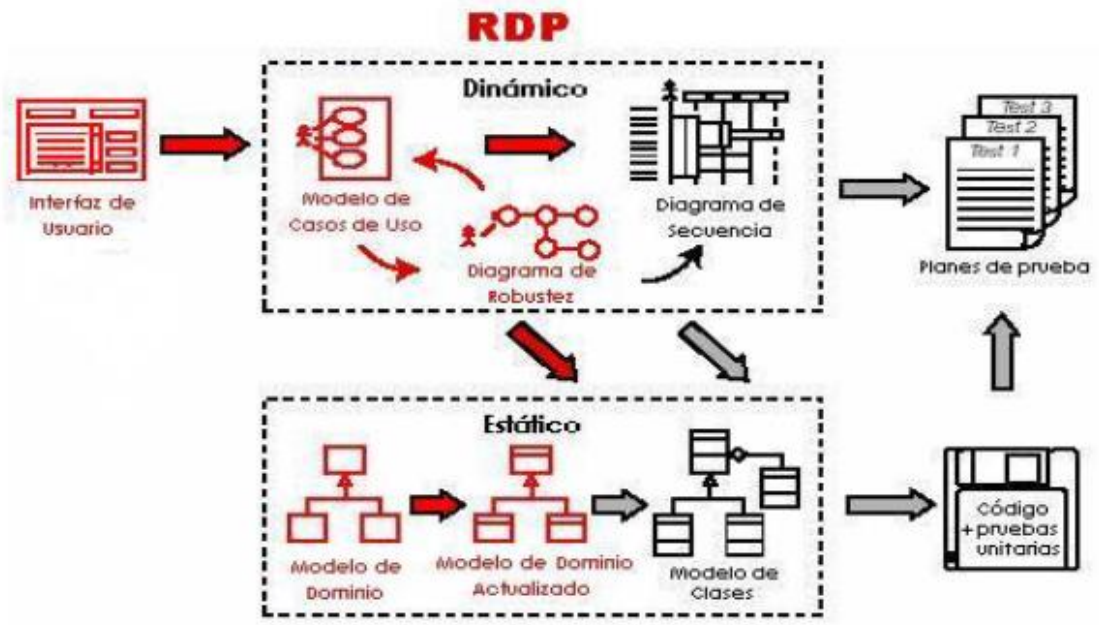


Figura N° 2.6: Revisión del Diseño Preliminar (Rosenberg y Stephens, 2007)

E. ARQUITECTURA TÉCNICA

El objetivo de arquitectura técnica (AT) es obtener un sentido general del sistema que vas a desarrollar. Es necesario utilizar un framework de aplicación específico (por ejemplo, un framework de una compañía estándar). No hay una notación estándar o un formato para documentar la AT, la profundidad y el formato de la arquitectura técnica y los convenios para crearla varían mucho de empresa a empresa, por lo que no insistiremos en esta área demasiado tiempo (Rosenberg y Stephens, 2007).

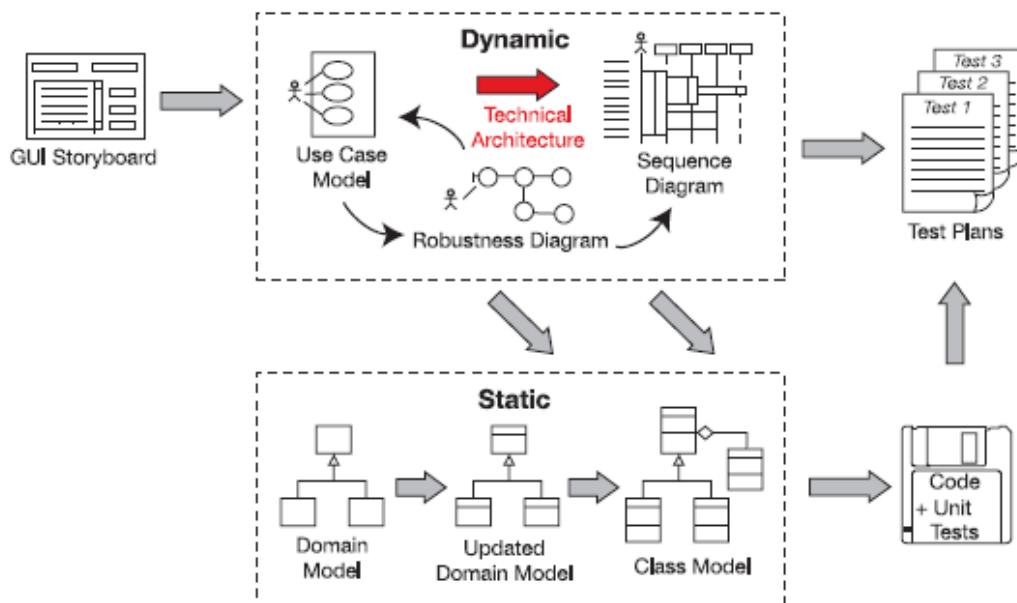


Figura N° 2.7: Arquitectura Técnica (Rosenberg y Stephens, 2007)

F. DISEÑO DETALLADO

F.1. DIAGRAMAS DE SECUENCIA

Una vez que se ha finalizado el análisis de robustez, y ha celebrado una Revisión del Diseño Preliminar, es tiempo de iniciar el diseño detallado (asignación de comportamiento). En este momento, la descripción de sus casos de uso debe ser completa, correcta, detallada y explícita; además debe tener descubiertas las clases de dominio que va a necesitar y haber diseñado la arquitectura técnica (Rosenberg y Stephens, 2007).

Según Rosenberg et al. (2005), el comportamiento de un caso de uso se detalla mediante el diagrama de secuencia, éste muestra la colaboración dinámica entre los objetos del sistema, también se observa la secuencia de mensajes enviados entre los objetos. El diagrama de secuencia, permite asignar operaciones a los objetos interfaz y entidad, porque los comportamientos (verbos) del diagrama de robustez, se convierten en mensajes (operaciones) entre los objetos interfaz y entidad (nombres). En conclusión, actualizamos el modelo estático con las operaciones y los atributos identificados en las fases anteriores.

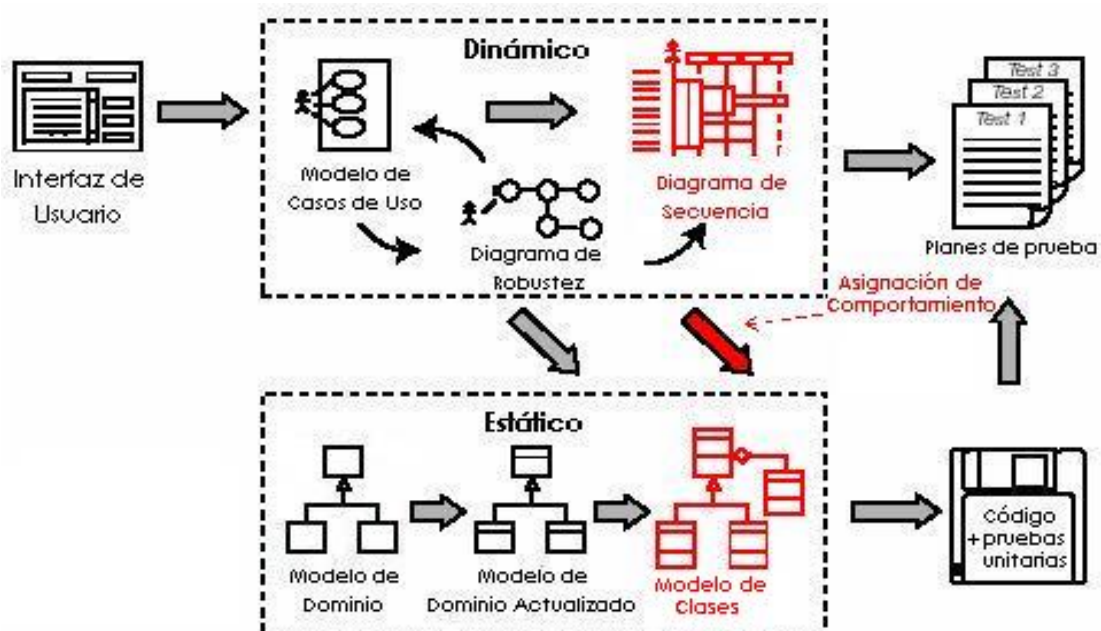


Figura N° 2.8: Diagramas de Secuencia (Rosenberg y Stephens, 2007)

F.2. DISEÑO DETALLADO

Se actualiza el diagrama de clases usando el diagrama de secuencia, adicionar operaciones y todos los atributos, visibilidad y navegabilidad a las clases (Rosenberg et al., 2005).

G. REVISIÓN CRÍTICA DEL DISEÑO

Según Rosenberg y Stephens (2007), la revisión crítica del diseño es un paso entre el diseño y la implementación, en el que se usa tres criterios: hacer coincidir la descripción de los casos de uso con los diagramas de secuencia, verificar la continuidad de los mensajes y revisarlo para un buen diseño. Si se diseña cada caso de uso, aplicando los tres criterios; entonces el diseño está listo para la codificación.

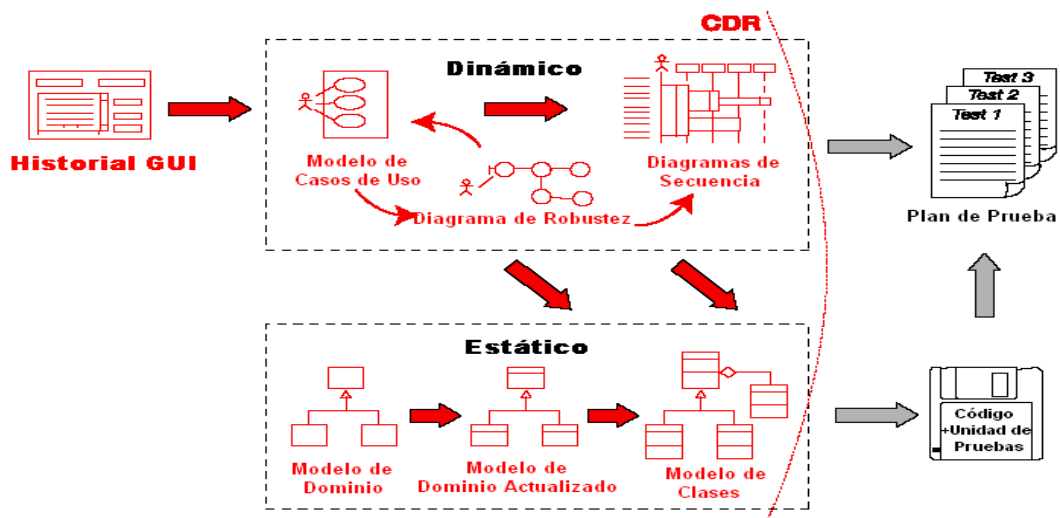


Figura N° 2.9: Revisión Crítica del Diseño (Rosenberg y Stephens, 2007)

H. IMPLEMENTACIÓN

Si usted ha pasado por todo el esfuerzo para crear un diseño agradable y detallado, vale la pena tener una buena idea de cómo traducir ese diseño detallado en el código fuente y la pruebas unitarias (Rosenberg y Stephens, 2007).

Según Rosenberg et al. (2005), la implementación se realiza utilizando los diagramas de secuencia y el diagrama de clases. Para lo cual, se muestra un método sistemático para convertir el diseño en código fuente, a su vez en la práctica se debe escribir las pruebas unitarias para los controladores. Si el diseño presenta errores al momento de codificarlo, se debe corregir.

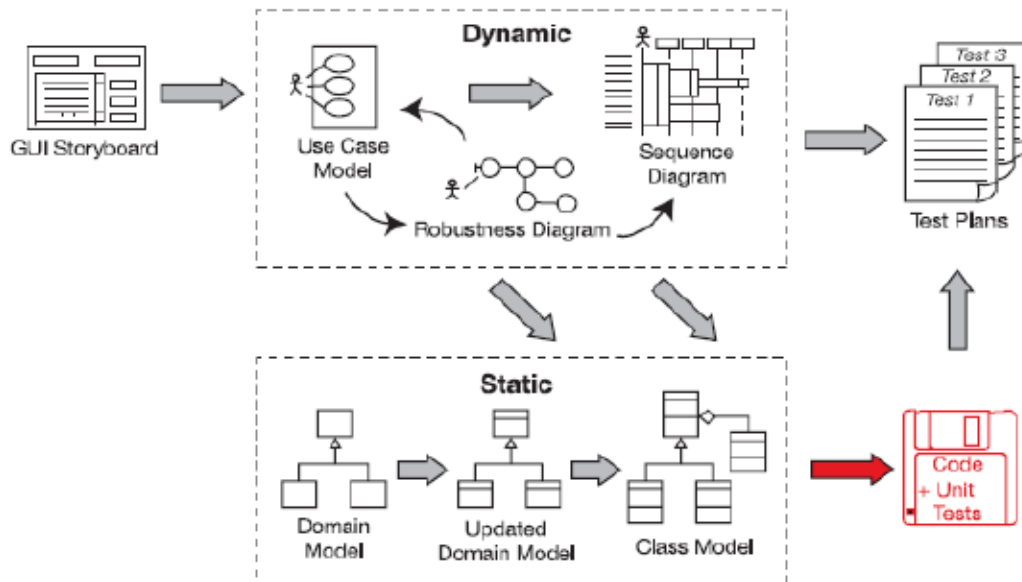


Figura N° 2.10: Implementación (Rosenberg y Stephens, 2007)

2.2.4. APLICACIÓN WEB

Una aplicación web (web-based application) es un tipo especial de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones (Luján, 2002).

Según Aumaille (2002), una aplicación web es un conjunto de recursos web que participan en el funcionamiento de la propia aplicación Web. Una aplicación Web está compuesta: a) Componentes de servidor dinámicos; Servlets, JSPs, b) Bibliotecas de clases Java utilitarias, c) Elementos web estáticos: páginas HTML, imágenes, sonidos, d) Componentes dinámicos: Applets, JavaBeans y clases, e) Un descriptor de desarrollo y de configuración de la aplicación web, en forma de uno o múltiples archivos en formato XML.

Según Seoane (2005), una aplicación web es un programa especialmente diseñado para ejecutarse dentro de un navegador web. Para ello se emplean tecnologías de tres capas, basándose en una arquitectura cliente-servidor; a) La primera capa reside en el ordenador del usuario, en el que se ejecutará la aplicación dentro del navegador web, se ocupa de la representación y obtención de datos, la generación de informes, gráficos, b) La segunda capa reside en el servidor de la lógica del negocio, que reside en el servidor, que además

de preparar el entorno en el que se presenta la aplicación, se ocupa del procedimiento real de los datos, también es conocido como middleware, c) La tercera capa reside en el servidor de base de datos de la empresa, donde el servidor se ocupa de procesar las consultas que se efectúan desde el servidor de la lógica del negocio, de esta forma, devuelve los datos solicitados, disponiendo de módulos para crear y gestionar las bases de datos y los usuarios de las mismas.

2.2.5. PROGRAMACION ORIENTADA A OBJETOS

La Programación Orientada a Objetos es un paradigma o modelo de programación. Esto significa que no es un lenguaje específico, o una tecnología, sino una forma de programar, una manera de plantearse la programación para la construcción de sistemas. No es la única, pero se ha constituido en una de las formas de programar más populares e incluso muchos de los lenguajes que usamos hoy día lo soportan o están diseñados bajo este modelo (Joyanes, 2008).

Según Rodríguez (2003), define que la Programación Orientada a Objeto, estructura un programa dividiéndolo en una cantidad determinada de objetos de alto nivel. Cada objeto le da forma a algún aspecto del problema que se intenta resolver. Menciona también que una secuencia de llamadas a procedimientos para controlar el flujo del programa, ya no es el enfoque principal de la programación orientada a objetos. En vez de ello, los objetos interactúan entre sí para dar lugar al flujo total del programa. De cierta forma, un programa orientado a objetos se convierte en una simulación real del problema que se intenta resolver.

Según Ceballos (2000), las características fundamentales de la Programación Orientada a objetos son: herencia, polimorfismo, abstracción y encapsulamiento.

“En la programación orientada a objetos se usa un mecanismo llamado herencia para diseñar dos o más entidades que son diferentes pero que comparten muchas características comunes de las entidades” (Wu, 2008, p. 23).

La clase de quien se hereda se conoce como clase padre o clase ascendente o superclase, mientras que la clase que hereda se conoce como clase hija o clase descendiente o

subclase. Esta clase hija a su vez puede convertirse en padre y así sucesivamente. A ésta descendencia se conoce como jerarquía de clases (Vásquez y Balta, 2008).

El término polimorfismo expresa la posibilidad de que el mismo mensaje, enviado a objetos distintos, ejecute métodos distintos. Esto significa que podemos definir dentro de dos clases distintas dos operaciones con el mismo nombre y aspecto externo, pero con distintas implementaciones para cada clase (Noriega, 2007).

Según Ceballos (2000), la abstracción consiste en la generalización conceptual de los atributos y propiedades de un determinado conjunto de objetos. Precisamente la clave de la programación orientada a objetos está en abstraer los métodos y los datos comunes a un conjunto de objetos y almacenarlos en una clase.

El encapsulamiento es como un envoltorio protector alrededor del código y los datos que se manipulan. El envoltorio define el comportamiento y protege el código y los datos para evitar que otro código acceda a ellos de manera arbitraria (Naughton, 2000).

2.2.6. SISTEMA GESTOR DE BASE DE DATOS

No debemos confundir una Base de Datos con un Sistema Gestor de Base de Datos. Una Base de datos es la información almacenada, que cumple una serie de características y restricciones, pero para que la información pueda ser almacenada y el acceso a la misma satisfaga las características exigidas a una base de datos, es necesario que exista una serie de procedimientos, un sistema software, que sea capaz de llevar a cabo tal labor. A este sistema de software es lo que llamamos Sistema Gestor de Base de Datos (SGBD) (Nevado, s.f).

“Los Sistemas de Gestión de Base de Datos (DBMS) son paquetes de software para la gestión de las bases de datos; en particular, para almacenar, manipular y recuperar datos en un computador” (Batini, Ceri y Navathe, 2000, p. 4).

Un SGBD permite: Definir los datos a los distintos niveles de abstracción (físico, lógico y externo), Manipulación de los datos en la base de datos, permitiendo insertar, modificar, borrar y consultar datos, Mantenimiento de la integridad de la base de datos, Control de la privacidad y seguridad de los datos en la base de datos (Nevado, s.f).

Es un módulo de programa que constituye la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de aplicaciones y las consultas hechas al sistema, teniendo como tareas básicas la interacción con el manejador de archivos, implantación de la integridad, puesta en práctica de la seguridad, respaldo y recuperación y control de concurrencia (Osorio, 2008).

Una de las principales razones para usar un SGBD es tener un control centralizado tanto de los datos como de los programas que acceden a esos datos. La persona que tiene este control central sobre el sistema se llama administrador de base de datos (Silberschatz, Korth y Sudarshan, 2001).

Las principales funciones de SGBD es garantizar el acceso, la recuperación y actualización de los datos en la base de datos. El SGBD proporciona los mecanismos físicos que permiten a varios usuarios tener acceso de forma rápida y eficiente a diferentes datos relacionados (Hansler y Hansen, 2000).

2.2.7. TECNOLOGIAS DE INTERNET

Internet es un conjunto de niveles de redes dispersas, que entre todas ellas conectan a millones de ordenadores, cuyos usuarios pueden intercambiar recursos informáticos, independientemente del ordenador que usen. Internet no es un sistema centralizado, no es una red, sino “red de redes”. Estas redes se conectan mediante líneas telefónicas convencionales, líneas de datos de alta velocidad, satélites, enlaces de microondas y fibra óptica (Jarabo y Elortegui, 2000).

Según Stallings (2009), internet es una red pública porque puede ser utilizada por cualquier persona, este servicio puede ser provisto por proveedores de servicios; entonces teniendo en cuenta que internet es un ambiente de trabajo sobre el cual debemos asumir, que no tenemos control alguno de quienes y como lo utilizan.

Entonces definiré a internet como un conjunto de redes interconectadas, que hacen uso de protocolos en especial del TCP/IP, de esta manera cada red se convierte en parte de otra red para poder conseguir una plataforma conjunta de servicios e interconexión para transferencia de datos, en la actualidad su lenguaje preferido el HTML tiene una nueva versión que es HTML 5.

CAPITULO III

METODOLOGÍA DE LA INVESTIGACIÓN

3.1. TIPO DE INVESTIGACIÓN

Según Caballero (2008), considera los tipos de investigación según el propósito, en investigación teórica y básica; investigación aplicada y tecnológica. La investigación aplicada “Cuyo propósito fundamental es dar solución a problemas prácticos”. De acuerdo a esta premisa se identifica que el tipo de investigación es Aplicada.

3.2. DISEÑO DE LA INVESTIGACIÓN

Según Lozano (2011), el diseño de la investigación señala al investigador lo que debe hacer para alcanzar sus objetivos de estudio y para contestar las interrogantes de conocimiento que se ha planteado, se recogen los datos que consideramos necesarios.

Según Hernández et al. (2010), la investigación no experimental es aquella que se realiza sin manipular deliberadamente las variables; es decir, se trata de estudios donde no hacemos variar en forma intencional las variables independientes para ver su efectos en otras variables. Simplemente se observan los fenómenos tal como se dan en su contexto natural, para posteriormente analizarlos.

Según Gómez (2006), Los diseños de investigación transaccional o transversal recolectan datos en un solo momento, en un tiempo único. Su propósito es describir variables y analizar su incidencia e interrelación en un momento dado. Es como tomar una fotografía de algo que sucede.

Según Hernández et al. (2010), Los estudios descriptivos buscan especificar las propiedades, las características y los perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis. Es decir, únicamente pretenden medir o recoger información de manera independiente o conjunta sobre los conceptos o las variables a las que se refieren, esto es, su objetivo no es indicar cómo se relacionan éstas.

“La investigación descriptiva es uno de los tipos o procedimientos investigativos más populares y utilizados por los principiantes en la actividad investigativa. Los trabajos de grado, en los pregrados y en muchas de las maestrías, son estudios de carácter eminentemente descriptivo. En tales estudios se muestran, narran, reseñan o identifican hechos, situaciones, rasgos, características de un objeto de estudio, o se diseñan productos, modelos, prototipos, guías, etcétera. Pero no se dan explicaciones o razones del porqué de las situaciones, los hechos, los fenómenos, etcétera” (Bernal, 2006, p.112).

“La investigación descriptiva se soporta principalmente en técnicas como la encuesta, la entrevista, la observación y la revisión documental” (Bernal, 2006, p. 113).

Según estos puntos citados, el diseño de la investigación es no experimental – transversal - descriptivo.

3.3. POBLACION Y MUESTRA

POBLACIÓN

La población está compuesta por todo el proceso de matrícula de todas las Escuelas de Formación Profesional de la Universidad Nacional de San Cristóbal de Huamanga, en el año 2015.

MUESTRA

Se tomó una muestra no probabilística por juicio de valor, conformada por el proceso de matrícula de la Escuela de Formación Profesional de Ingeniería de Sistemas de la Universidad Nacional de San Cristóbal de Huamanga, en el año 2015.

3.4. VARIABLES

3.4.1. DEFINICION CONCEPTUAL DE LAS VARIABLES

VARIABLE 1

Automatización.- Se refiere a mejorar y simplificar procesos, integrando procesos internos para ahorrar tiempo, dinero; acceder y recuperar información.

DIMENSIONES DE LA VARIABLE 1

Artefacto.- Es una pieza de información que es usada o producida mediante un proceso de desarrollo de software. Un artefacto puede ser un documento, un modelo o un elemento de modelo.

Prototipo.- Es el modelo que se desarrolla de un software para reflejar cómo se comporta un sistema. Estos prototipos se utilizan para comprender cómo funciona el sistema en cuestión.

VARIABLE 2

Proceso de Matrícula.- Es un acto formal, por el cual el estudiante se compromete voluntariamente a llevar determinadas asignaturas en un semestre académico y cumplir las disposiciones legales que rigen la vida universitaria. La matrícula se registra en una ficha única y se realiza en la Secretaría de la Escuela de Formación Profesional correspondiente.

DIMENSIONES DE LA VARIABLE 2

Ficha de Matrícula.- Es el único documento que acredita la condición de estudiante, en éste se registra bajo exclusiva responsabilidad las asignaturas, prácticas o actividades cocurriculares que se deseen cursar en el semestre. Tiene naturaleza de Declaración Jurada.

Registro de Evaluación.- Es un instrumento de trabajo que sirve para controlar y verificar la labor de enseñanza de los docentes, en esta plantilla se registra los resultados obtenidos por el conjunto de alumnos que conforman un grupo, respecto a los indicadores de evaluación establecidos para cada una de las unidades didácticas o temas, de una materia.

Acta de Evaluación Final.- Es el documento oficial, en que queda constancia de la calificación final obtenida por los estudiantes en una asignatura o curso académico. Estas se remiten y distribuyen en los plazos que señala el calendario de actividades académicas de la Universidad, bajo la responsabilidad de los jefes de las respectivas dependencias; y su inscripción es normada por la Oficina de Secretaría General.

Esquela de Pagos.- Es el documento donde se registra los conceptos de pago de matrícula, de los estudiantes para cada semestre académico.

3.4.2. DEFINICION OPERACIONAL DE LAS VARIABLES

VARIABLE 1

X: Automatización

DIMENSIONES DE LA VARIABLE 1

X1: Artefacto

X2: Prototipo

VARIABLE 2

Y: Proceso de Matrícula

DIMENSIONES DE LA VARIABLE 2

Y1: Ficha de Matrícula

Y2: Registro de Evaluación

Y3: Acta de Evaluación Final

Y4: Esquela de Pagos

3.5. TÉCNICAS E INSTRUMENTOS

Se usó la técnica del análisis documental para la recolección de datos sobre el Proceso de Matrícula, en el que se analizará el “Estatuto Reformulado de la UNSCH” y “El Reglamento Interno de la UNSCH”, desarrollado por la misma universidad.

Asimismo, se utilizó la técnica de la observación para registrar detalladamente las distintas etapas del Proceso de Matrícula de la Universidad Nacional de San Cristóbal de Huamanga.

3.5.1. INSTRUMENTOS PARA RECOLECTAR INFORMACION

Los instrumentos para recolectar la información de la presente investigación se detallan a continuación:

Se aplicó la Técnica de Análisis Documental a los documentos que referencian el Proceso de Matrícula denominada “Ficha de Análisis Documental del Proceso de Matrícula” presentado en el anexo A, tabla N° A.1. Muestra la información documental recolectada sobre las etapas del Proceso de Matrícula de la Universidad Nacional de San Cristóbal de Huamanga.

Se aplicó la Técnica de la Observación al Proceso de Matrícula denominada “Ficha de observación del Proceso de Matrícula” presentado en el anexo B, tabla N° B.1. Muestra la información registrada de las distintas etapas del Proceso de Matrícula de la Universidad Nacional de San Cristóbal de Huamanga.

3.5.2. HERRAMIENTAS PARA EL TRATAMIENTO DE DATOS E INFORMACION

Las herramientas tecnológicas que se utilizó fueron seleccionadas en función a las limitaciones como; recursos económicos públicos, recursos humanos limitados y por ser una aplicación pequeña.

SOFTWARE	FABRICANTE	SERVICIO
WINDOWS 8	Producida por Microsoft Corporation	Es una versión de Microsoft Windows, producido para su uso en computadoras personales, incluidas computadoras de escritorio en casa y de negocios, computadoras portátiles, notebooks, tabletas, servidores y centros multimedia.
LENGUAJE JAVA	Creado por James Gosling de Sun Microsystems, hoy es Administrado por ORACLE	Es un lenguaje de programación orientado a objetos, que deriva en gran medida su sintaxis de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera de ellos, y posee un sistema que interpreta y ejecuta los archivos para ser compilados en tiempo real.

SOFTWARE	FABRICANTE	SERVICIO
IDE NETBEANS	ORGANIZACIÓN NETBEANS	Es un entorno de desarrollo integrado, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. El IDE NetBeans es un producto libre y gratuito sin restricciones de uso.
BOOTSTRAP	Mark Otto y Jacobd Thornton de Twitter	Es un framework o conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web. Un entorno de desarrollo con una serie de recursos que simplifican el desarrollo de un proyecto web con html5, css3 y JQuery, de manera que simplifica mucho el trabajo a la hora de diseñar, ya que el framework bootstrap ya tiene una buena parte del trabajo hecho lo cual simplifica mucho la tarea del desarrollo.
ORACLE DATABASE	Oracle Corporation Rod Johnson	Es un sistema de gestión de base de datos de tipo objeto-relacional. Se le considera uno de los sistemas de bases de datos más completos, destacando: soporte de transacciones, estabilidad, escalabilidad y soporte mutiplataforma.

Tabla N° 3.1: Herramientas tecnológicas para tratamiento de datos (Elaboración Propia)

3.5.3. TECNICAS PARA APLICAR ICONIX

Revisado el marco teórico desarrollado en el capítulo II, sección 2.2.3, formulamos el proceso, que considera las fases para desarrollar la aplicación web usando la metodología Iconix, como se muestra en las tablas 3.2 a 3.9.

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Identificar requisitos	<ul style="list-style-type: none"> Requisitos funcionales y no funcionales. Casos de prueba de aceptación. 	<ul style="list-style-type: none"> Entrevistas. Definir lo que el sistema de hacer. Escribir al menos un caso de prueba para cada requisito. 	Usuario Cliente Analista
Identificar objetos del mundo real y dibujar modelo de dominio	Modelo de dominio	<ul style="list-style-type: none"> Identificar clases claves del negocio. Identificar sustantivos y depurar. Identificar objetos en requisitos funcionales y asignar al modelo de dominio. Utilizar agregación y generalización. 	Analista
Asignar requisitos funcionales a los casos de uso	Relación entre requisitos funcionales y casos de uso	<ul style="list-style-type: none"> Asignar requisitos funcionales a los casos de uso. 	
Descubrir casos de uso	Lista de casos de uso	<ul style="list-style-type: none"> Utilizar requisitos funcionales. Entrevistas. 	Usuario Cliente Analista
Dibujar y empaquetar casos de uso	<ul style="list-style-type: none"> Diagrama de casos de uso Paquete de casos de uso 	<ul style="list-style-type: none"> Identificar roles y responsabilidades de actores. Asociar actores con casos de uso. Relacionar casos de uso. Agrupar lógicamente casos de uso. 	Analista
Realizar prototipo de interfaz gráfica	Prototipo GUI	<ul style="list-style-type: none"> Utilizar historia de eventos del usuario (storyboards). Utilizar los requisitos funcionales. Diseñar interfaz gráfica básica. 	Programador Analista
Escribir el primer borrador de caso de uso	Primer borrador de caso de uso	<ul style="list-style-type: none"> Utilizar glosario de objetos del modelo de dominio. Utilizar la regla de dos párrafos. Escribir el caso de uso como flujo de eventos/respuesta. 	Analista

		<ul style="list-style-type: none"> • Escribir el caso de uso con estructura sustantivo-verbo-sustantivo. • Escribir caso de uso en voz activa. • Referenciar por su nombre las pantallas. 	
--	--	--	--

Tabla N° 3.2: Análisis de Requisitos (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Revisar el modelo de dominio	Modelo de dominio	<ul style="list-style-type: none"> • Identificar al menos 80% de clases clave de dominio del problema. 	Usuario Cliente Analista Programador
Revisar el prototipo GUI	Prototipo GUI	<ul style="list-style-type: none"> • Diseñar con precisión la GUI relacionada al caso de uso. 	
Revisar la descripción de casos de uso	Caso de uso revisado	<ul style="list-style-type: none"> • Eliminar clases fuera del dominio del problema. • Cambiar descripción de voz pasiva activa. • Describir todos los cursos alternos. • Asociar todos los requisitos a los casos de uso. • Describir que intenta hacer el usuario para cada caso de uso. 	

Tabla N° 3.3: Revisión de requisitos (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Rescribir el primer borrador para cada caso de uso	Caso de uso desambiguado	<ul style="list-style-type: none"> • Rescribir el caso de uso durante el análisis de robustez. 	Analista
Identificar el primer corte de objetos que completan escenarios para cada caso de uso	Diagrama de robustez	<ul style="list-style-type: none"> • Copiar la descripción del caso de uso en el diagrama de robustez. • Usar las clases del modelo de dominio. • Crear un objeto interfaz por cada GUI y nombrarlo. • Transformar verbos del caso de uso en controladores. 	

		<ul style="list-style-type: none"> • Relacionar un caso de uso al diagrama de robustez cuando es invocado. • Utilizar las reglas para construir el diagrama de robustez. 	
Actualizar el modelo de dominio	Modelo de dominio actualizado	<ul style="list-style-type: none"> • Actualizar el modelo de dominio con nuevas clases y atributos durante el análisis de robustez. 	
Actualizar el diagrama de clases de análisis	Modelo de dominio actualizado	<ul style="list-style-type: none"> • Actualizar el diagrama de clases de análisis al finalizar el análisis de robustez. • Asignar atributos a las clases entidad. 	

Tabla N° 3.4: Diseño Preliminar (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Revisar descripción de casos de uso	Caso de uso	<ul style="list-style-type: none"> • Coincidir la descripción del caso de uso con el diagrama de robustez. 	
Revisar diagrama de robustez	Diagrama de robustez	<ul style="list-style-type: none"> • Coincidir el diagrama de robustez con descripción del caso de uso. • Verificar que el diagrama de robustez cumpla las reglas. • Verificar que el diagrama de robustez tiene todo los cursos alternos. 	Usuario Cliente Analista Programador
Revisar modelo de dominio actualizado	Modelo de dominio actualizado	<ul style="list-style-type: none"> • Coincidir los objetos entidad del diagrama de robustez con el modelo de dominio actualizado. 	

Tabla N° 3.5: Revisión de diseño preliminar (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Diseñar diagrama de componentes	Diagrama de componentes	<ul style="list-style-type: none"> • Entrevistas • Características del negocio. • Utilizar la arquitectura MVC. • Integrar framework. 	Cliente Analista Programador
Diseñar diagrama de despliegue	Diagrama de despliegue	<ul style="list-style-type: none"> • Entrevistas. • Características del negocio. • Utilizar modelo cliente servidor. 	Arquitecto de software

Tabla N° 3.6: Arquitectura Técnica (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Dibujar un diagrama de secuencia para cada caso de uso	Diagrama de secuencia	<ul style="list-style-type: none"> • Copiar la descripción del caso de uso. • Copiar objetos entidad, interfaz y actores del diagrama de robustez. • Verificar que un mensaje del diagrama de secuencia es verbo en el caso de uso. • Hacer refactoring al diagrama de secuencia andes de codificar. 	Programador Diseñador
Actualizar el diagrama de clases de un caso de uso	Diagrama de clase	<ul style="list-style-type: none"> • Asignar operaciones a las clases a partir de mensajes del diagrama de secuencia. • Establecer multiplicidad en las clases. • Depurar las clases, operaciones y atributos del diagrama de clases. 	
Extraer controladores para pruebas unitarias	Lista de controladores	<ul style="list-style-type: none"> • Identificar controladores para la lógica del negocio desde un diagrama de robustez. 	

Tabla N° 3.7: Diseño Detallado (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Revisar diagrama de secuencia	Diagrama de secuencia	<p>Verificar que el diagrama de secuencia coincide con la descripción del caso de uso.</p> <p>Verificar que el diagrama de secuencia representa los cursos básico y alterno.</p> <p>Verificar en los mensajes que los atributos y los valores de retorno son correctos.</p>	Diseñador Programador
Revisar diagrama de clases	Diagrama de clases	<ul style="list-style-type: none"> • Verificar que el nombre, atributos y operaciones se asignaron correctamente a las clases. • Asignar requisitos no funcionales a los casos de uso y clases. 	
Revisar modelo de dominio actualizado	Modelo de dominio actualizado	<ul style="list-style-type: none"> • Verificar nombres y atributos del modelo de dominio actualizado. 	

Tabla N° 3.8: Revisión Crítica del Diseño (Rosenberg y Stephens, 2007)

TAREA	ARTEFACTOS	TÉCNICA	RESPONSABLE
Implementar la base de datos física	Diagrama Físico de la Base de Datos	<ul style="list-style-type: none"> • Utilizar el script usando el modelo de dominio actualizado. 	Programador
Implementar código SQL para tablas entidad	Código SQL para tablas entidad	<ul style="list-style-type: none"> • Ejecutar el script usando un DBMS. 	Programador
Implementar código para las interfaces	Código Fuente para las interfaces	<ul style="list-style-type: none"> • Generar código fuente usando una herramienta. 	Programador
Implementar código para clases DAO	Código Fuente para clases DAO	<ul style="list-style-type: none"> • Generar código fuente usando una herramienta. 	Programador
Ejecutar pruebas unitarias para cada controlador	Reporte de Pruebas Unitarias	<ul style="list-style-type: none"> • Ejecutar el módulo de cada prueba unitaria. • Modificar código fuente si la prueba unitaria muestra resultado incorrecto. 	Programador
Ejecutar pruebas de aceptación para cada caso de uso	Reporte de pruebas de aceptación	<ul style="list-style-type: none"> • Utilizar los casos de prueba de aceptación. • Ejecutar el módulo de un caso de uso. • Modificar código fuente si la prueba de aceptación muestra resultado incorrecto. 	Programador Usuario Cliente

Tabla N° 3.9: Implementación (Rosenberg y Stephens, 2007)

CAPITULO IV

ANALISIS Y RESULTADOS DE LA INVESTIGACIÓN

4.1. PROCESO DE MIGRACIÓN DE ARCHIVOS RELACIONADOS DE FOXPRO A UNA BASE DE DATOS ORACLE

La información académica de los estudiantes de la Escuela de Formación Profesional de Ingeniería de Sistemas del periodo académico 2008-2015, proporcionada por la Oficina de Informática y Sistemas (OGIS), son almacenados en el FOXPRO, estos registros son guardados en archivos con extensión .DBF, estos pueden ser visualizados por el programa de Microsoft Excel.

Para realizar la migración de estos archivos relacionados de FOXPRO, y como se mencionó líneas arriba, estos pueden ser visualizados por Microsoft Excel, utilizaremos las herramientas que nos proporciona el Excel, como son los macros; para lo cual se elaboró el Algoritmo de Migración presentado en el anexo C, tabla N° C.1, este permite estructurar la información de estos archivos relacionados en el mismo Excel, según el modelado de la base de datos física, implementada en el ORACLE, para así facilitarnos la migración utilizando herramientas que nos proporciona el mismo Gestor de Base de Datos.

4.2. ANÁLISIS DE LOS PROCESOS QUE INTERVIENEN EN LA INVESTIGACIÓN

4.2.1. PROCESO DE MATRÍCULA

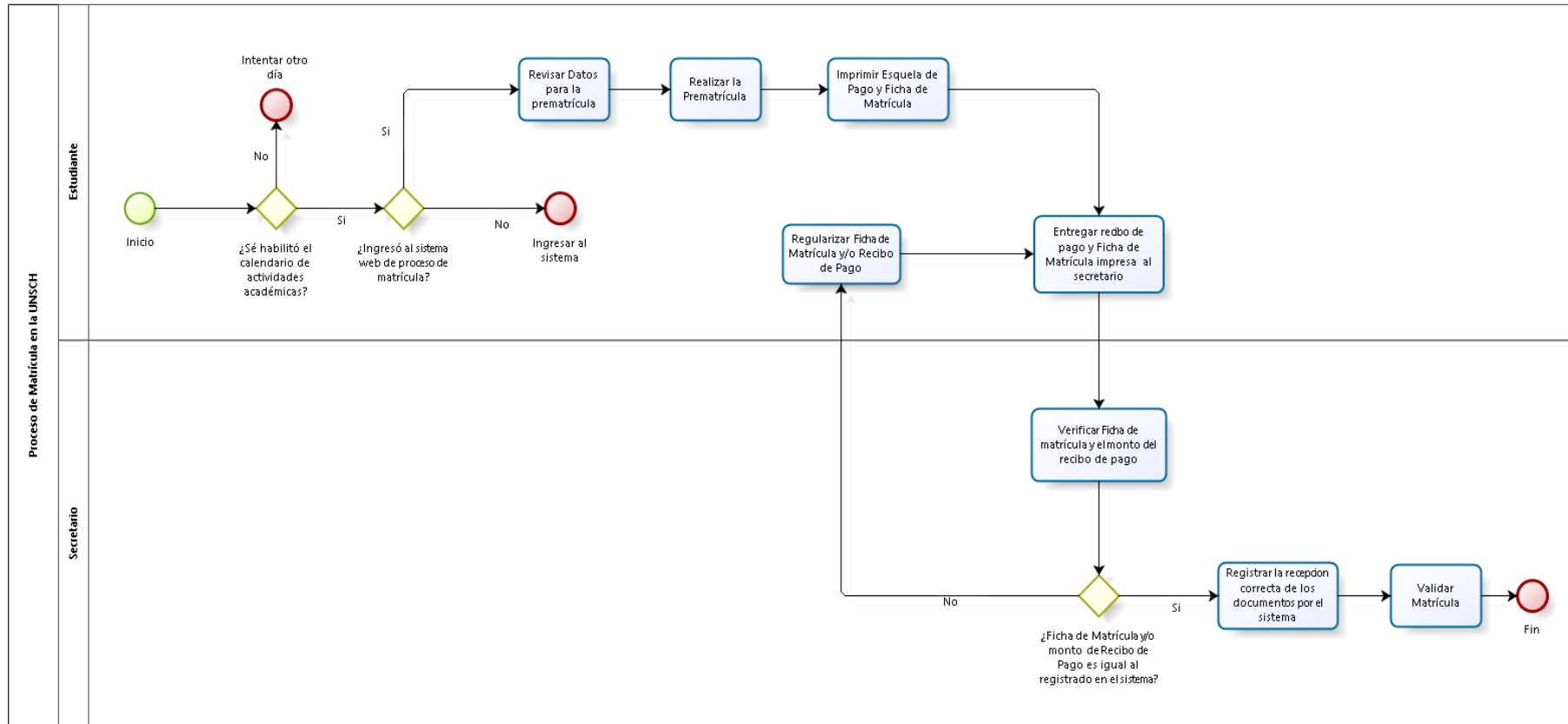


Figura N° 4.1: Flujo del Proceso de Matrícula (Elaboración Propia)

4.2.2. PROCESO DE DESMATRÍCULA

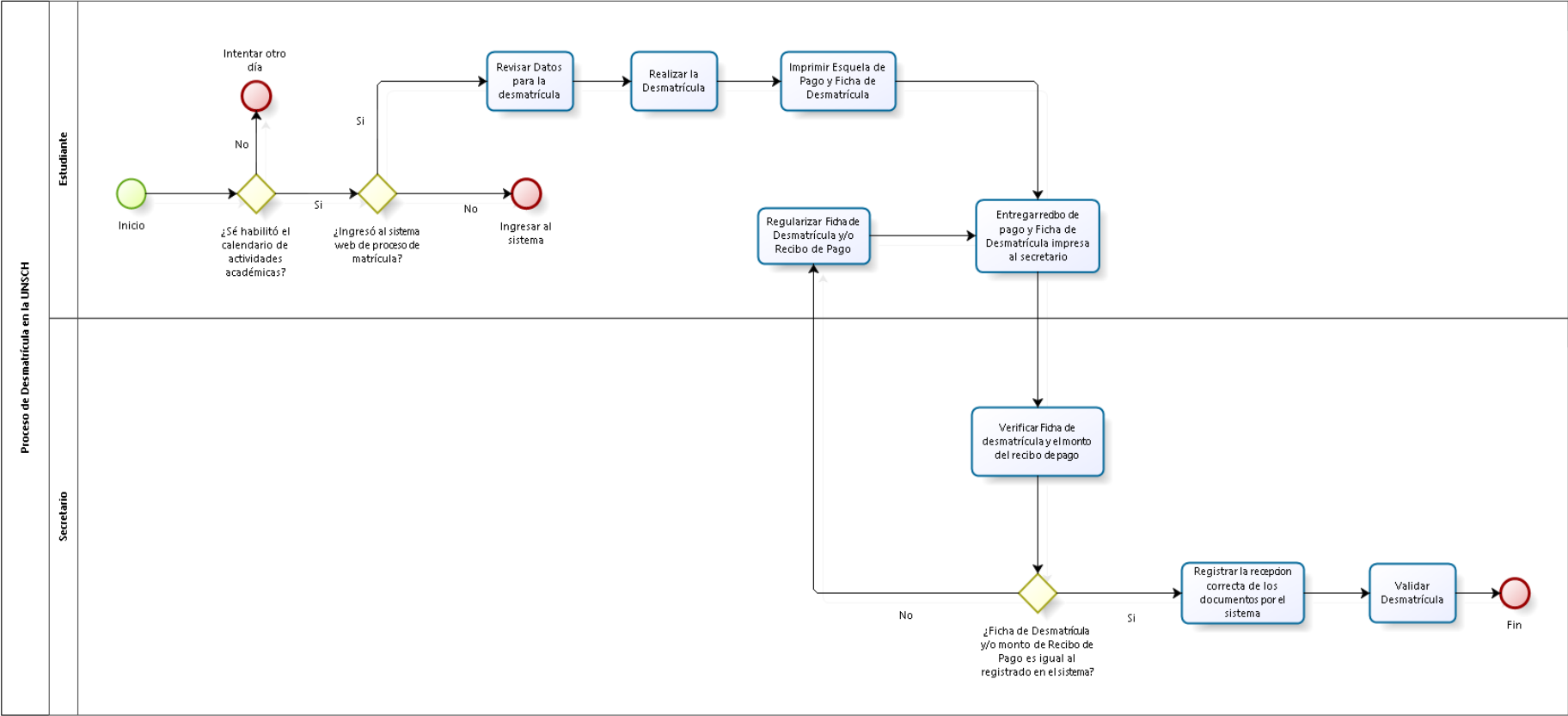


Figura N° 4.2: Flujo del Proceso de Desmatriculación (Elaboración Propia)

4.2.3. PROCESO DE CURSOS APLAZADOS

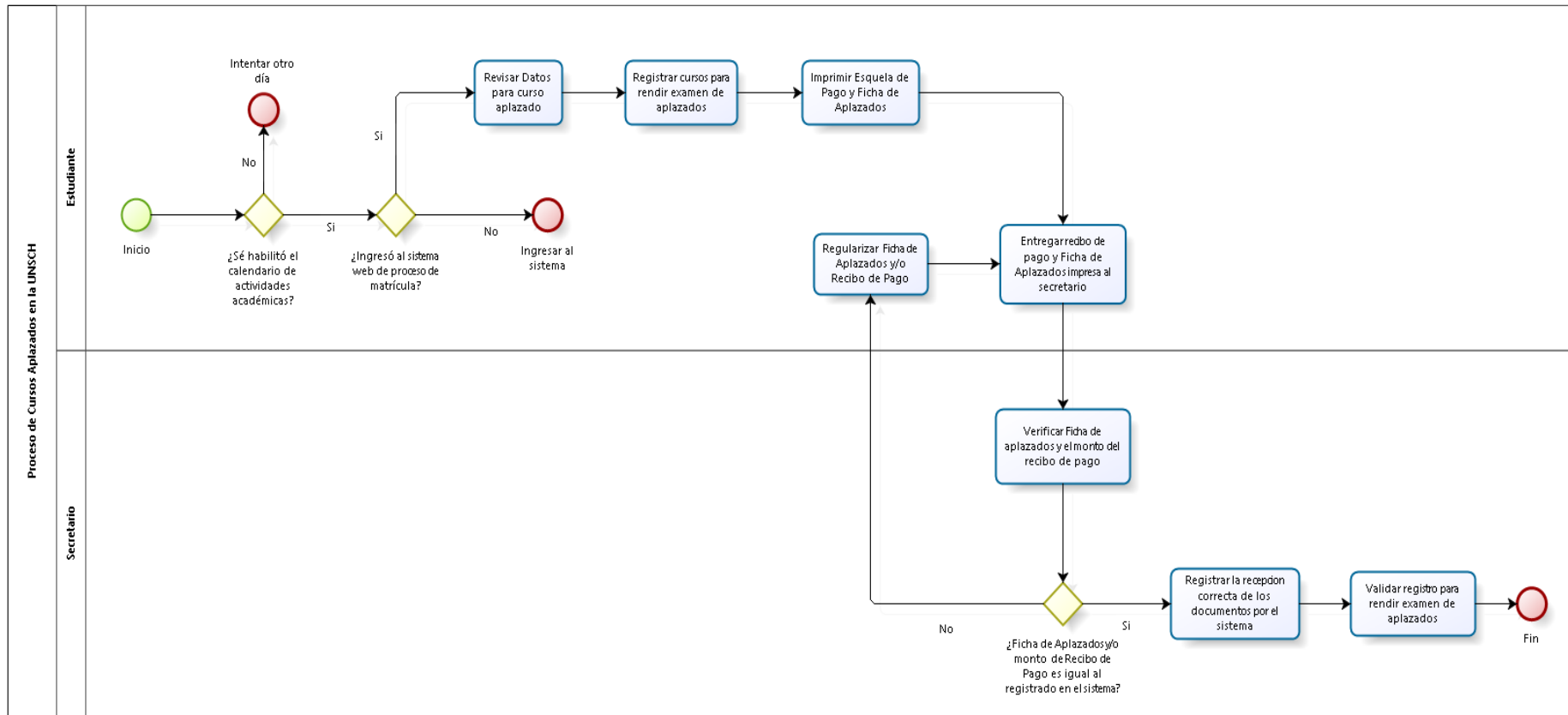


Figura N° 4.3: Flujo del Proceso de Cursos Aplazados (Elaboración Propia)

4.2.4. GENERAR REGISTRO DE EVALUACIÓN

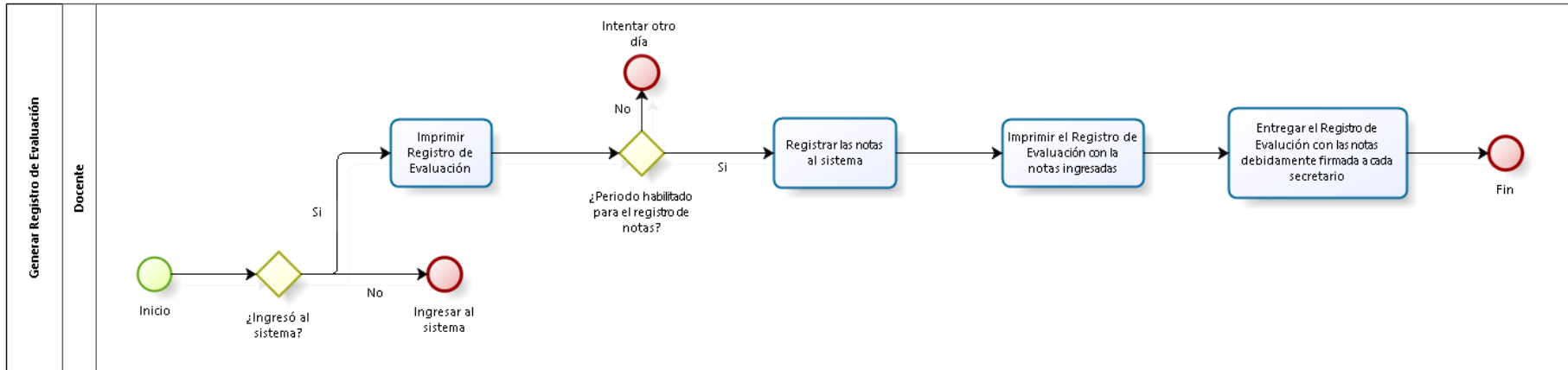


Figura N° 4.4: Flujo de Generar Registro de Evaluación (Elaboración Propia)

4.2.5. GENERAR ACTA DE EVALUACION FINAL

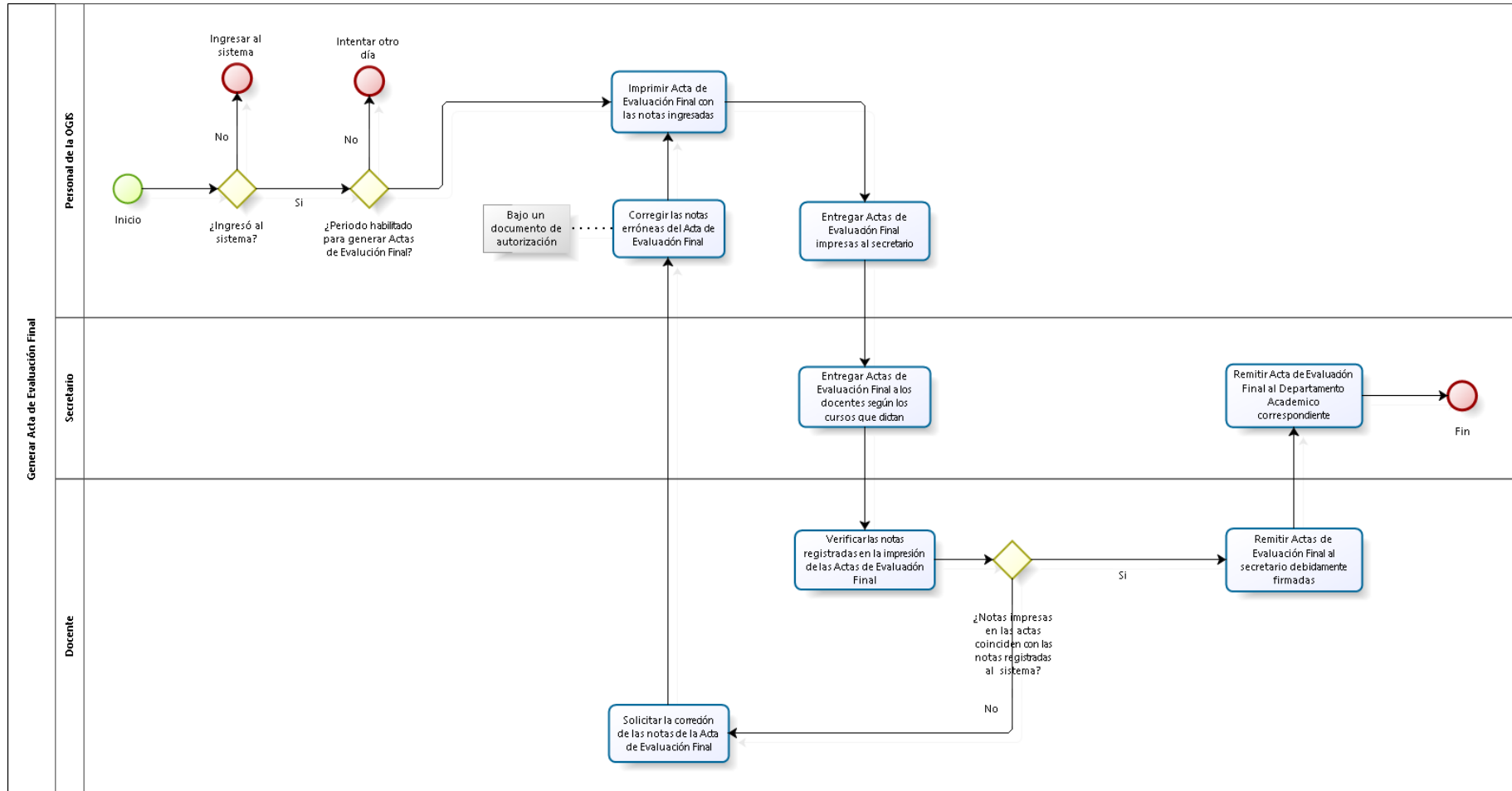


Figura N° 4.5: Flujo de Generar Acta de Evaluación Final (Elaboración Propia)

4.3. ARTEFACTOS DEL SOFTWARE APLICANDO LA METODOLOGIA ICONIX

Según las fases para desarrollar la aplicación web usando la metodología Iconix, desarrollado en las tablas 3.2 a 3.9, descrito en el capítulo II, sección 2.2.3, se obtienen los artefactos análisis de requisitos, revisión de requisitos, diseño preliminar, revisión de diseño preliminar, arquitectura técnica, diseño detallado, revisión crítica de diseño e implementación. A continuación mostramos los resultados de los artefactos de acuerdo a la metodología Iconix.

4.3.1. ANÁLISIS DE REQUISITOS

A. REQUISITOS FUNCIONALES

N°	REQUISITOS FUNCIONALES
01	El sistema debe ser accedido a través de una cuenta de usuario (administrador, secretario, estudiante, docente y personal del OGIS) y una contraseña.
02	El sistema debe permitir al administrador gestionar una cuenta de usuario .
03	El sistema debe permitir al administrador registrar, actualizar y deshabilitar los datos del personal administrativo .
04	El sistema debe permitir gestionar los datos del Estudiante.
05	El sistema debe permitir gestionar Facultad .
06	El sistema debe permitir gestionar Escuela de Formación Profesional y Especialidad .
07	El sistema de permitir gestionar Departamento Académico .
08	El sistema debe permitir gestionar Plan de Estudio .
09	El sistema debe permitir gestionar Periodo Académico .
10	El sistema debe permitir gestionar Calendario Académico .
11	El sistema debe permitir gestionar Curso , según el Plan de Estudio de cada Escuela de Formación Profesional.
12	El sistema debe permitir emitir una Esquela de Pago para la matrícula .
13	El sistema debe permitir al estudiante realizar matrícula .
14	El sistema debe permitir emitir Ficha de Matrícula de un estudiante.
15	El sistema debe permitir emitir una Esquela de Pago para la desmatrícula .
16	El sistema debe permitir al estudiante realizar desmatrícula.

N°	REQUISITOS FUNCIONALES
17	El sistema debe permitir emitir Ficha de Desmatrícula de un estudiante.
18	El sistema debe permitir gestionar carga académica .
19	El sistema debe permitir emitir el Registro de Evaluación de los cursos matriculados.
20	El sistema debe permitir registrar y actualizar las notas de los cursos matriculados de un estudiante.
21	El sistema debe permitir emitir una Esquela de Pago para rendir examen de curso aplazado.
22	El sistema debe permitir al estudiante realizar la inscripción del curso aplazado.
23	El sistema debe permitir emitir Ficha de Matrícula por curso aplazado.
24	El sistema debe permitir registrar y actualizar las notas de los cursos aplazados.
25	El sistema debe permitir emitir una Esquela de Pago para la matrícula de curso vacacional.
26	El sistema debe permitir al estudiante realizar matrícula de curso vacacional.
27	El sistema debe permitir emitir Ficha de Matrícula de curso vacacional.
28	El sistema debe permitir registrar y actualizar las notas de los cursos vacacionales.
29	El sistema debe permitir al Secretario realizar la Inscripción Especial de los cursos especiales solicitados por el estudiante.
30	El sistema debe permitir registrar y actualizar las notas de los cursos especiales.
31	El sistema debe permitir emitir las Actas de Evaluación Final de los cursos matriculados en el semestre regular.
32	El sistema debe permitir emitir el duplicado de las Actas de Evaluación Final de los cursos matriculados en el semestre regular.
33	El sistema debe permitir emitir las Actas de Evaluación Final de cursos especiales.
34	El sistema debe permitir emitir duplicado de las Actas de Evaluación Final de cursos especiales.
35	El sistema debe permitir emitir las Actas de Evaluación Final de cursos vacacionales.
36	El sistema debe permitir emitir duplicado de las Actas de Evaluación Final de cursos vacacionales.

N°	REQUISITOS FUNCIONALES
37	El sistema debe permitir al secretario emitir Reporte General de Matriculados por serie.
38	El sistema debe permitir emitir la boleta de notas de cada semestre académico.

Tabla N° 4.1: Requisitos funcionales (Elaboración Propia).

B. REQUISITOS NO FUNCIONALES

N°	REQUISITOS NO FUNCIONALES
01	El sistema debe proporcionar una interfaz amigable.
02	El sistema debe proporcionar un adecuado tiempo de respuesta.
03	El sistema debe usar la programación orientada a objetos.
04	El sistema debe ser capaz de brindar niveles de seguridad.

Tabla N° 4.2: Requisitos no funcionales (Elaboración Propia).

C. CASOS DE PRUEBA DE ACEPTACIÓN

N° Req.	N° C.P	CASOS DE PRUEBA DE ACEPTACIÓN
12 13 14	01	Comprobar que los datos de los estudiantes son correctos.
	02	Comprobar que los cursos matriculados, los créditos, la serie y el semestre académico en una Ficha de Matrícula de un estudiante son correctos.
	03	Verificar que el monto a pagar en una Esquela de Pago de un estudiante es correcto.
	04	Imprimir la Esquela de Pago y la Ficha de Matrícula de los estudiantes para un semestre académico.
18 19 20	05	Comprobar que los datos del personal administrativo son correctos.
	06	Comprobar que los cursos que dictan los docentes son correctos.
	07	Comprobar que el Registro de Evaluación, presenta información correcta.
	08	Verificar que los datos registrados en el Registro de Evaluación son correctos.
	09	Imprimir el Registro de Evaluación para un semestre académico.

31 32	10	Comprobar que la Acta de Evaluación Final, presenta información correcta.
	11	Imprimir el Acta de Evaluación Final para un semestre académico.

Tabla N° 4.3: Casos de Prueba de Aceptación (Elaboración Propia).

D. GLOSARIO DE TÉRMINOS

Cuenta de usuario	Facultad	Curso
Administrador	Escuela	Esquela de Pago
Personal administrativo	Especialidad	Matrícula
Secretario	Departamento Académico	Inscripción Especial
Docente	Plan de Estudio	Carga Académica
Personal OGIS	Calendario Académico	
Estudiante	Periodo Académico	

Tabla N° 4.4: Glosario de términos (Elaboración Propia).

E. MODELO DE DOMINIO

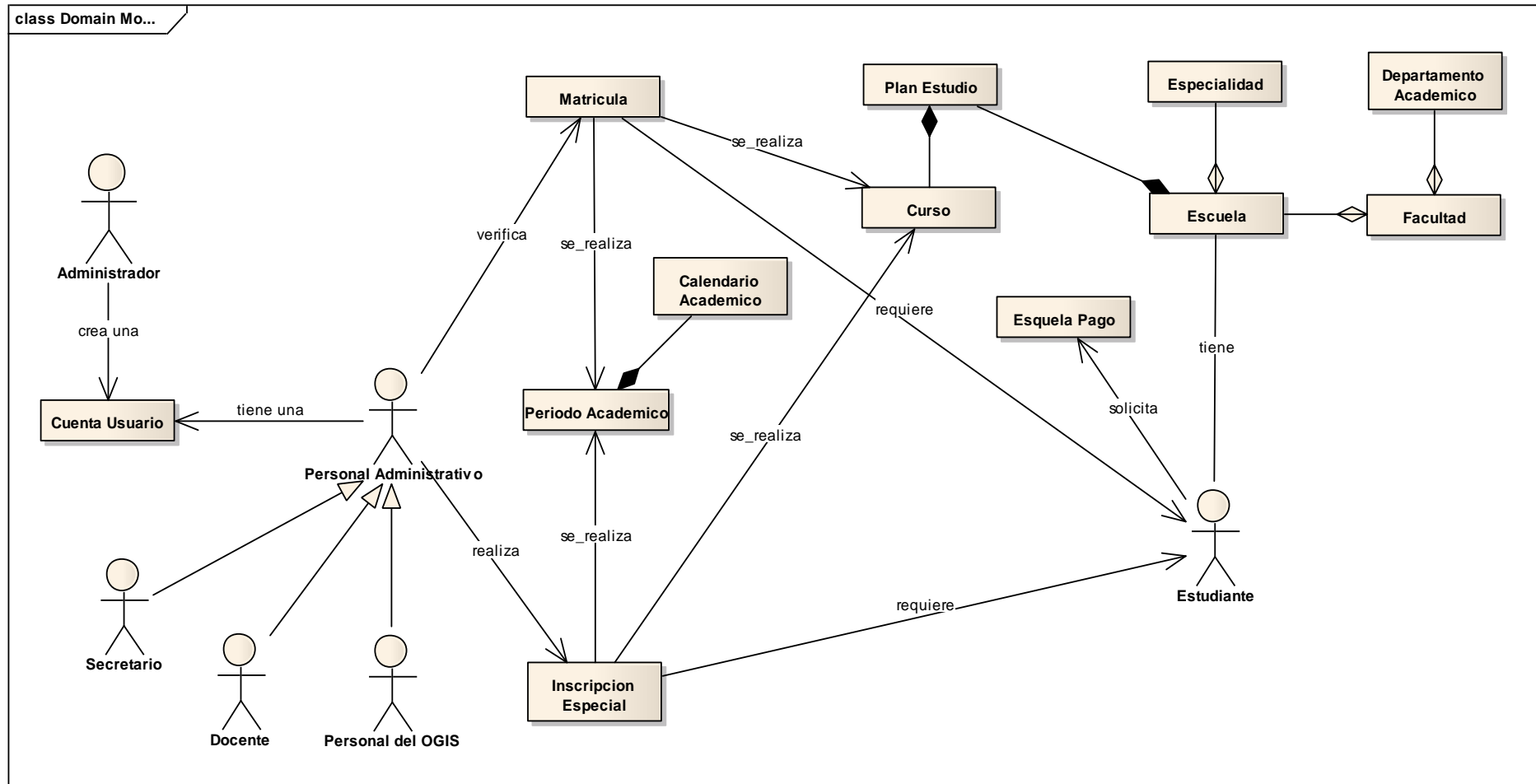


Figura N° 4.6: Modelo de Dominio Inicial (Elaboración Propia)

F. RELACIÓN ENTRE REQUISITOS FUNCIONALES Y CASOS DE USO

N°	REQUISITOS FUNCIONALES	CASOS DE USO
01	El sistema debe ser accedido a través de una cuenta de usuario y una contraseña.	CU01: Iniciar Sesión
02	El sistema debe permitir al administrador gestionar una cuenta de usuario.	CU02: Crear cuenta de usuario CU03: Actualizar cuenta de usuario
03	El sistema debe permitir a un administrador registrar, actualizar y deshabilitar los datos del personal administrativo.	CU04: Registrar datos del personal Administrativo
		CU05: Mantener datos del personal administrativo
04	El sistema debe permitir gestionar los datos del Estudiante.	CU06: Registrar Estudiante CU07: Actualizar Estudiante
05	El sistema debe permitir gestionar Facultad.	CU08: Registrar Facultad CU09: Actualizar Facultad
06	El sistema debe permitir gestionar Escuela de Formación Profesional y Especialidad.	CU10: Registrar Escuela de Formación Profesional CU11: Actualizar Escuela de Formación Profesional
07	El sistema de permitir gestionar Departamento Académico.	CU12: Registrar Departamento Académico CU13: Actualizar Departamento Académico
08	El sistema debe permitir gestionar Plan de Estudio.	CU14: Registrar Plan de Estudio CU15: Actualizar Plan de Estudio
09	El sistema debe permitir gestionar Periodo Académico.	CU16: Registrar Periodo Académico CU17: Actualizar Periodo Académico
10	El sistema debe permitir gestionar Calendario Académico.	CU18: Registrar Calendario Académico CU19: Actualizar Calendario Académico
11	El sistema debe permitir gestionar los Cursos, según el Plan de Estudio de cada Escuela de Formación Profesional.	CU20: Registrar curso CU21: Actualizar curso

N°	REQUISITOS FUNCIONALES	CASOS DE USO
12	El sistema debe permitir emitir una Esquela de Pago para la matrícula.	CU22: Emitir Esquela de Pago de matrícula regular
13	El sistema debe permitir al estudiante realizar matrícula.	CU23: Registrar Matrícula (Prematrícula)
14	El sistema debe permitir emitir la Ficha de Matrícula de un estudiante.	CU24: Emitir Ficha de Matrícula
15	El sistema debe permitir emitir una Esquela de Pago para la desmatrícula.	CU25: Emitir Esquela de Pago por desmatrícula
16	El sistema debe permitir al estudiante realizar desmatrícula.	CU26: Registrar Desmatrícula
17	El sistema debe permitir emitir Ficha de Desmatrícula de un estudiante.	CU27: Emitir Ficha de Desmatrícula
18	El sistema debe permitir gestionar carga académica.	CU28: Registrar carga académica CU29: Actualizar carga académica
19	El sistema debe permitir emitir el Registro de Evaluación de los cursos matriculados.	CU30: Emitir Registro de Evaluación
20	El sistema debe permitir registrar y actualizar las notas de los cursos matriculados de un estudiante.	CU31: Registrar nota del curso del semestre regular CU32: Actualizar nota del curso del semestre regular
21	El sistema debe permitir emitir una Esquela de Pago para rendir examen de curso aplazado.	CU33: Emitir Esquela de Pago por curso aplazado
22	El sistema debe permitir emitir Ficha de Matrícula por curso aplazado.	CU34: Registrar inscripción de curso aplazado
23	El sistema debe permitir emitir Ficha de Matrícula por curso aplazado.	CU35: Emitir Ficha de Matrícula por curso aplazado
24	El sistema debe permitir registrar y actualizar las notas de los cursos aplazados.	CU36: Registrar nota de curso aplazado CU37: Actualizar nota de curso aplazado

N°	REQUISITOS FUNCIONALES	CASOS DE USO
25	El sistema debe permitir emitir una Esquela de Pago para la matrícula de curso vacacional.	CU38: Emitir Esquela de Pago por curso vacacional
26	El sistema debe permitir al estudiante realizar matrícula de curso vacacional.	CU39: Registrar matrícula de curso vacacional
27	El sistema debe permitir emitir Ficha de Matrícula de curso vacacional.	CU40: Emitir Ficha de Matrícula por curso vacacional
28	El sistema debe permitir registrar y actualizar las notas de los cursos vacacionales.	CU41: Registrar nota de curso vacacional CU42: Actualizar nota de curso vacacional
29	El sistema debe permitir al Secretario realizar la Inscripción Especial de los cursos especiales solicitados por el estudiante.	CU43: Registrar inscripción de curso especial
30	El sistema debe permitir registrar y actualizar las notas de los cursos especiales.	CU44: Registrar nota de curso especial CU45: Actualizar nota de curso especial
31	El sistema debe permitir emitir las Actas de Evaluación Final de los cursos matriculados en el semestre regular.	CU46: Emitir Acta de Evaluación Final del semestre regular
32	El sistema debe permitir emitir el duplicado de las Actas de Evaluación Final de los cursos matriculados en el semestre regular.	CU47: Emitir duplicado de Acta de Evaluación Final del semestre regular
33	El sistema debe permitir emitir las Actas de Evaluación Final de cursos especiales.	CU48: Emitir Acta de Evaluación Final de curso especial
34	El sistema debe permitir emitir duplicado de las Actas de Evaluación Final de cursos especiales.	CU49: Emitir duplicado de Acta de Evaluación Final de curso especial

N°	REQUISITOS FUNCIONALES	CASOS DE USO
35	El sistema debe permitir emitir las Actas de Evaluación Final de cursos vacacionales.	CU50: Emitir Acta de Evaluación Final de curso vacacional
36	El sistema debe permitir emitir duplicado de las Actas de Evaluación Final de cursos vacacionales.	CU51: Emitir duplicado de Acta de Evaluación Final de curso vacacional
37	El sistema debe permitir al secretario emitir Reporte General de Matriculados por serie.	CU52: Emitir Reporte General de Matriculados por serie
38	El sistema debe permitir emitir la boleta de notas de cada semestre académico.	CU53: Emitir Boleta de Notas

Tabla N° 4.5: Relación entre requisitos funcionales y casos de uso (Elaboración Propia).

G. LISTA DE CASOS DE USO

N°	CASOS DE USO
01	Iniciar Sesión
02	Crear cuenta de usuario
03	Actualizar cuenta de usuario
04	Registrar datos del personal Administrativo
05	Mantener datos del personal administrativo
06	Registrar Estudiante
07	Actualizar Estudiante
08	Registrar Facultad
09	Actualizar Facultad
10	Registrar Escuela de Formación Profesional
11	Actualizar Escuela de Formación Profesional
12	Registrar Departamento Académico
13	Actualizar Departamento Académico
14	Registrar Plan de Estudio
15	Actualizar Plan de Estudio

N°	CASOS DE USO
16	Registrar Periodo Académico
17	Actualizar Periodo Académico
18	Registrar Calendario Académico
19	Actualizar Calendario Académico
20	Registrar curso
21	Actualizar curso
22	Emitir Esquela de Pago de matrícula regular
23	Registrar Matrícula (Prematrícula)
24	Emitir Ficha de Matrícula
25	Emitir Esquela de Pago por desmatrícula
26	Registrar Desmatrícula
27	Emitir Ficha de Desmatrícula
28	Registrar carga académica
29	Actualizar carga académica
30	Emitir Registro de Evaluación
31	Registrar nota del curso del semestre regular
32	Actualizar nota del curso del semestre regular
33	Emitir Esquela de Pago por curso aplazado
34	Registrar inscripción de curso aplazado
35	Emitir Ficha de Matrícula por curso aplazado
36	Registrar nota de curso aplazado
37	Actualizar nota de curso aplazado
38	Emitir Esquela de Pago por curso vacacional
39	Registrar matrícula de curso vacacional
40	Emitir Ficha de Matrícula por curso vacacional
41	Registrar nota de curso vacacional
42	Actualizar nota de curso vacacional
43	Registrar inscripción de curso especial
44	Registrar nota de curso especial

N°	CASOS DE USO
45	Actualizar nota de curso especial
46	Emitir Acta de Evaluación Final del semestre regular
47	Emitir duplicado de Acta de Evaluación Final del semestre regular
48	Emitir Acta de Evaluación Final de curso especial
49	Emitir duplicado de Acta de Evaluación Final de curso especial
50	Emitir Acta de Evaluación Final de curso vacacional
51	Emitir duplicado de Acta de Evaluación Final de curso vacacional
52	Emitir Reporte General de Matriculados por serie
53	Emitir Boleta de Notas

Tabla N° 4.6: Lista de Casos de Uso (Elaboración Propia).

H. PAQUETES DE CASOS DE USO

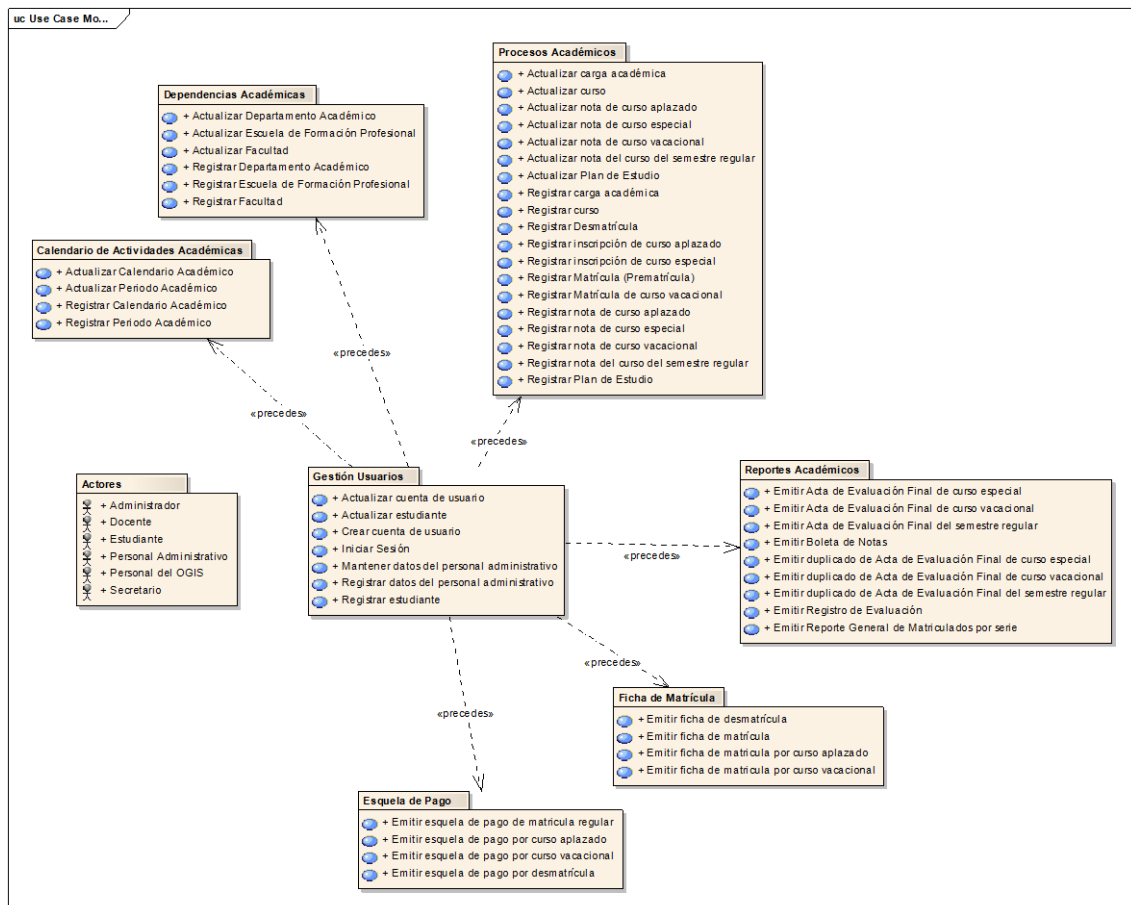


Figura N° 4.7: Paquetes de casos de uso (Elaboración Propia)

I. DIAGRAMA DE CASOS DE USO

H.1. Paquete Gestión Usuarios

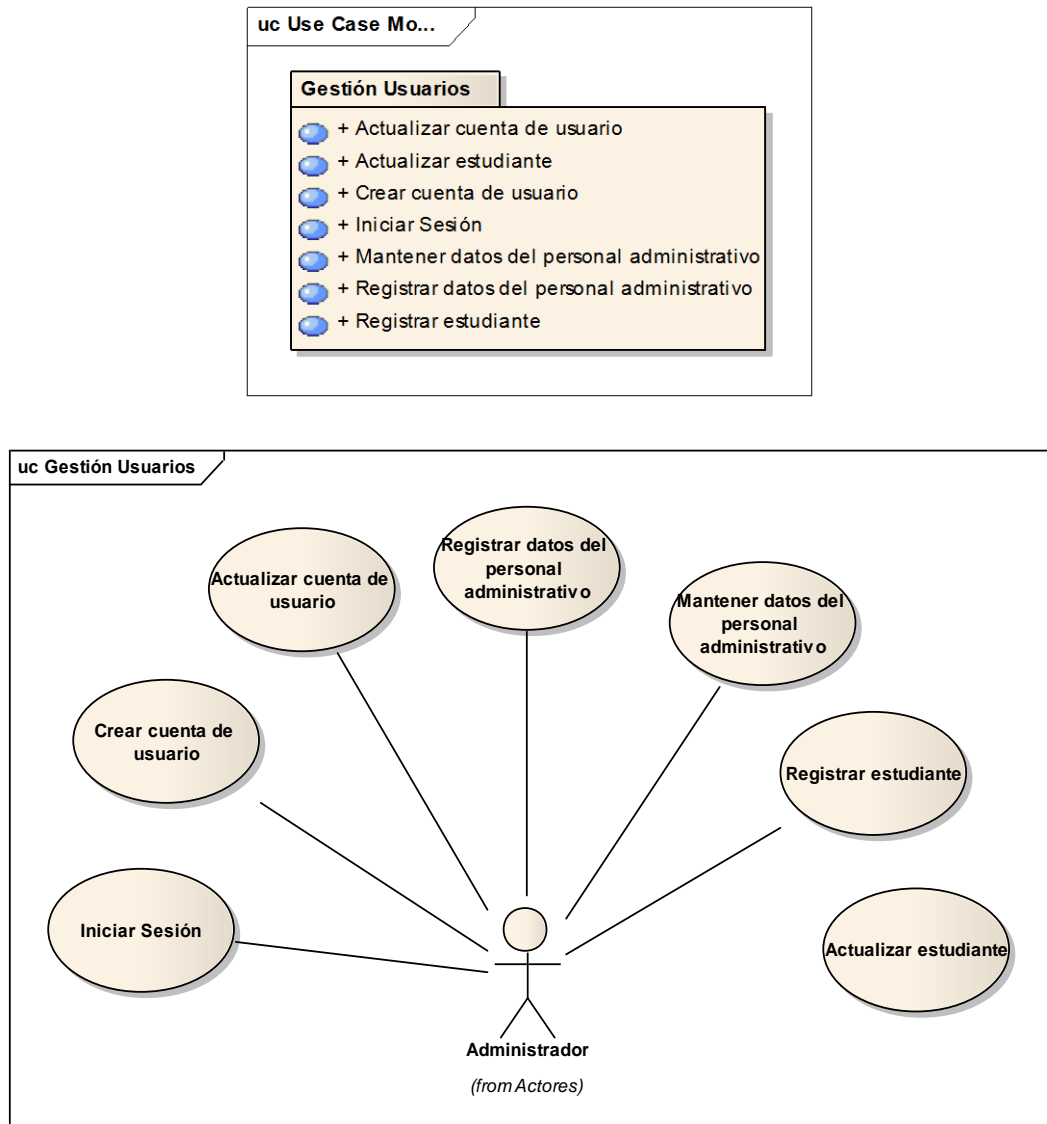
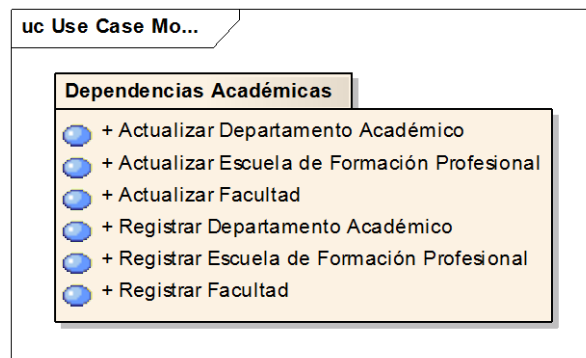


Figura N° 4.8: Paquete Gestión Usuarios (Elaboración Propia)

H.2. Paquete Dependencias Académicas



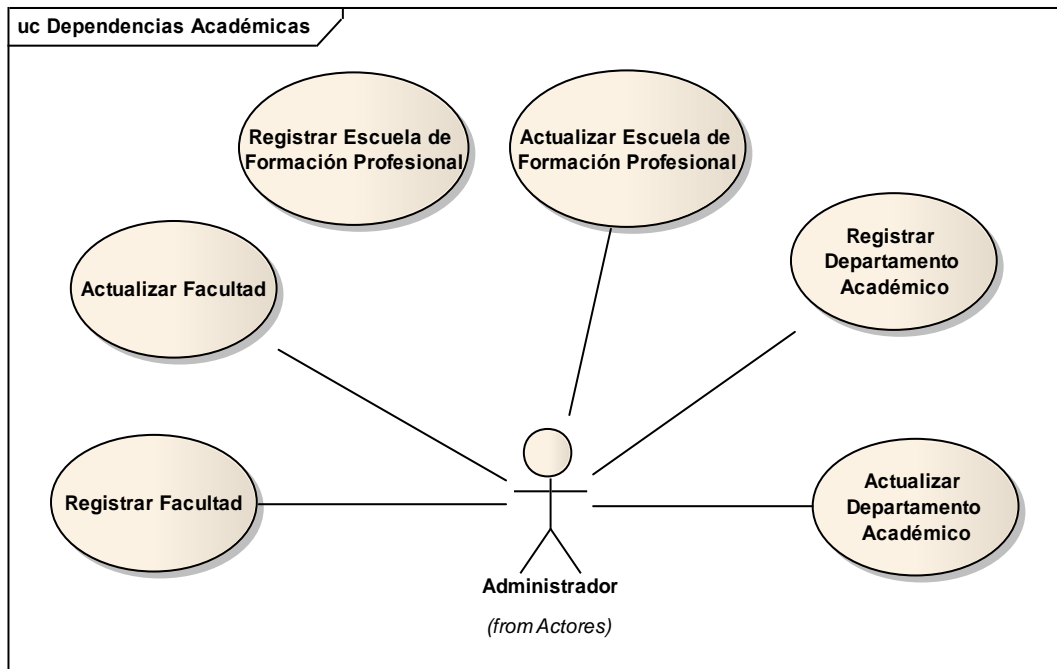
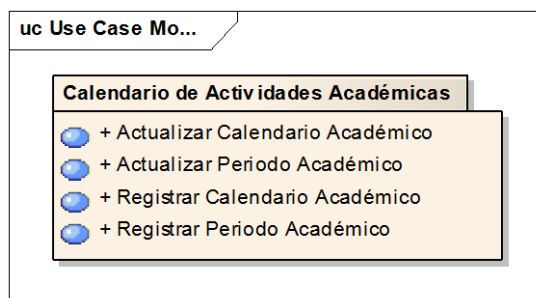


Figura N° 4.9: Paquete Dependencias Académicas (Elaboración Propia)

H.3. Paquete Calendario de Actividades Académicas



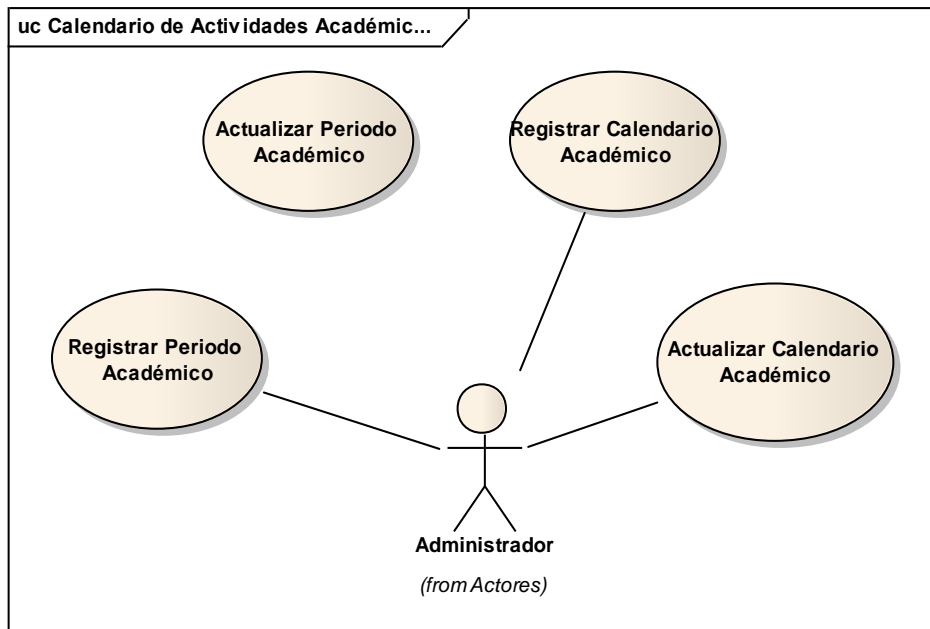
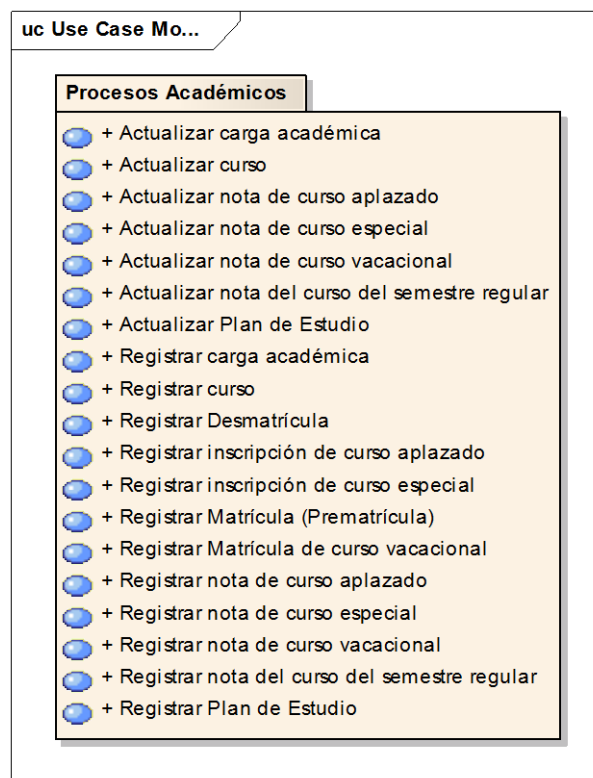


Figura N° 4.10: Paquete Calendario de Actividades Académicas (Elaboración Propia)

H.4. Paquete Procesos Académicos



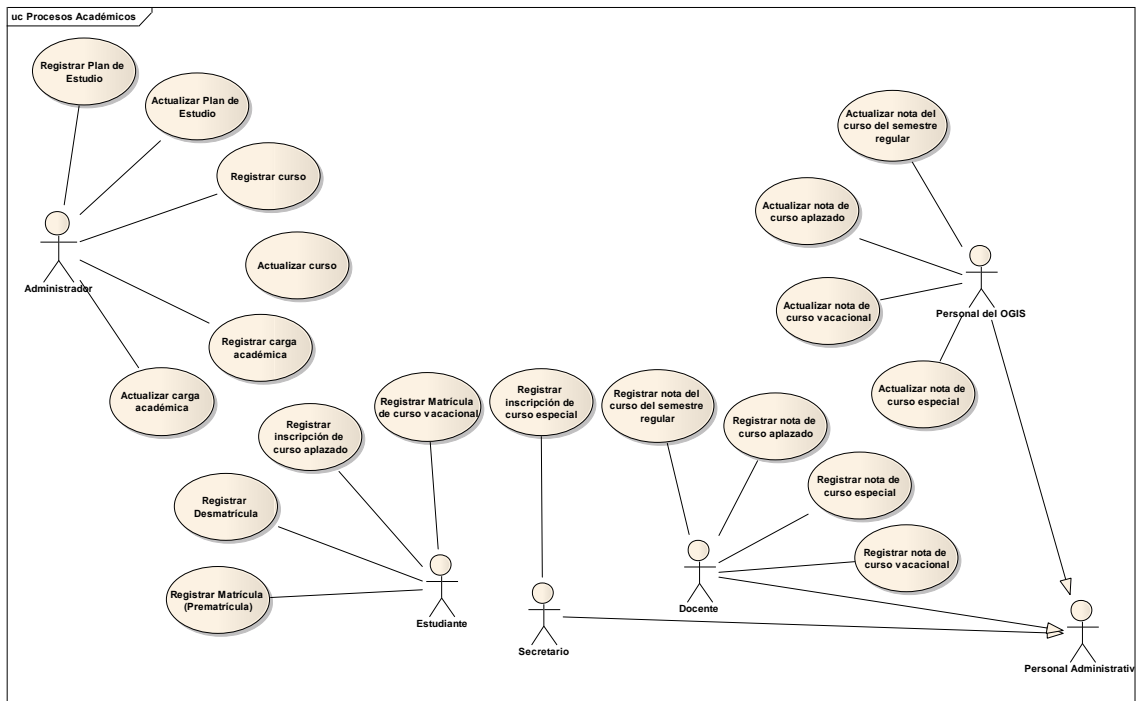


Figura N° 4.11: Paquete Procesos Académicas (Elaboración Propia)

H.5. Paquete Reportes Académicos

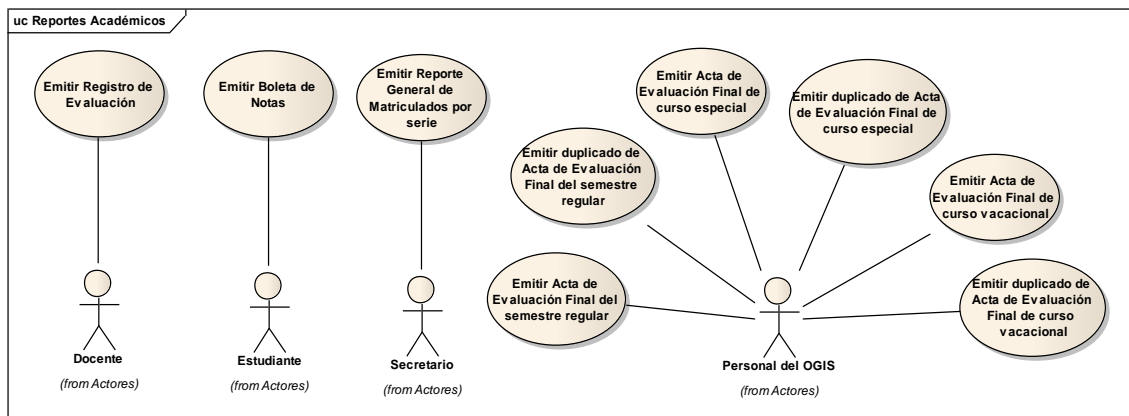
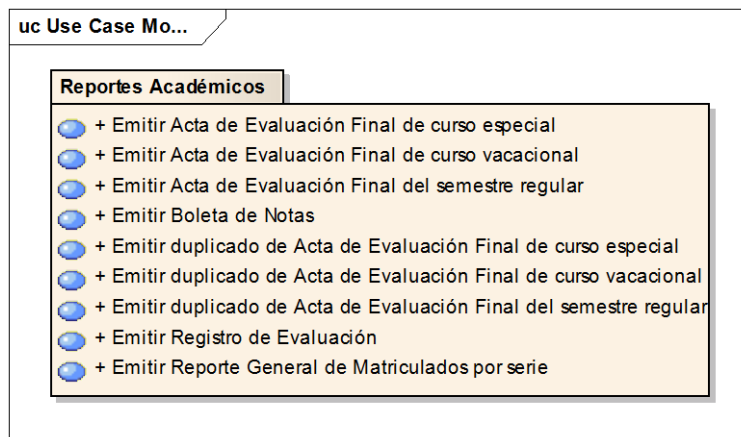


Figura N° 4.12: Paquete Reportes Académicas (Elaboración Propia)

H.6. Paquete Ficha de Matrícula

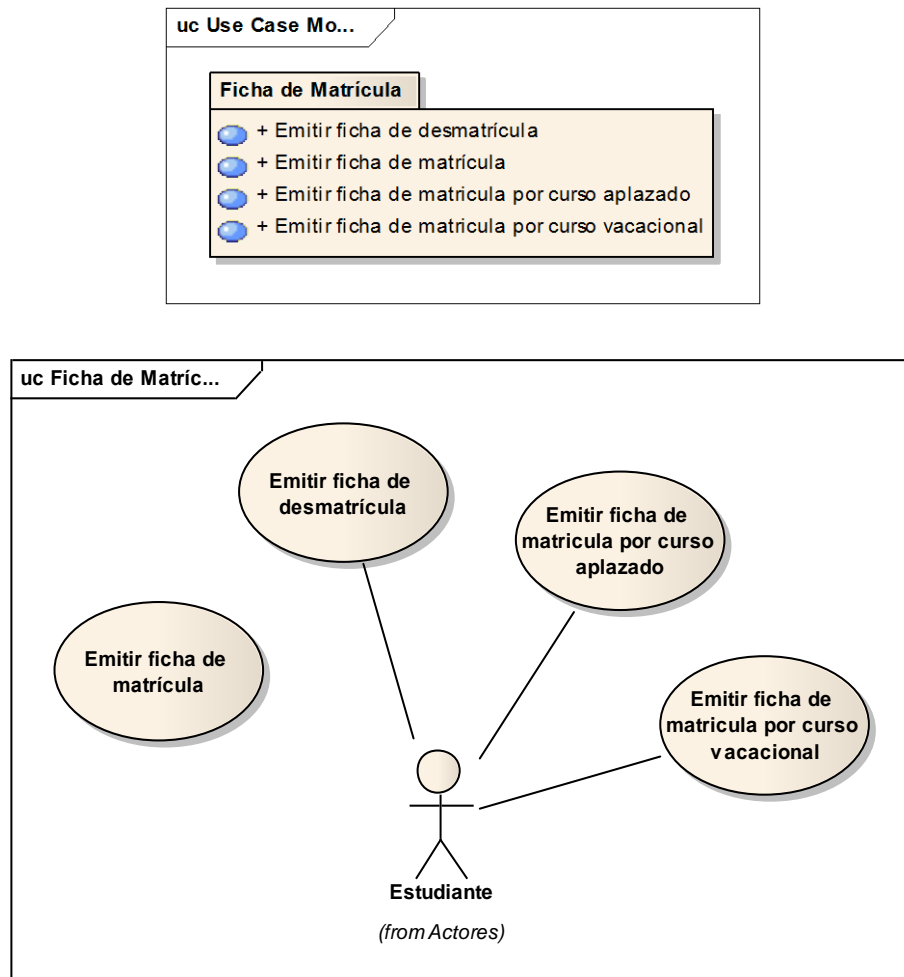
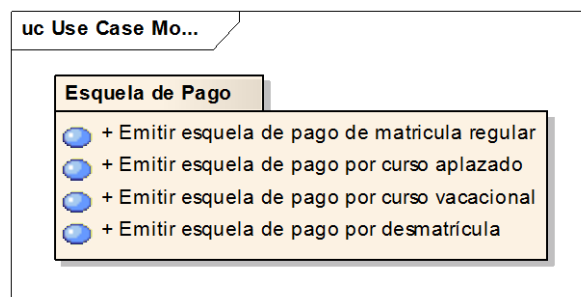


Figura N° 4.13: Paquete Ficha de Matrícula (Elaboración Propia)

H.7. Paquete Esquela de Pago



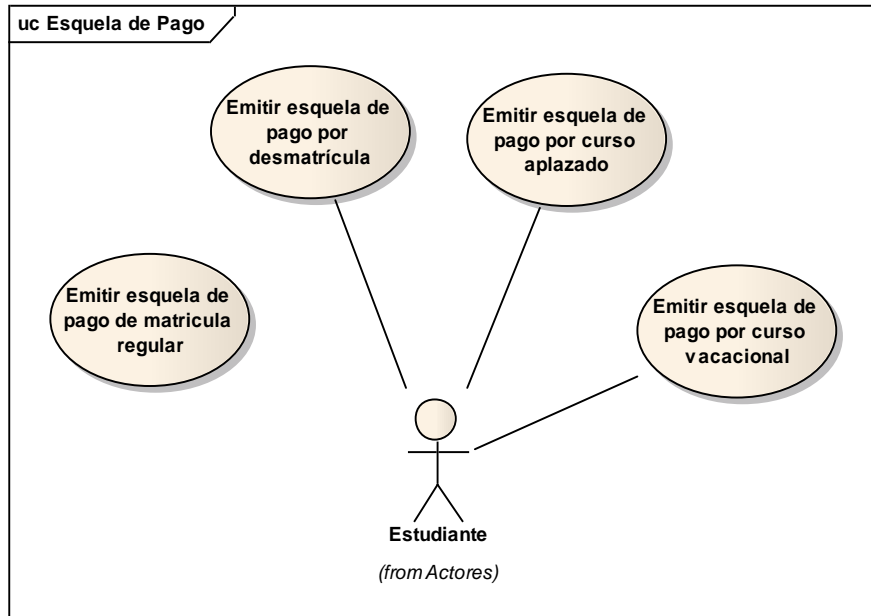


Figura N° 4.14: Paquete Esqueta de Pago (Elaboración Propia)

H.8. Paquete Actores

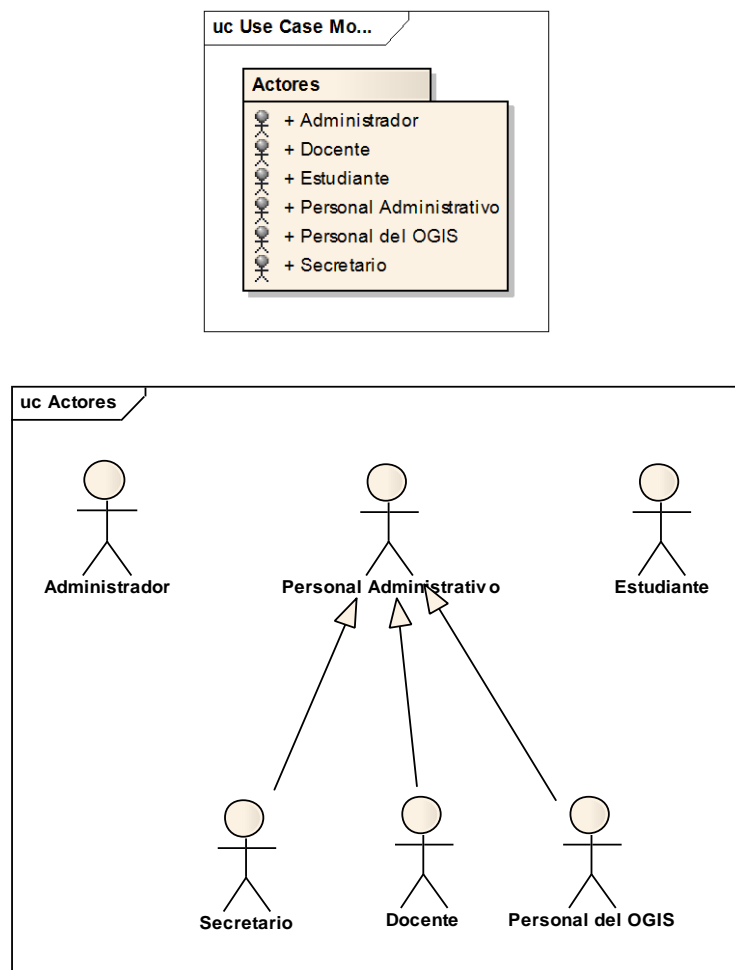


Figura N° 4.15: Paquete Actores (Elaboración Propia)

CU24: Emitir Ficha de Matrícula

Prototipo de interfaz gráfica para "Emitir Ficha de Matrícula". El navegador muestra la URL <http://www.matricula.unsch.edu.pe/>. El título de la página es "Ficha de Matrícula".

Semestre Académico: 2016 - I

SIGLA	TIPO	PLAN	NOMBRES	REQUISITOS	CREDITO
IS241	R	2005	Estructura de Datos	IS142	4
MA241	R	2005	Algebra Lineal	MA141	3

N° DE CURSOS	SERIE	TOTAL DE CRÉDITOS
02	200	7

Botón: Imprimir

Figura N° 4.18: Prototipo de interfaz gráfica "Emitir Ficha de Matrícula" (Elaboración Propia)

CU30: Emitir Registro de Evaluación

Prototipo de interfaz gráfica para "Emitir Registro de Evaluación". El navegador muestra la URL <http://www.matricula.unsch.edu.pe/>. El título de la página es "Registro de Evaluación".

Escuela - Curso: Ingeniería de Sistemas - IS141 Algoritmos

N°	CODIGO	APELLIDOS Y NOMB	PARCIA	PARCIA	PRACTIC	LABORATO	TRABAJO F	PROMEDIO F
01	271257	CALLE MELGAR Kath						
02	271156	CENTERO QUISPE M						
03	271262	ESPINOZA MENDOZ,						

Botón: Imprimir

Figura N° 4.19: Prototipo de interfaz gráfica "Emitir Registro de Evaluación" (Elaboración Propia)

CU31: Registrar nota del curso del semestre regular

Nº	CODIGO	APELLIDOS Y NOMB	PARCIA	PARCIA	PRACTIC	LABORATC	TRABAJO F	PROMEDIO F
01	271257	CALLE MELGAR Kath	12	13	12	16	14	13
02	271156	CENTERO QUISPE M	13	14	12	16	16	14
03	271262	ESPINOZA MENDOZ,	15	13	12	14	15	14

Figura N° 4.20: Prototipo de interfaz gráfica “Registrar nota del curso del semestre regular” (Elaboración Propia)

CU46: Emitir Acta de Evaluación Final del semestre regular

Nº	CODIGO	APELLIDOS Y NOMBRES	EVALUACION FINAL	OBSERVACIONES
01	27125790	CALLE MELGAR Katherine	13	Trece
02	27115690	CENTERO QUISPE Marcelino	14	Catorce
03	27126229	ESPINOZA MENDOZA Yuri	14	Catorce

Figura N° 4.21: Prototipo de interfaz gráfica “Emitir Acta de Evaluación Final del semestre regular” (Elaboración Propia)

K. DESCRIPCIÓN DEL PRIMER BORRADOR DE CASOS DE USO

CASO DE USO	DESCRIPCIÓN
<p>CU23. Registrar Matrícula (Prematrícula)</p>	<p>Curso Básico:</p> <p>El estudiante hace clic en pre matrícula, el sistema muestra los datos necesarios del estudiante para su matrícula, el estudiante verifica sus datos, y si no es correcto, hace clic en el botón reportar error y registra el error encontrado, luego hace clic en el botón enviar, el sistema envía y guarda el reporte de error.</p> <p>Si no encuentra un error en los datos, hace clic en siguiente y el sistema le muestra los cursos disponibles a matricularse, el estudiante selecciona los cursos a matricularse, el sistema agrega y registra los cursos y muestra la cantidad de cursos matriculados y total de créditos. Finalmente, el estudiante hace clic en siguiente, el sistema incluye el caso de uso Emitir Esquela de Pago y el caso de uso Emitir Ficha de Matrícula.</p> <p>Curso Alternativo:</p> <ul style="list-style-type: none"> - El sistema muestra un mensaje de error, se excede la cantidad de crédito a matricularse. - El sistema muestra un mensaje de error, no se registró correctamente los cursos matriculados.

Tabla N° 4.7: Primer Borrador de caso de uso “Registrar Matrícula” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU22: Emitir Esquela de Pago de matrícula regular</p>	<p>Curso Básico:</p> <p>El estudiante hace clic en Esquela de Pago, el sistema genera la esquela de pago del semestre correspondiente y el estudiante hace clic en imprimir.</p> <p>Curso Alternativo:</p> <ul style="list-style-type: none"> - El sistema muestra un mensaje de error, no se puede imprimir Esquela de Pago.

Tabla N° 4.8: Primer Borrador de caso de uso “Emitir Esquela de Pago de matrícula regular” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU24: Emitir Ficha de Matrícula</p>	<p>Curso Básico:</p> <p>El estudiante hace clic en Ficha de Matrícula, el sistema muestra el semestre académico a seleccionar, el estudiante selecciona el semestre académico, el sistema genera la ficha de matrícula del semestre seleccionado y el estudiante hace clic en imprimir.</p> <p>Curso Alternativo:</p> <ul style="list-style-type: none"> - El sistema muestra un mensaje de error, no se puede generar ficha de matrícula de semestre académico seleccionado. - El sistema muestra un mensaje de error, no se puede imprimir Ficha de Matrícula.

Tabla N° 4.9: Primer Borrador de caso de uso “Emitir Ficha de Matrícula” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU30: Emitir Registro de Evaluación</p>	<p>Curso Básico:</p> <p>El docente hace clic en Emitir Registro de Evaluación, el sistema muestra la escuela - curso a seleccionar, el docente selecciona los campos mencionados, el sistema genera el Registro de Evaluación y posteriormente el docente hace clic en imprimir.</p> <p>Curso Alternativo:</p> <ul style="list-style-type: none"> - El sistema muestra un mensaje de error, no se puede generar registro de evaluación. - El sistema muestra un mensaje de error, no se puede imprimir Registro de Evaluación.

Tabla N° 4.10: Primer Borrador de caso de uso “Emitir Registro de Evaluación” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU31. Registrar nota del curso del semestre regular</p>	<p>Curso Básico:</p> <p>El docente hace clic en Registrar nota del curso del semestre regular, el sistema muestra la escuela - curso a seleccionar, el docente selecciona los campos mencionados, el sistema muestra el Registro de Evaluación con los datos correspondientes, el docente registra las notas en el Registro de Evaluación y hace clic en el botón guardar, y el sistema valida, guarda las notas registradas y muestra un mensaje de confirmación "Se guardó correctamente las notas registradas", y luego el docente hace clic en imprimir.</p> <p>Curso Alternativo:</p> <ul style="list-style-type: none"> - El sistema muestra un mensaje de error, no se muestra el Registro de Evaluación. - El sistema muestra un mensaje de error, error de validación. - El sistema muestra un mensaje de error, no se guardó correctamente las notas registradas. - El sistema muestra un mensaje de error, no se puede imprimir Registro de Evaluación.

Tabla N° 4.11: Primer Borrador de caso de uso "Registrar nota del curso del semestre regular"

(Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU46. Emitir Acta de Evaluación Final del semestre regular</p>	<p>Curso Básico:</p> <p>El personal del OGIS hace clic en Emitir Acta de Evaluación Final del semestre regular, el sistema muestra la escuela - curso a seleccionar, el personal del OGIS selecciona los campos mencionados, el sistema genera el Acta de Evaluación Final del semestre regular y el personal del OGIS hace clic en imprimir.</p> <p>Curso Alternativo:</p> <ul style="list-style-type: none"> - El sistema muestra un mensaje de error, no se puede generar Acta de Evaluación Final del semestre regular.

	- El sistema muestra un mensaje de error, no se puede imprimir Acta de Evaluación Final
--	--

Tabla N° 4.12: Primer Borrador de caso de uso “Emitir Acta de Evaluación Final del semestre regular”
(Elaboración Propia)

4.3.2. REVISIÓN DE REQUISITOS

A. MODELO DE DOMINIO REVISADO

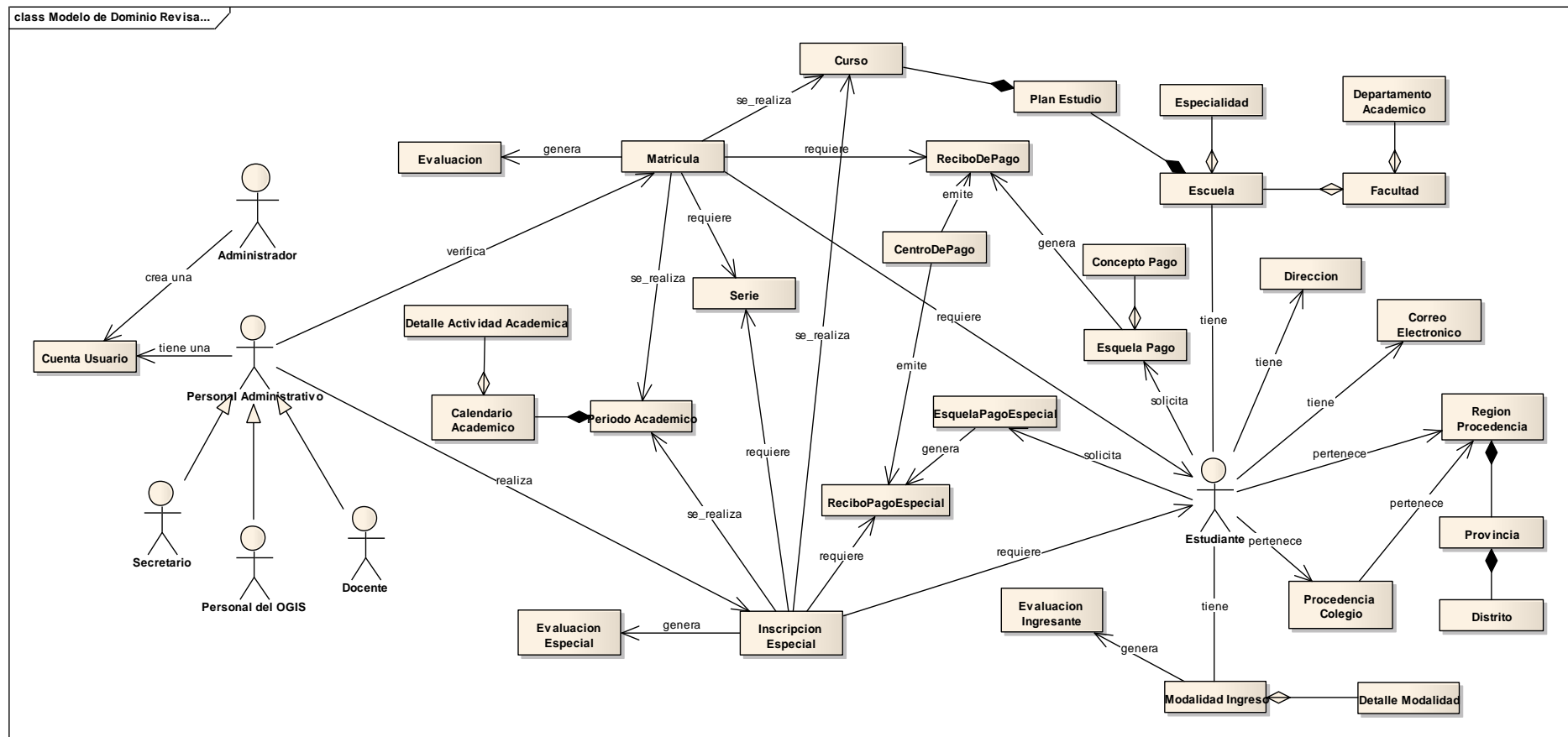


Figura N° 4.22: Modelo de Dominio Revisado (Elaboración Propia)

B. PROTOTIPO GUI REVISADO

CU23: Registrar Matrícula (Prematrícula)

The image shows a web interface for pre-registration. On the left is a dark sidebar with navigation links: 'Pre-Matricula', 'Fichas de Matricula', 'Boletas de Notas', 'Record Academico', and 'Mi cuenta'. The main content area is titled 'Pre-Matricula (3 pasos)' and includes a progress bar with three steps: '1. Inicio' (highlighted in green), '2. Cursos', and '3. Finalizar'. Below the progress bar, the user is identified as 'Joven Estudiante' and a warning message states: 'Antes de poder seguir con su Pre Matricula, le sugerimos tenga definido los cursos que desea matricularse, una vez que guarde y cierre el formulario ya no podra realizar modificaciones.' A prompt asks the user to 'Revisar si sus datos son correctos:'. The form contains the following fields: 'Facultad' (INGENIERIA DE MINAS, GEOLOGIA Y CIVIL), 'Escuela' (INGENIERIA DE SISTEMAS), 'Codigo' (27160125), 'Apellidos y Nombres' (GIL VARGAS, Sheyla), 'Modalidad' (Primera Profesion), and 'Semestre' (2016 - I). An orange button labeled 'Reportar un error' is located below the form. At the bottom right, there are four buttons: 'Anterior' (disabled), 'Siguiente' (active), 'Finalizar' (disabled), and 'Cancelar' (disabled).

Figura N° 4.23: Prototipo de interfaz gráfica revisada “Registrar Matrícula” [Paso 1] (Elaboración Propia)

Pre-Matricula

Fichas de Matricula

Boletas de Notas

Record Academico

Mi cuenta

Pre Matricula Habilitada

Pre-Matricula (3 pasos)

Este formulario esta habilitado hasta el 12/08/2016.

1. Inicio
2. Cursos
3. Finalizar

Seleccione los cursos que desea matricularse

Asignaturas de la Ficha de Matricula

SIGLO	TIPO*	PLAN	NOMBRE DE ASIGNATURA	REQUISITOS	CREDITOS
CC121	Regular	2005	Computación Básica	Ninguno	2
IS141	Regular	2005	Algoritmos	Ninguno	4
MD141	Regular	2005	Método de Estudio Universitario	Ninguno	3
MA141	Regular	2005	Matemática Básica	Ninguno	3
EC141	Regular	2005	Economía	Ninguno	3
LE141	Regular	2005	Comunicación Oral y Escrita	Ninguno	2
MA143	Regular	2005	Cálculo I	Ninguno	5

Nº de Cursos
Serie
Total de Creditos

Cursos Disponibles

	Nombre del Curso	Sigla	Creditos	Requisitos	Disponible
<input checked="" type="checkbox"/>	Computación Básica	CC121	2	Ninguno	✓
<input type="checkbox"/>	Folklore	AC121	2	Ninguno	✓
<input checked="" type="checkbox"/>	Algoritmos	IS141	4	Ninguno	✓

Figura N° 4.24: Prototipo de interfaz gráfica revisada “Registrar Matrícula” [Paso 2] (Elaboración Propia)

CU30: Emitir Registro de Evaluación

Curso

Ing. Sistemas - IS441 - Redes de Computadoras

UNIVERSIDAD NACIONAL DE SAN CRISTOBAL DE HUAMANGA
VICERRECTORADO ACADEMICO - OFICINA DE INFORMATICA Y SISTEMAS
REGISTRO DE EVALUACION PERMANENTE (Regular)
30/08/2016 UNSCH
SEMESTRE: 2016-I
SIGLA: IS441
ASIGNATURA: IS441 REDES DE COMPUTADORAS PLAN: 2005 SEMESTRE: 2016-I CREDITOS: 4.0
PLAN: 2005
ESCUELA PROF: INGENIERIA DE SISTEMAS DEPARTAMENTO: MATEMATICA Y FISICA
CREDITOS: 4.0

No	CODIGO	APELLIDOS Y NOMBRES	Parcial 1	Parcial 2	Practicas	Laboratorio	Trabajo Final	Promedio Final	ApI	CODIGO	NOTA
001	27120524	RAMIREZ PIZARRO, Marvin								27120524	
002	27120548	HUARACA PAREDES, Romel Telesforo								27120548	
003	27125877	ARONES MEDINA, Christian Augusto								27125877	

DOCENTE: KAREL PERALTA SOTOMAYOR

Imprimir

Figura N° 4.27: Prototipo de interfaz gráfica revisada “Emitir Registro de Evaluación” (Elaboración Propia)

CU31: Registrar nota del curso del semestre regular

Curso

Ing. Sistemas - IS441 - Redes de Computadoras

UNIVERSIDAD NACIONAL DE SAN CRISTOBAL DE HUAMANGA
VICERRECTORADO ACADEMICO - OFICINA DE INFORMATICA Y SISTEMAS
REGISTRO DE EVALUACION PERMANENTE (Regular)
31/08/2016 UNSCH
SEMESTRE: 2016-I
SIGLA: IS441
ASIGNATURA: IS441 REDES DE COMPUTADORAS PLAN: 2005 SEMESTRE: 2016-I CREDITOS: 4.0
PLAN: 2005
ESCUELA PROF: INGENIERIA DE SISTEMAS DEPARTAMENTO: MATEMATICA Y FISICA
CREDITOS: 4.0

No	CODIGO	APELLIDOS Y NOMBRES	Parcial 1	Parcial 2	Practicas	Laboratorio	Trabajo Final	Promedio Final	ApI	CODIGO	NOTA
001	27120524	RAMIREZ PIZARRO, Marvin	12	12	14	12	13	13	27120524	13	
002	27120548	HUARACA PAREDES, Romel Telesforo	12	12	13	12	12	12	27120548	12	
003	27125877	ARONES MEDINA, Christian Augusto	13	14	13	15	14	14	27125877	14	

DOCENTE: KAREL PERALTA SOTOMAYOR

Guardar Imprimir

Figura N° 4.28: Prototipo de interfaz gráfica revisada “Registrar nota del curso del semestre regular” (Elaboración Propia)

CU46: Emitir Acta de Evaluación Final del semestre regular

- Calendario Académico
- Apertura de Semestre
- Aperturar Eventos
- Registrar Personas
- Asignar Cursos
- Acta de Evaluación Final
- Reversiones

- Mi cuenta

Seleccione la Escuela Profesional y el Curso.

Escuela:

Curso:

UNIVERSIDAD NACIONAL DE SAN CRISTOBAL DE HUAMANGA
ACTA DE EVALUACION FINAL

FACULTAD DE:	INGENIERIA DE MINAS, GEOLOGIA Y CIVIL	SEMESTRE:	2016-I
ESCUELA PROFESIONAL:	INGENIERIA DE SISTEMAS	PAGINA:	001
DEPARTAMENTO ACADEMICO:	MATEMATICA Y FISICA	SIGLA:	IS441
PLAN DE ESTUDIOS:	2005	CREDITOS:	4.0
A SIGNATURA:	REDES DE COMPUTADORAS	DOCENTE:	NAREL PERALTA SOTOMAYOR
TIPO DE ACTA:	REGULAR	FECHA:	31/08/2016
TURNO:	9564	SECCION:	UNICA
		SERIE:	400

No Orden	CODIGO	APELLIDOS Y NOMBRES	Promedio Final	OBSERVACIONES
001	27120524	RAMIREZ PIZARRO, Marvin		13 TRECE
002	27125877	ARONES MEDINA, Christian Augusto		14 CATORCE
003	27120548	HUARACA PAREDES, Romel Teleforo		12 DOCE

Ayacucho, 31 de agosto del 2016 - 00:29

.....

Prof. de la Asignatura

.....

Jefe de Departamento

.....

Director de Escuela

.....

Decano

[Imprimir](#)

Figura N° 4.29: Prototipo de interfaz gráfica revisada “Emitir Acta de Evaluación Final del semestre regular” (Elaboración Propia)

C. CASO DE USO REVISADO

CASO DE USO	DESCRIPCIÓN
<p>CU23. Registrar Matrícula (Prematrícula)</p>	<p>Curso Básico:</p> <ol style="list-style-type: none"> 1. El estudiante hace clic en la opción “Pre matrícula” de la interface “Página Estudiante”, el sistema muestra la interface “Pre matrícula” con los datos necesarios del estudiante para su matrícula. 2. El estudiante verifica sus datos, y si no es correcto, hace clic en la opción "Reportar Error", el sistema muestra una ventana, donde el estudiante registra el error, luego hace clic en el botón "Enviar", el sistema envía y guarda el reporte de error. 3. El estudiante verifica sus datos, y si es correcto, hace clic en el botón siguiente, el sistema valida, guarda sus datos, muestra un mensaje de confirmación "Se guardó correctamente los datos", y luego muestra los cursos disponibles para la matrícula. 4. El estudiante selecciona el curso, el sistema agrega y registra el curso seleccionado a la tabla de la ficha de matrícula.

	<p>5. El estudiante hace clic en siguiente, el sistema incluye el caso de uso Emitir Esquela de Pago y al caso de uso Emitir Ficha de Matrícula y los muestra en la interface "Pre matrícula".</p> <p>Curso Alternativo:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error, "Error de Validación". 2. El sistema muestra un mensaje de error, "Se excede la cantidad de crédito a matricularse". 3. El sistema muestra un mensaje de error, "No se registró correctamente los cursos matriculados".
--	---

Tabla N° 4.13: Primer Borrador de caso de uso revisado "Registrar Matrícula" (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU22: Emitir Esquela de Pago de matrícula regular</p>	<p>Curso Básico:</p> <ol style="list-style-type: none"> 1. El estudiante hace clic en la opción "Emitir Esquela de Pago de matrícula regular" de la interface "Página Estudiante", el sistema muestra la interface "Esquela de Pago". 2. El sistema muestra la Esquela de Pago del semestre correspondiente. 3. El estudiante hace clic en el botón "Imprimir", el sistema obtiene la Esquela de Pago impresa. <p>Curso Alternativo:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error, "No se puede generar Esquela de Pago del semestre académico correspondiente". 2. El sistema muestra un mensaje de error, "No se puede imprimir Esquela de Pago".

Tabla N° 4.14: Primer Borrador de caso de uso revisado "Emitir Esquela de Pago de matrícula regular" (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU24: Emitir Ficha de Matrícula</p>	<p>Curso Básico:</p> <ol style="list-style-type: none"> 1. El estudiante hace clic en la opción “Emitir Ficha de Matrícula” de la interface “Página Estudiante”, el sistema muestra la interface “Ficha de Matrícula” con el semestre académico a seleccionar. 2. El estudiante selecciona el semestre académico, el sistema muestra la ficha de matrícula del semestre seleccionado. 3. El estudiante hace clic en el botón “Imprimir”, el sistema obtiene la Ficha de Matrícula impresa. <p>Curso Alterno:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error, “No se puede generar ficha de matrícula de semestre académico seleccionado”. 2. El sistema muestra un mensaje de error, “No se puede imprimir Ficha de Matrícula”.

Tabla N° 4.15: Primer Borrador de caso de uso revisado “Emitir Ficha de Matrícula” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU30: Emitir Registro de Evaluación</p>	<p>Curso Básico:</p> <ol style="list-style-type: none"> 1. El docente hace clic en la opción “Emitir Registro de Evaluación” de la interface “Página Docente”, el sistema muestra la interface “Registro de Evaluación” con la escuela - curso a seleccionar. 2. El docente selecciona la escuela - curso, el sistema muestra el Registro de Evaluación. 3. El docente hace clic en el botón “Imprimir”, el sistema obtiene el Registro de Evaluación impreso. <p>Curso Alterno:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error, “No se puede generar Registro de Evaluación”.

	2. El sistema muestra un mensaje de error, “No se puede imprimir Registro de Evaluación” .
--	---

Tabla N° 4.16: Primer Borrador de caso de uso revisado “Emitir Registro de Evaluación” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
CU31. Registrar nota del curso del semestre regular	<p>Curso Básico:</p> <ol style="list-style-type: none"> 1. El docente hace clic en la opción “Registrar nota del curso del semestre regular” de la interface “Página Docente”, el sistema muestra la interface “Registro nota de curso del semestre regular” con la escuela - curso a seleccionar. 2. El docente selecciona la escuela - curso, el sistema muestra el Registro de Evaluación con los datos correspondientes. 3. El docente registra las notas en el Registro de Evaluación y hace clic en el botón “Guardar”, el sistema valida, guarda las notas registradas y muestra un mensaje de confirmación “Se guardó correctamente las notas registradas”. 4. El docente hace clic en el botón “Imprimir”, el sistema obtiene el Registro de Evaluación impreso. <p>Curso Alternativo:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error, “No se muestra el Registro de Evaluación”. 2. El sistema muestra un mensaje de error, "Error de validación". 3. El sistema muestra un mensaje de error, "No se guardó correctamente las notas registradas". 4. El sistema muestra un mensaje de error, "No se puede imprimir Registro de Evaluación".

Tabla N° 4.17: Primer Borrador de caso de uso revisado “Registrar nota del curso del semestre regular” (Elaboración Propia)

CASO DE USO	DESCRIPCIÓN
<p>CU46. Emitir Acta de Evaluación Final del semestre regular</p>	<p>Curso Básico:</p> <ol style="list-style-type: none"> 1. El personal del OGIS hace clic en la opción “Emitir Acta de Evaluación Final del semestre regular” de la interface “Página Personal del OGIS”, el sistema muestra la interface “Acta de Evaluación Final” con la escuela - curso a seleccionar. 2. El personal del OGIS selecciona la escuela - curso, el sistema muestra el Acta de Evaluación Final del semestre regular. 3. El personal del OGIS hace clic en el botón “Imprimir”, el sistema obtiene el Acta de Evaluación Final del semestre regular impreso. <p>Curso Alternativo:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error, “No se puede generar Acta de Evaluación Final del semestre regular”. 2. El sistema muestra un mensaje de error, “No se puede imprimir Acta de Evaluación Final”.

Tabla N° 4.18: Primer Borrador de caso de uso revisado “Emitir Acta de Evaluación Final del semestre regular” (Elaboración Propia)

433. DISEÑO PRELIMINAR

A. CASOS DE USO DESAMBIGUADO Y DIAGRAMA DE ROBUSTEZ

CU23: Registrar Matrícula (Prematrícula)

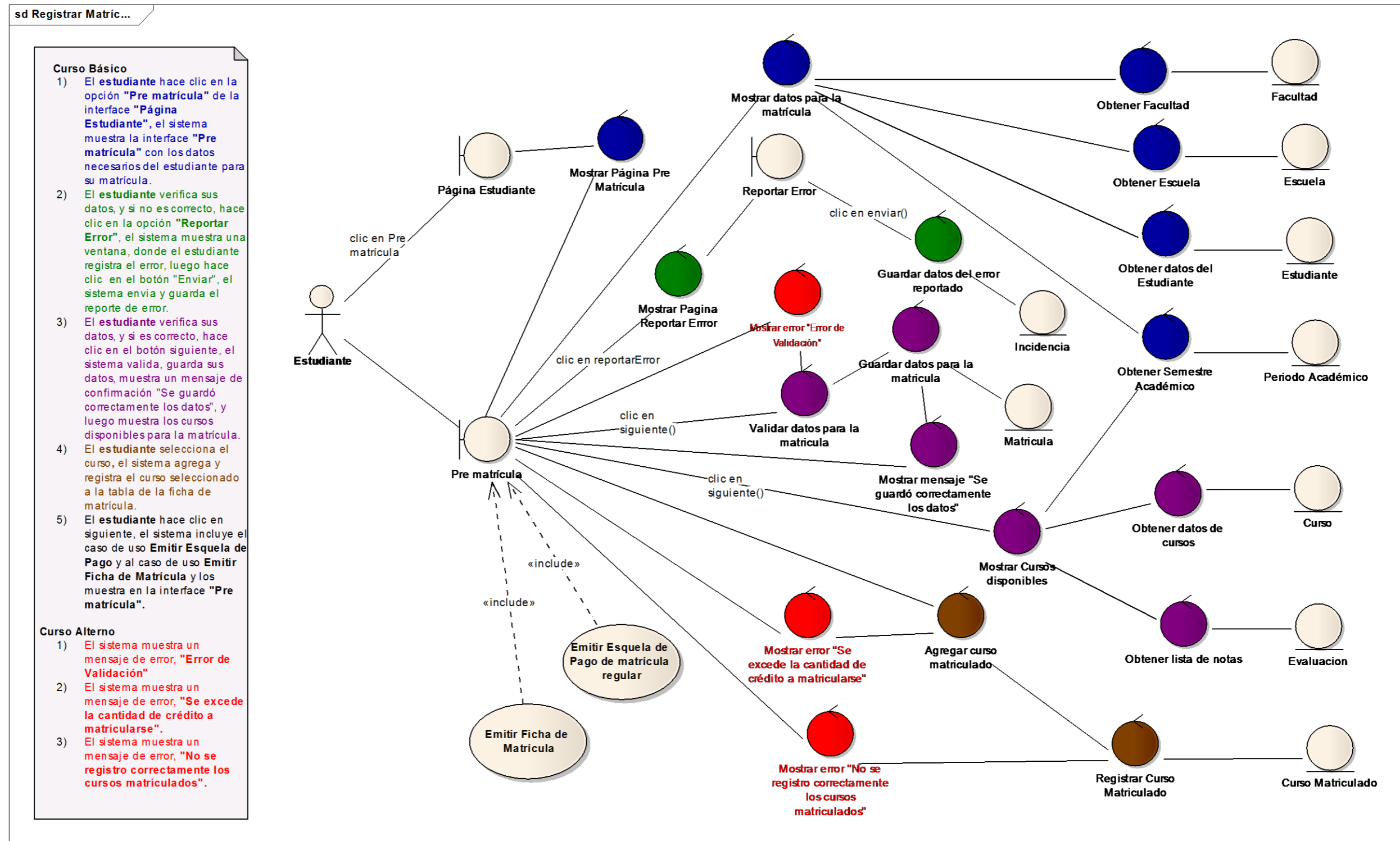


Figura N° 4.30: Diagrama de Robustez, caso de uso "Registrar Matrícula" (Elaboración Propia)

CU22: Emitir Esquela de Pago de matrícula regular

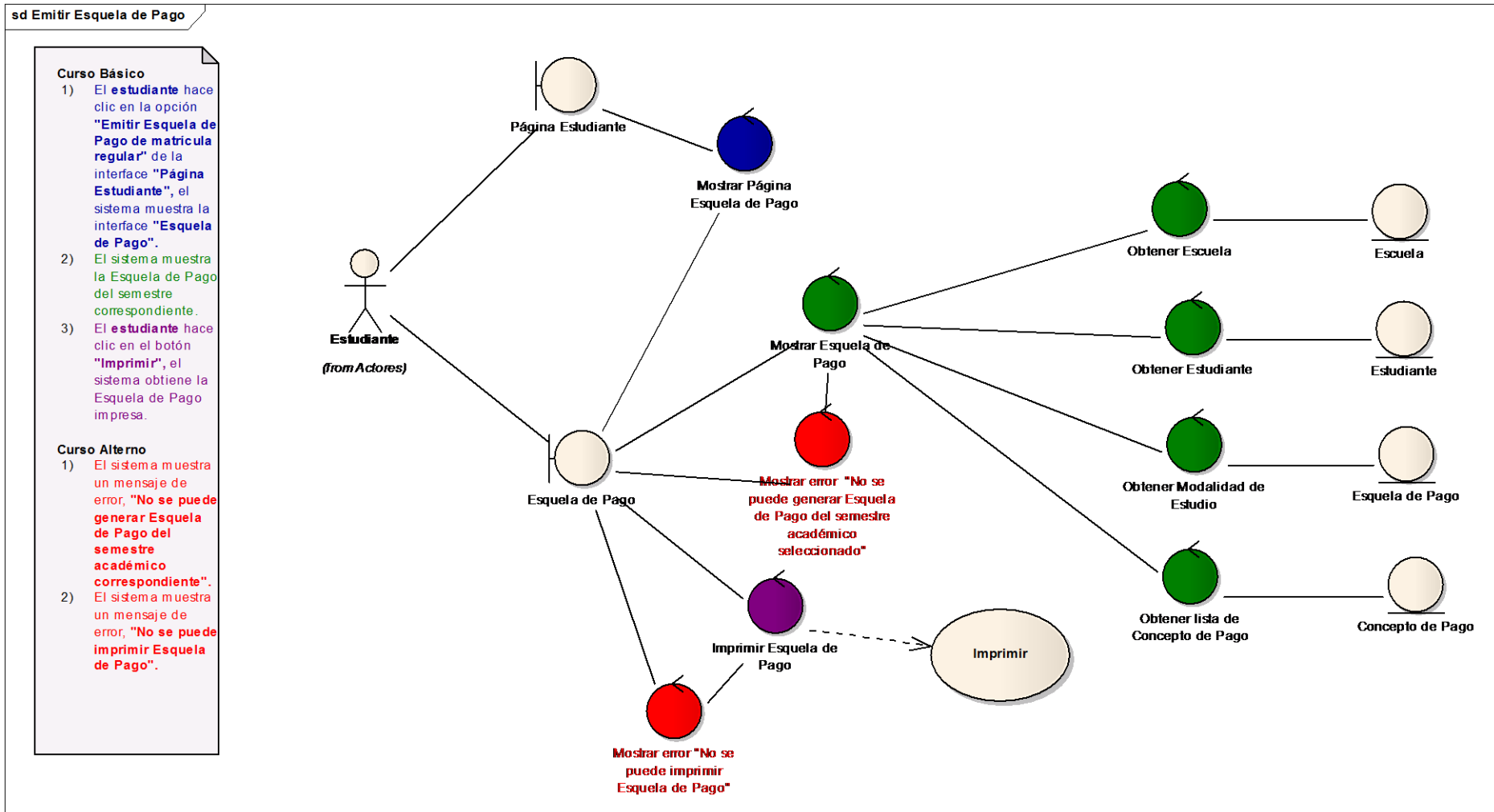


Figura N° 4.31: Diagrama de Robustez, caso de uso "Emitir Esquela de Pago de matrícula regular" (Elaboración Propia)

CU24: Emitir Ficha de Matrícula

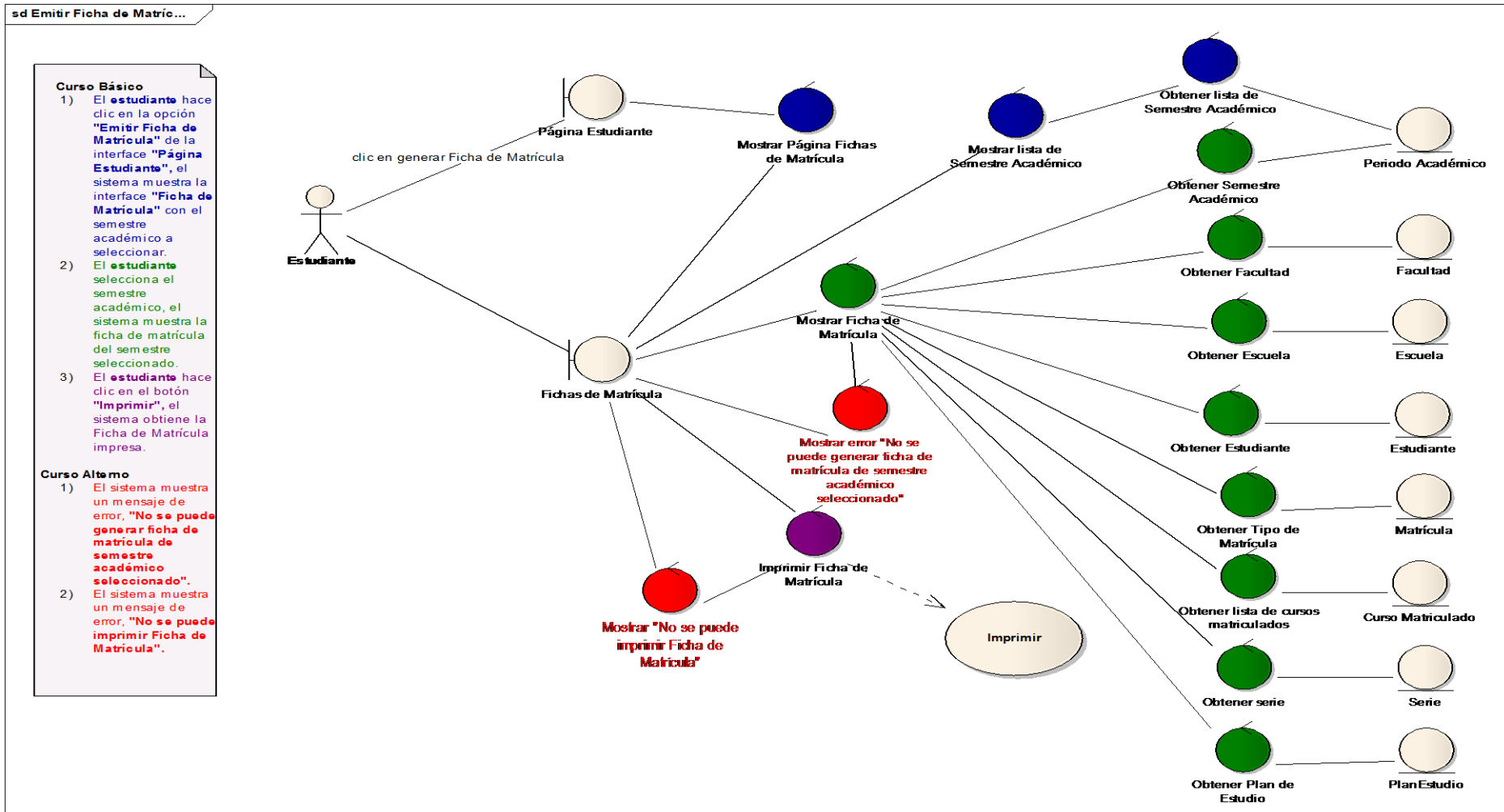


Figura N° 4.32: Diagrama de Robustez, caso de uso "Emitir Ficha de Matrícula" (Elaboración propia)

CU30: Emitir Registro de Evaluación

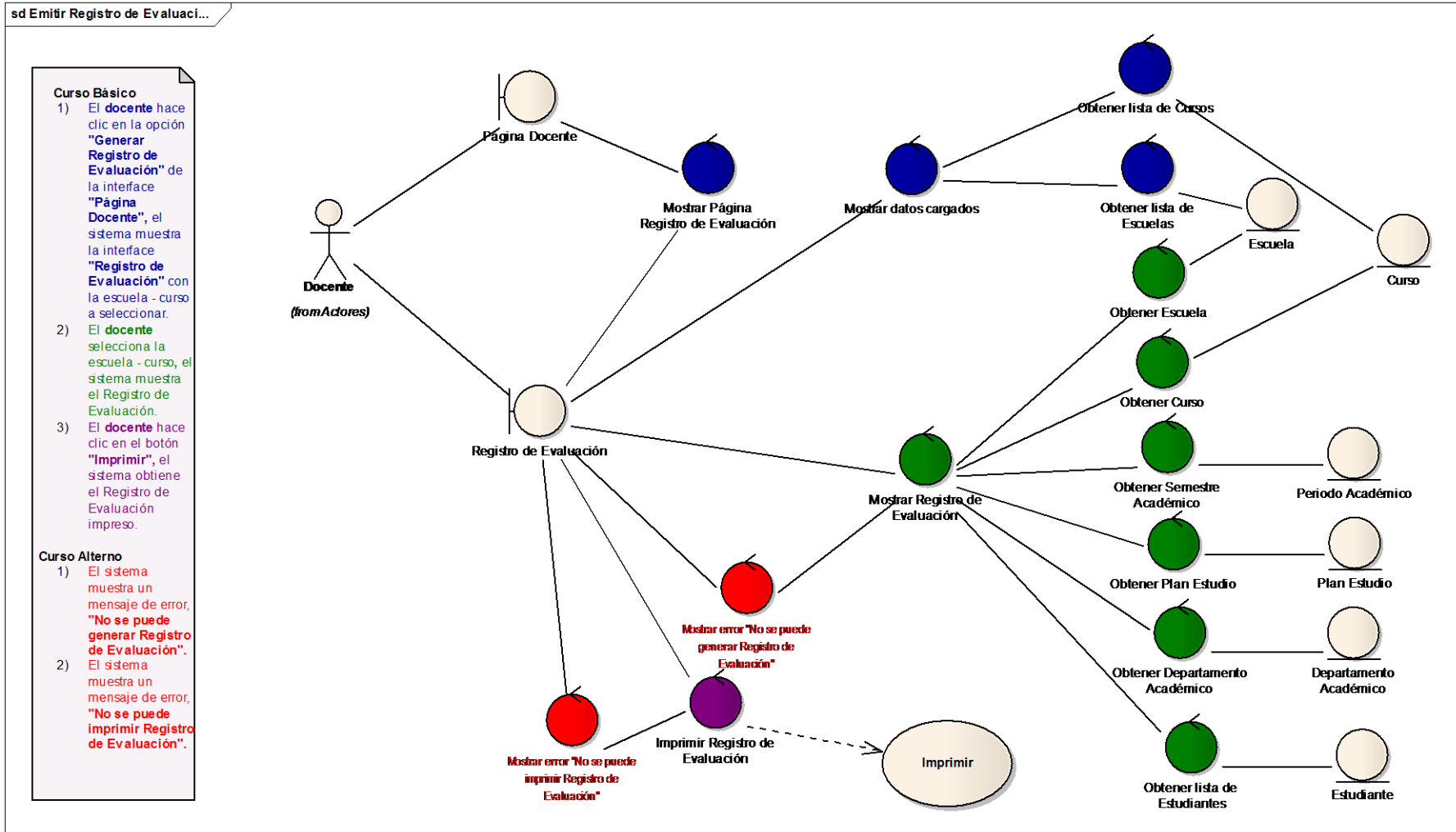


Figura Nº 4.33: Diagrama de Robustez, caso de uso "Emitir Registro de Evaluación" (Elaboración Propia)

CU31: Registrar nota del curso del semestre regular

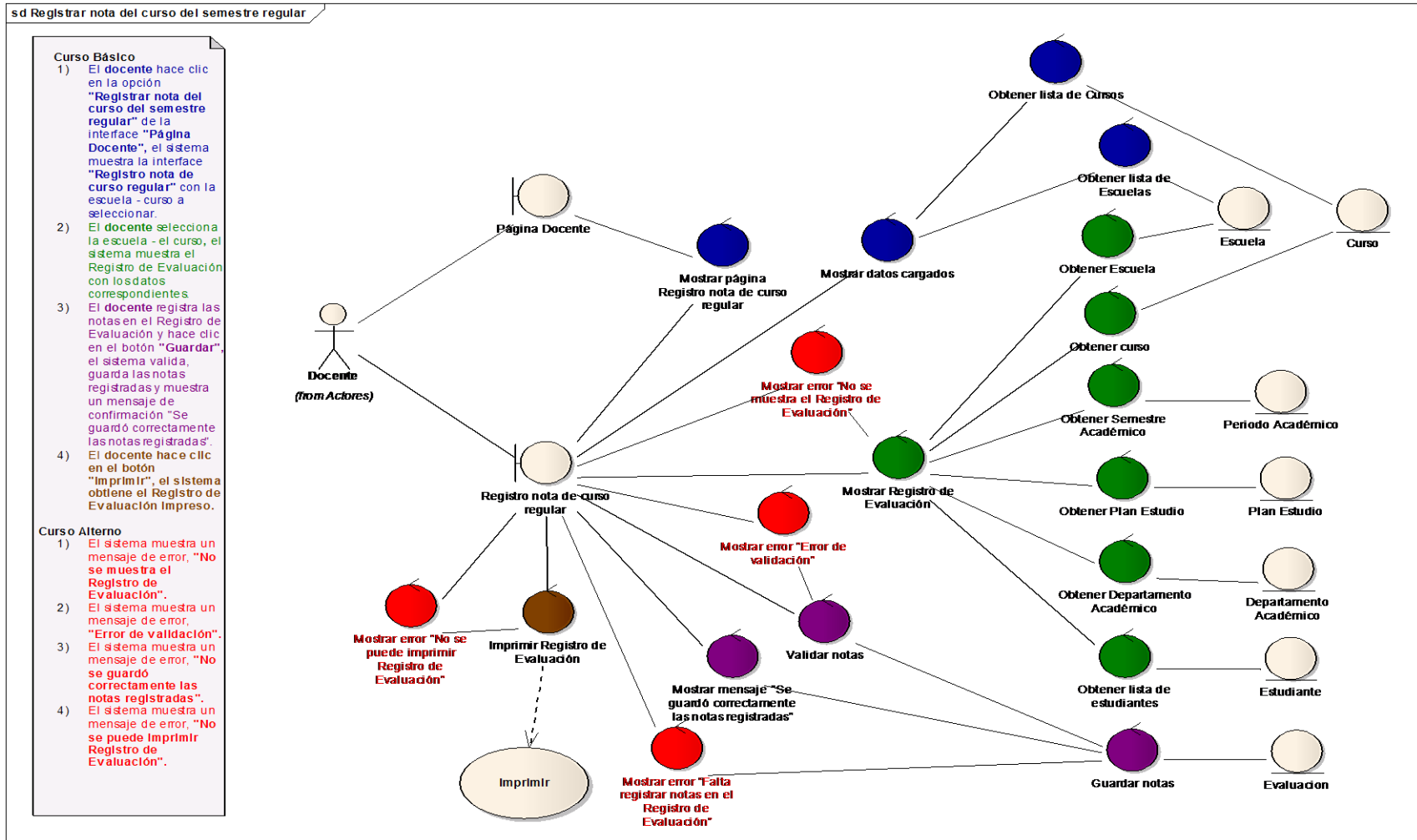


Figura N° 4.34: Diagrama de Robustez, caso de uso "Registrar nota del curso del semestre regular" (Elaboración Propia)

CU46: Emitir Acta de Evaluación Final del semestre regular

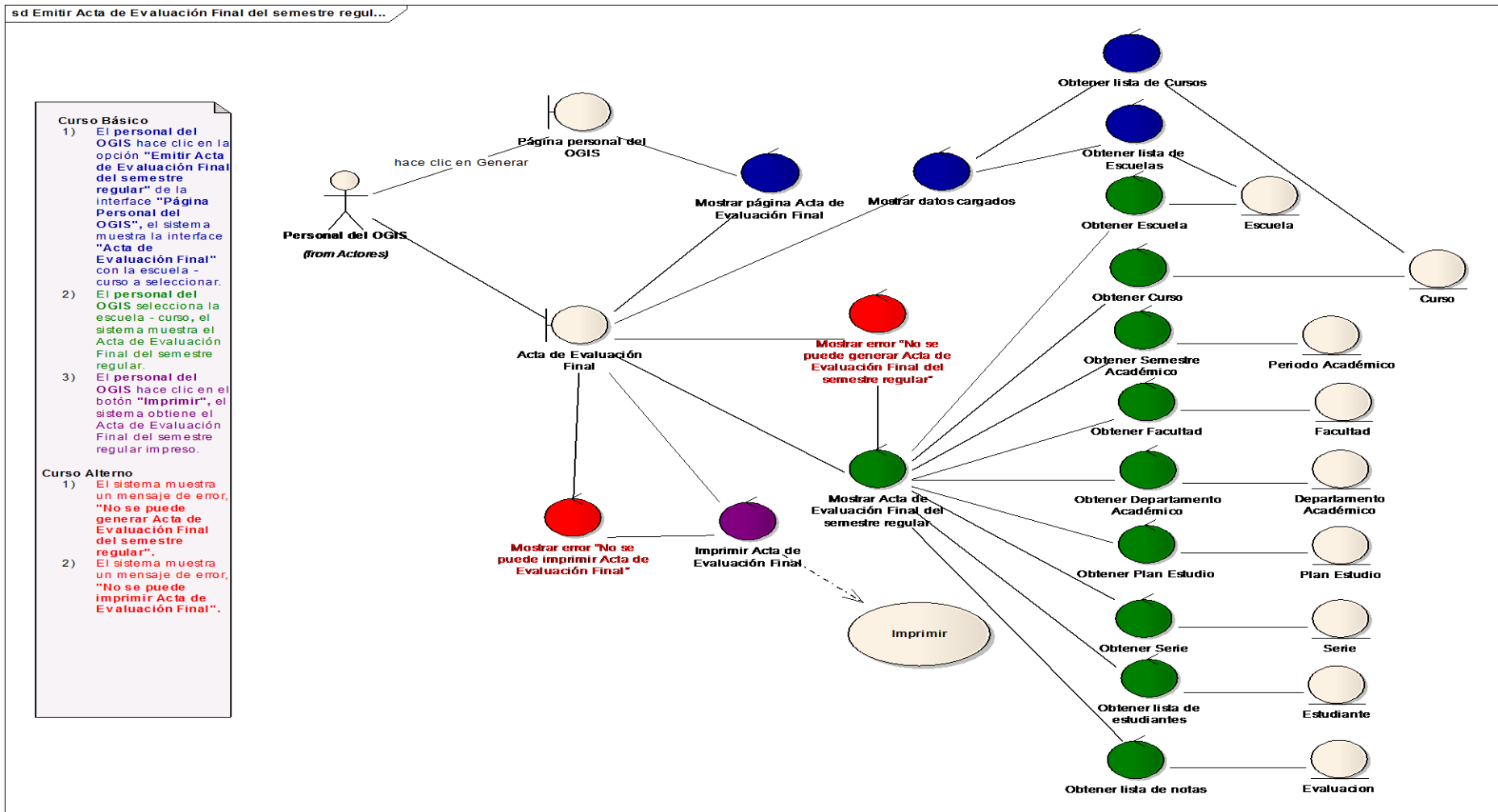


Figura N° 4.35: Diagrama de Robustez, caso de uso "Emitir Acta de Evaluación Final del semestre regular" (Elaboración Propia)

4.3.4. REVISION DEL DISEÑO PRELIMINAR

Durante la revisión no se encontró cambios en los artefactos del modelado del software, para los seis casos de uso.

4.3.5. ARQUITECTURA TÉCNICA

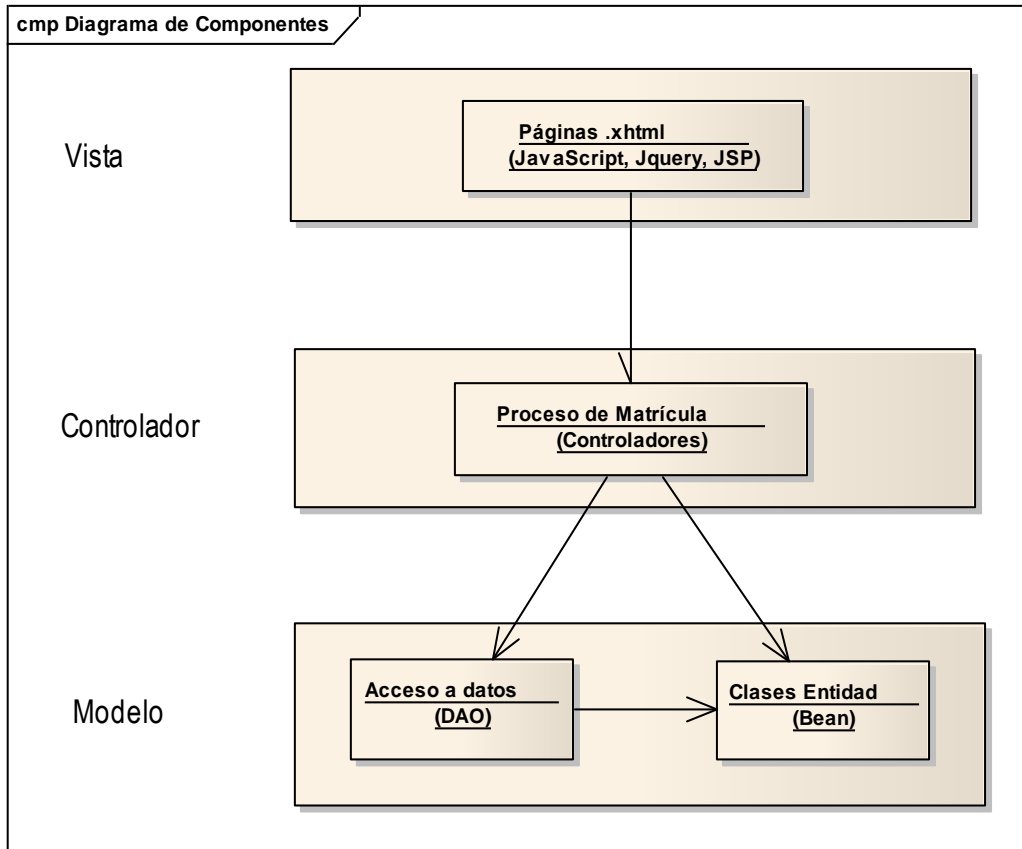


Figura N° 4.37: Arquitectura Técnica. Diagrama de componentes (Elaboración Propia)

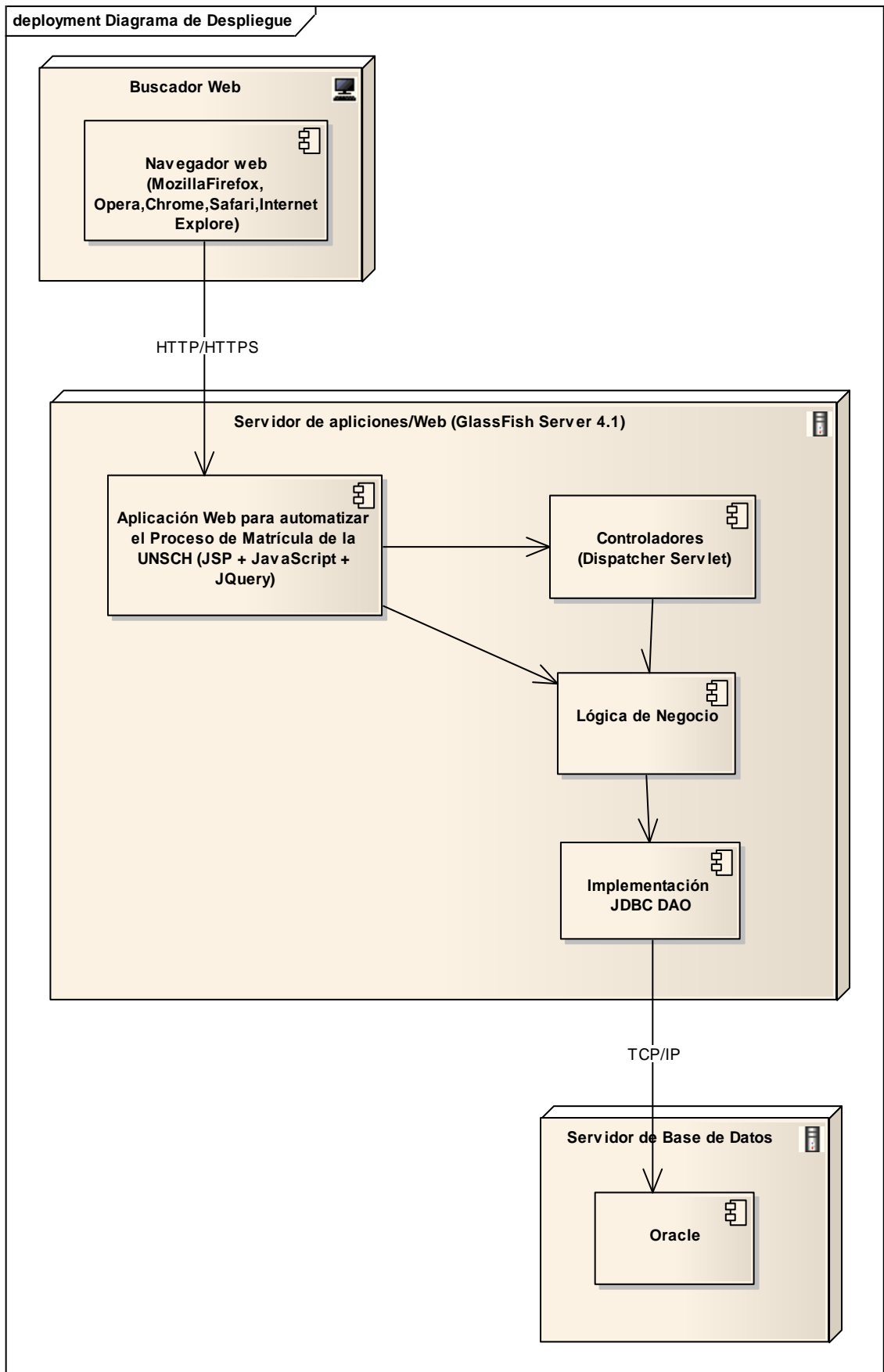


Figura N° 4.38: Arquitectura Técnica. Diagrama de Despliegue (Elaboración Propia)

43.6. DISEÑO DETALLADO

A. DIAGRAMA DE SECUENCIA

CU23: Registrar Matrícula (Prematrícula)

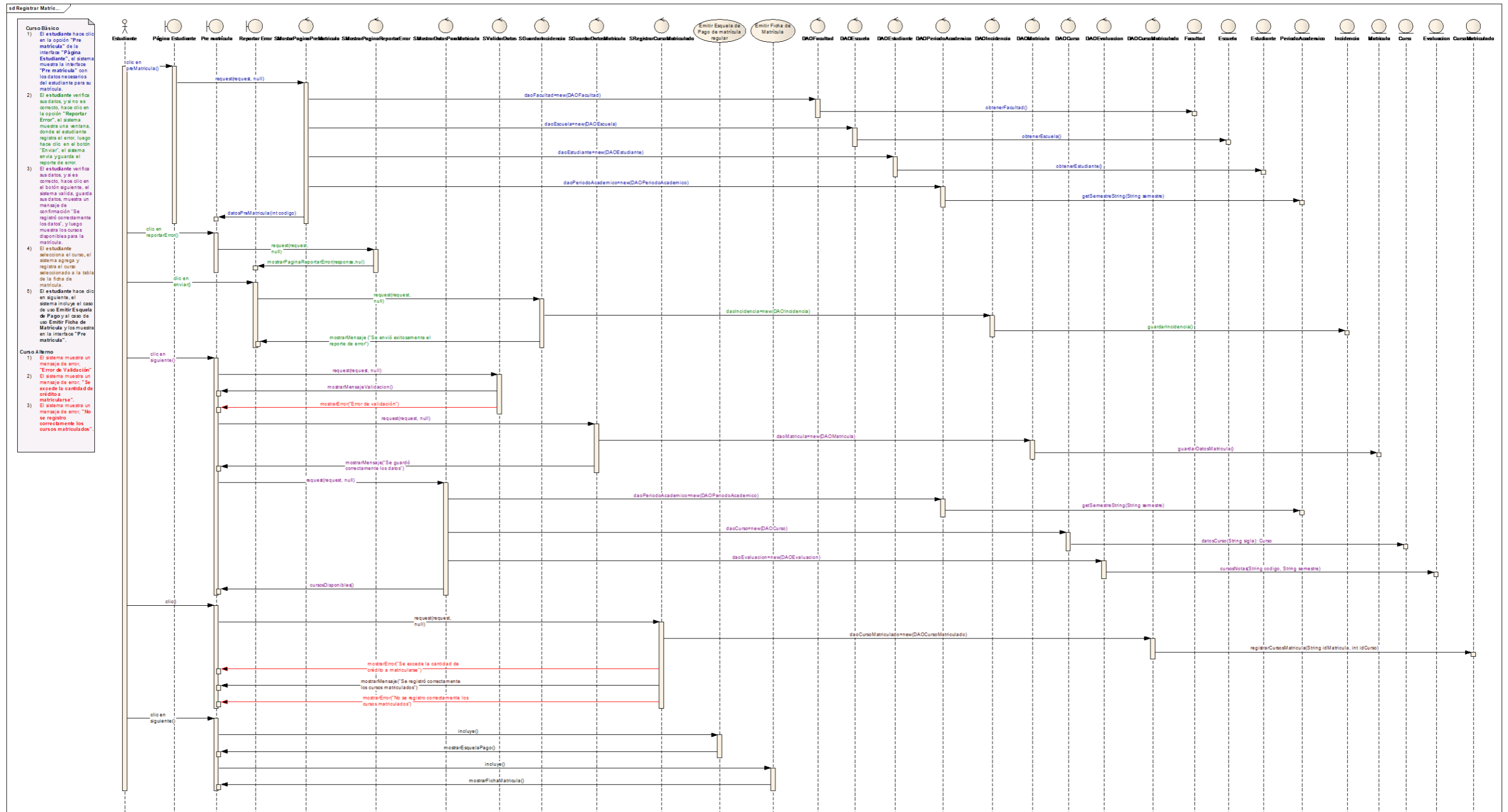


Figura Nº 4.39: Diagrama de Secuencia, caso de uso "Registrar Matrícula" (Elaboración Propia)

CU2: Emitir Esquela de Pago de matrícula regular

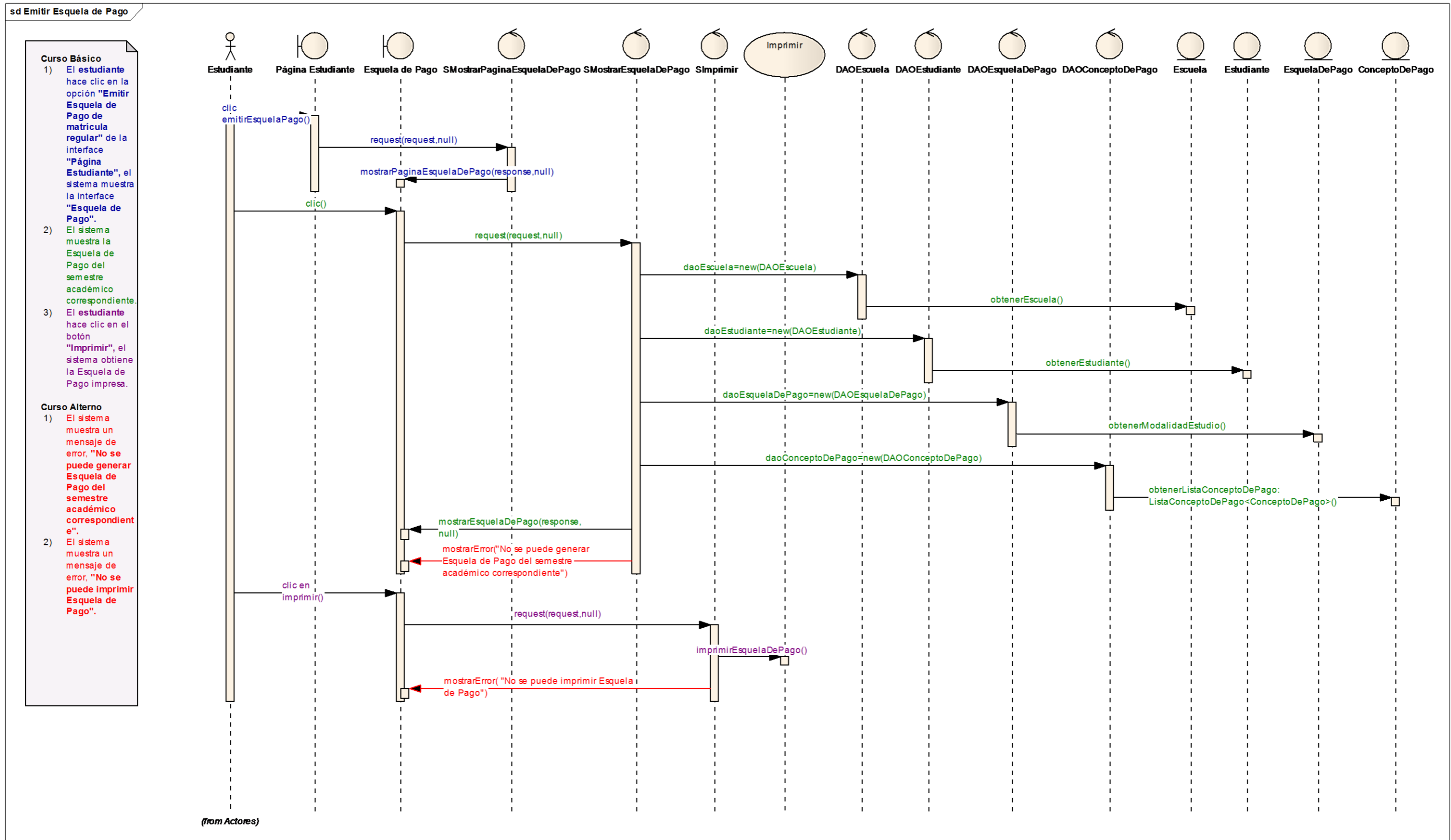


Figura N° 4.40: Diagrama de Secuencia, caso de uso 'Emitir Esquela de Pago de matrícula regular' (Elaboración Propia)

CU24: Emitir Ficha de Matrícula

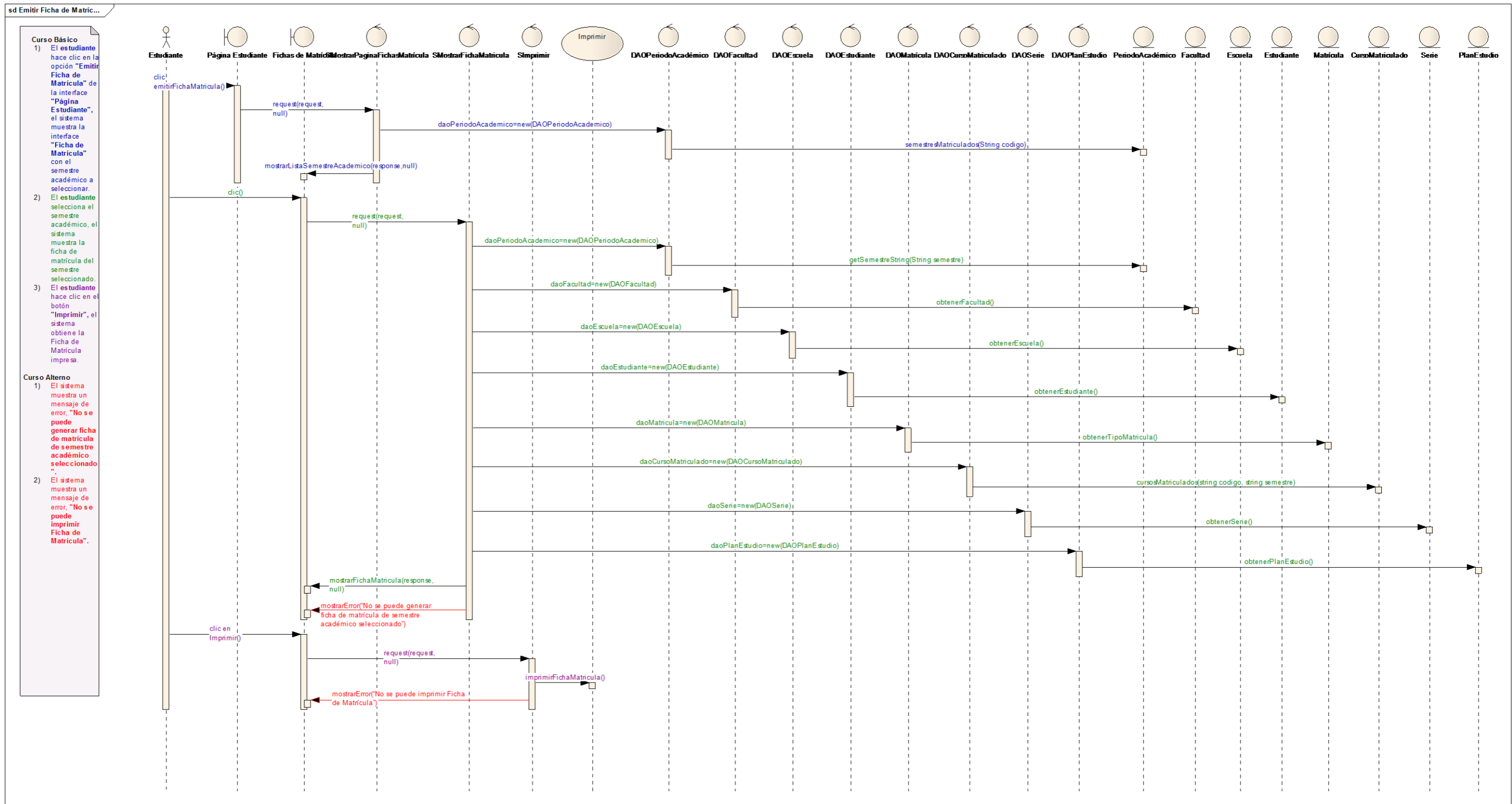


Figura N° 4.41: Diagrama de Secuencia, caso de uso 'Emitir Ficha de Matrícula' (Elaboración Propia)

CU30: Emitir Registro de Evaluación

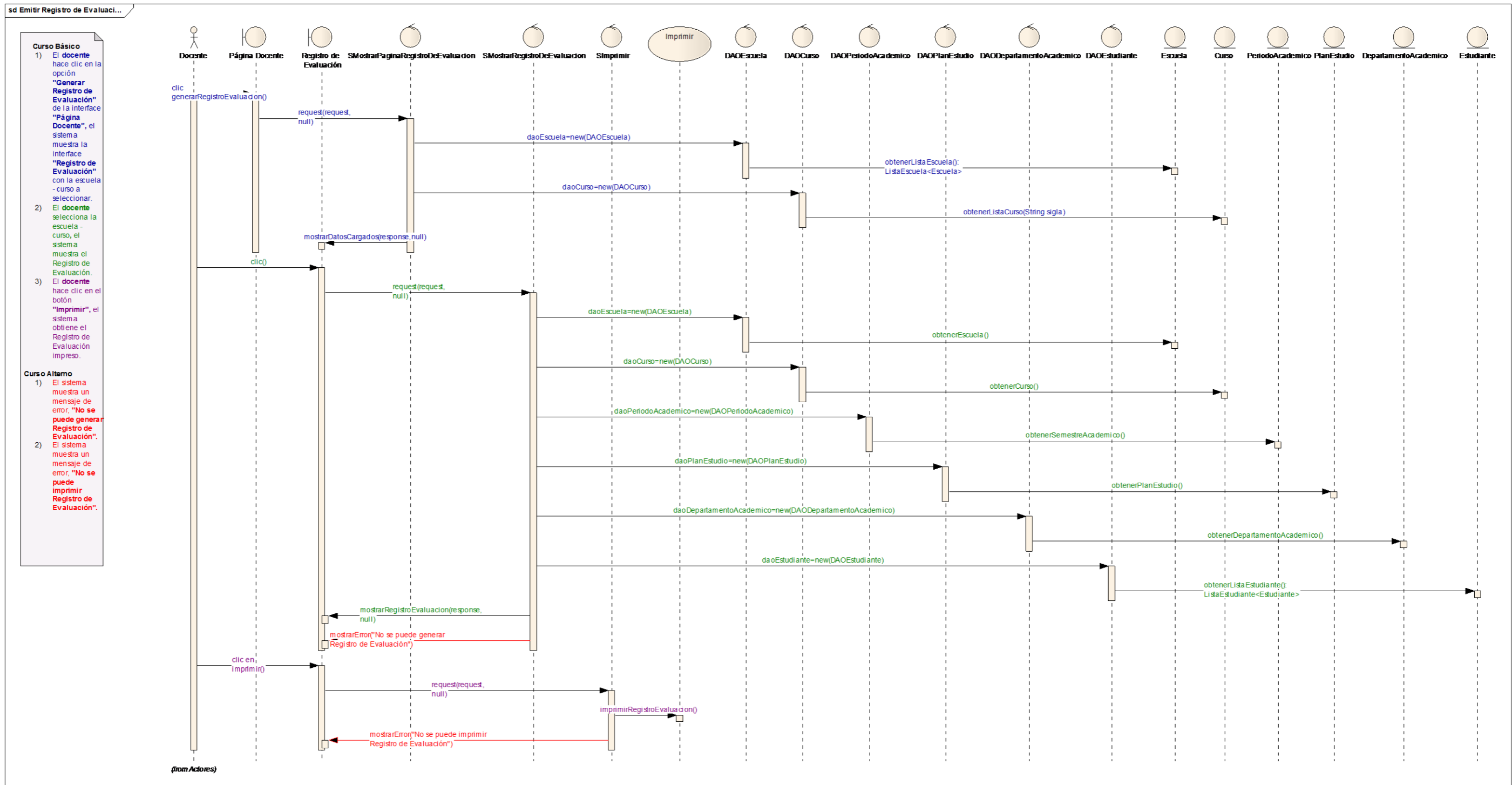


Figura N° 4.42: Diagrama de Secuencia, caso de uso 'Emitir Registro de Evaluación' (Elaboración Propia)

CU31: Registrar nota del curso del semestre regular

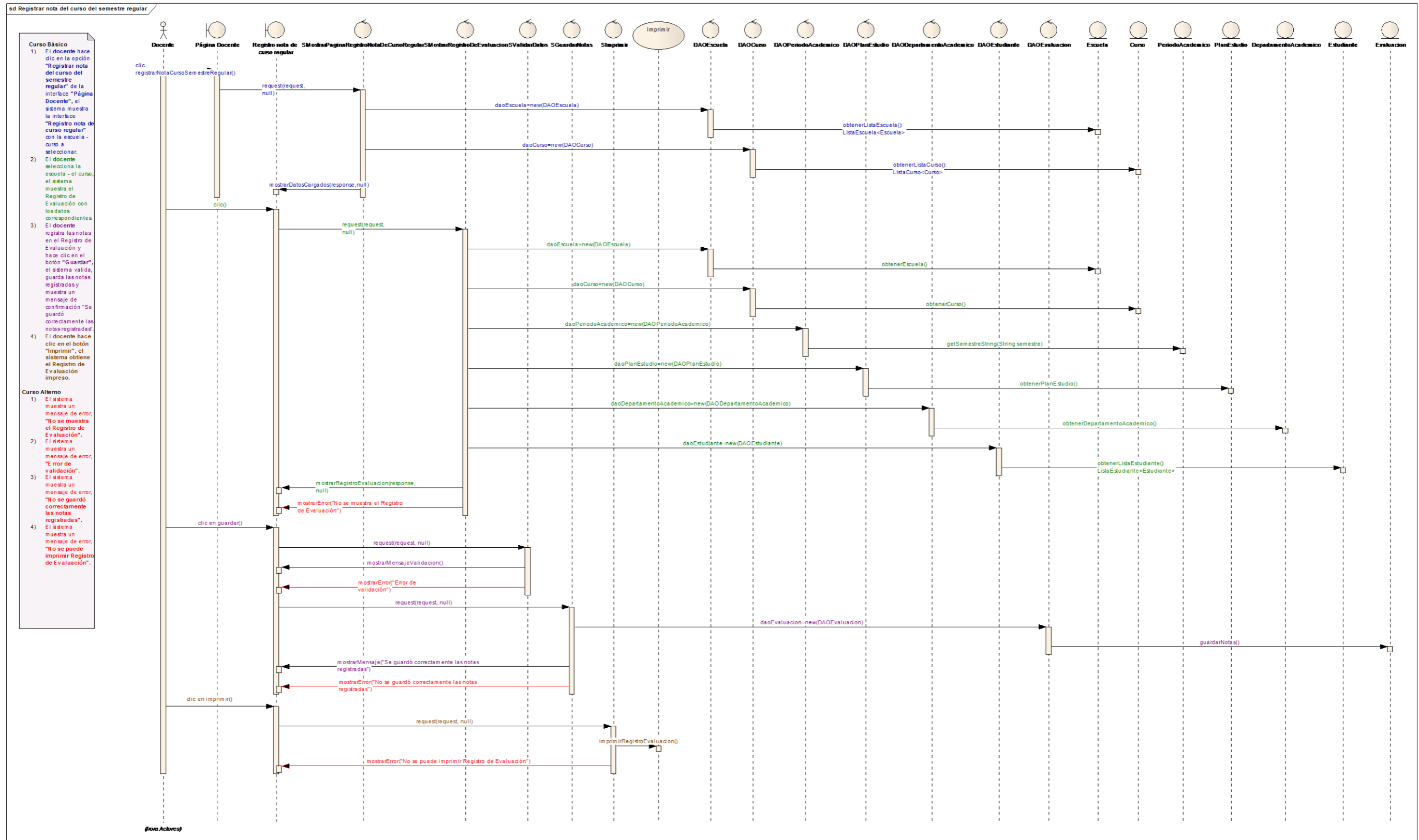


Figura N° 4.43: Diagrama de Secuencia, caso de uso "Registrar nota del curso del semestre regular" (Elaboración Propia)

CU46: Emitir Acta de Evaluación Final del semestre regular

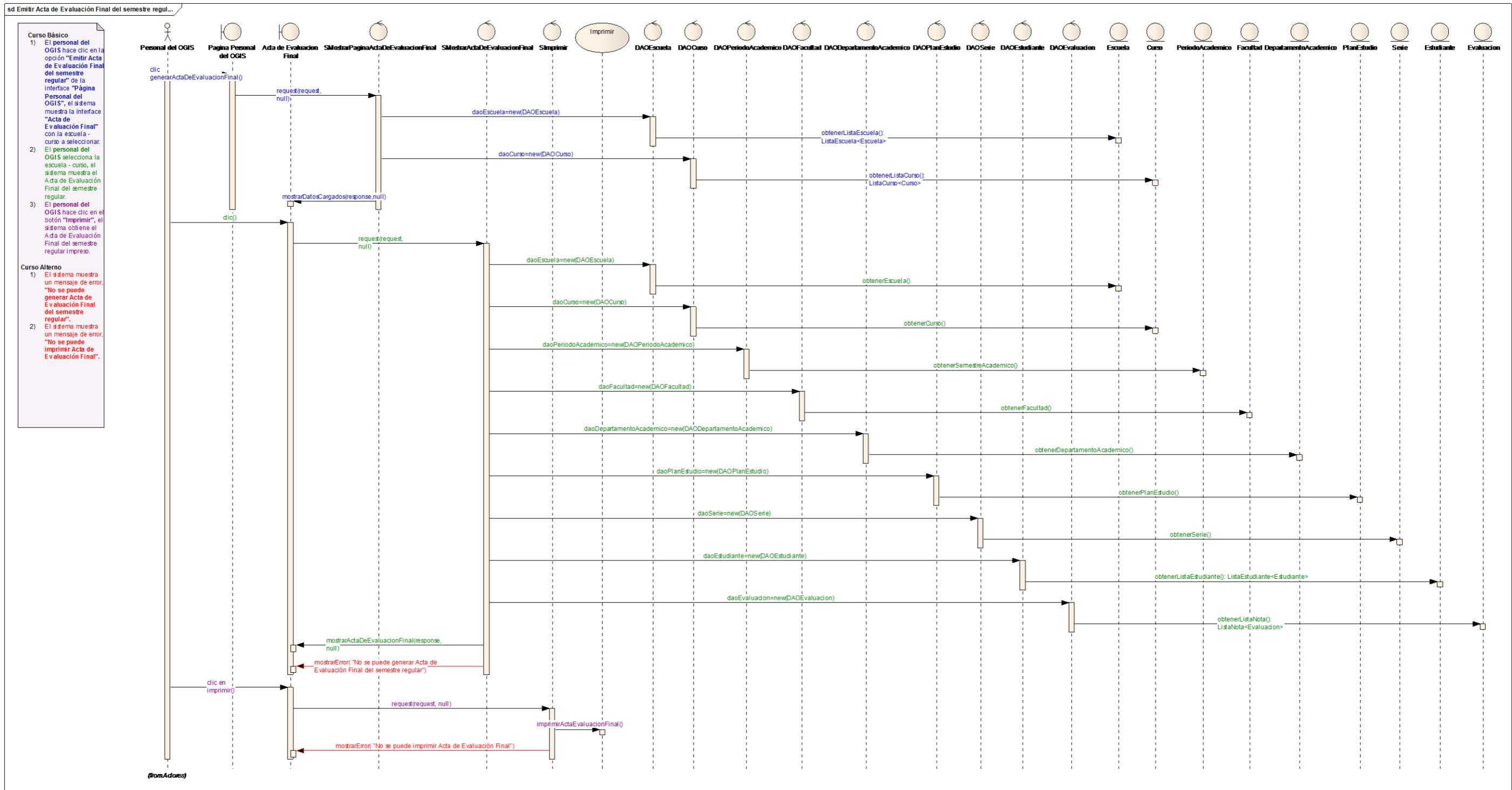


Figura Nº 4.44: Diagrama de Secuencia, caso de uso 'Emitir Acta de Evaluación Final del semestre regular' (Elaboración Propia)

C. LISTA DE CONTROLADORES

N°	NOMBRE CLASE CONTROL
1	SMostrarPaginaPreMatricula.java
2	SMostarPaginaReportarError.java
3	SMostrarDatosParaMatricula.java
4	SGuardarIncidencia.java
5	SGuardarDatosMatricula.java
6	SRegistrarCursoMatriculado.java
7	SMostrarEsquelaDePago.java
8	SMostrarFichaMatricula.java
9	SMostrarRegistroDeEvaluacion.java
10	SGuardarNotas.java
11	SMostrarActaDeEvaluacionFinal.java
12	SImprimir.java

Tabla N° 4.19: Lista de Controladores (Elaboración Propia)

4.3.7. REVISION CRÍTICA DE DISEÑO

Durante la revisión no se encontró cambios en los artefactos del modelado del software, para los seis casos de uso.

438. IMPLEMENTACIÓN

A. DIAGRAMA FÍSICO DE LA BASE DE DATOS

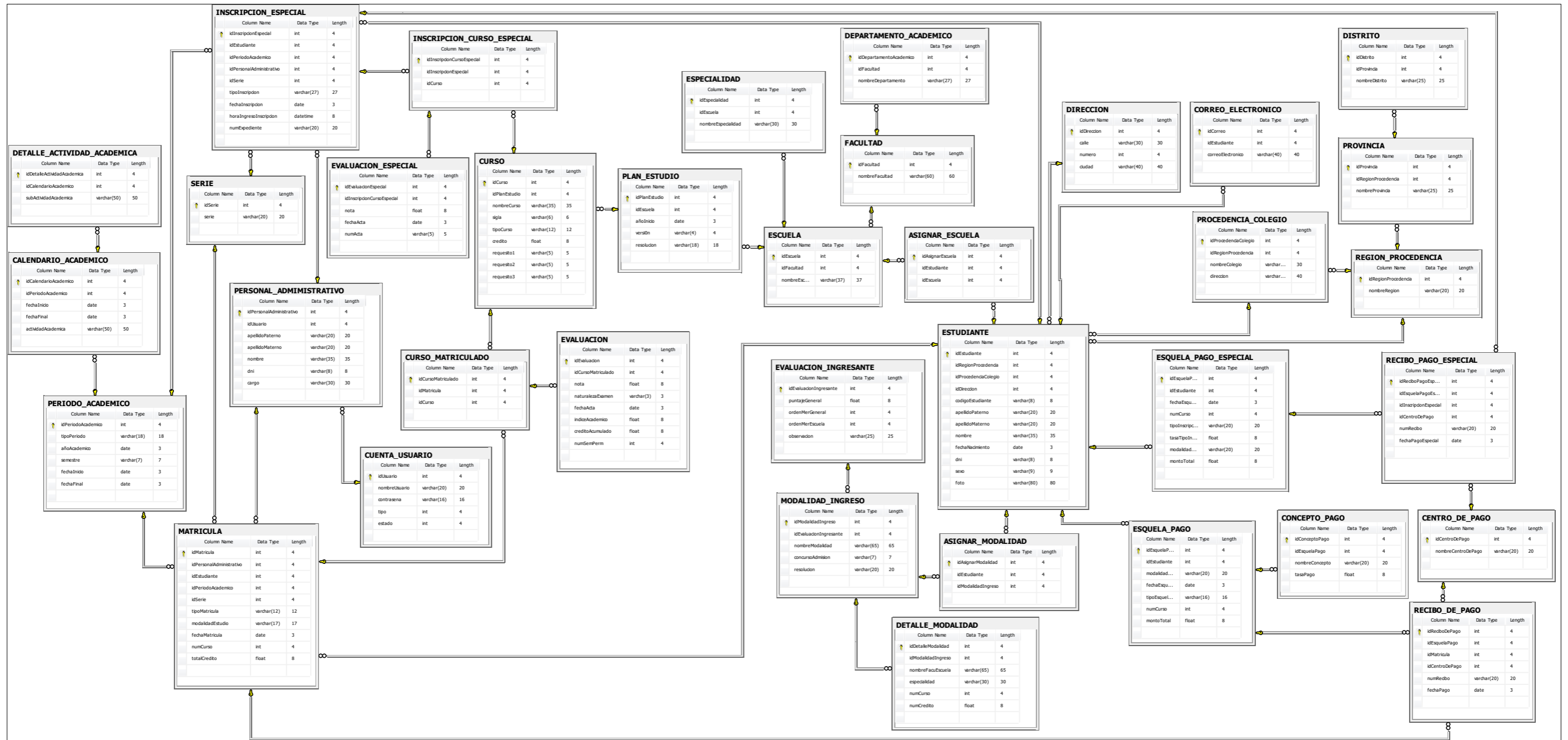


Figura Nº 4.46: Diagrama Físico de la Base de Datos (Elaboración Propia)

B. CÓDIGO SQL PARA TABLAS ENTIDAD

```
CREATE TABLE ESTUDIANTE (  
    codigoEstudiante          number (8) NOT NULL,  
    apellidoPaterno           varchar (20),  
    apellidoMaterno           varchar (20),  
    nombre                    varchar (35),  
    fechaNacimiento           date,  
    dni                       varchar (8),  
    sexo                      varchar (9),  
    foto                      varchar (80));  
commit;  
  
ALTER TABLE ESTUDIANTE  
    ADD PRIMARY KEY (codigoEstudiante);  
commit;
```

Figura N° 4.47: Código SQL para la tabla “Estudiante” (Elaboración Propia)

```
CREATE TABLE MATRICULA(  
    idMatricula               number NOT NULL,  
    idPersonalAdministrativo   number NOT NULL,  
    codigoEstudiante           number (8) NOT NULL,  
    idPeriodoAcademico         number NOT NULL,  
    idSerie                   number NOT NULL,  
    tipoMatricula              varchar (12),  
    modalidadEstudio           varchar (17),  
    fechaMatricula             date,  
    numCurso                   number,  
    totalCredito               number);  
commit;  
  
ALTER TABLE MATRICULA  
    ADD PRIMARY KEY (idMatricula);  
commit;  
  
ALTER TABLE MATRICULA  
    ADD FOREIGN KEY (codigoEstudiante)  
    REFERENCES ESTUDIANTE;  
commit;  
  
ALTER TABLE MATRICULA  
    ADD FOREIGN KEY (idPeriodoAcademico)  
    REFERENCES PERIODO_ACADEMICO;  
commit;  
  
ALTER TABLE MATRICULA  
    ADD FOREIGN KEY (idPersonalAdministrativo)  
    REFERENCES PERSONAL_ADMINISTRATIVO;  
commit;  
  
ALTER TABLE MATRICULA  
    ADD FOREIGN KEY (idSerie)  
    REFERENCES SERIE;  
commit;
```

Figura N° 4.48: Código SQL para la tabla “Matrícula” (Elaboración Propia)

```

CREATE TABLE CURSO(
    idCurso                number NOT NULL,
    idPlanEstudio          number NOT NULL,
    nombreCurso            varchar(50),
    sigla                  varchar(6),
    tipoCurso              varchar(12),
    credito                number(10,1),
    requisito1             varchar(5) NULL,
    requisito2             varchar(5) NULL,
    requisito3             varchar(5) NULL);
commit;

ALTER TABLE CURSO
    ADD PRIMARY KEY (idCurso);
commit;

ALTER TABLE CURSO
    ADD FOREIGN KEY (idPlanEstudio)
    REFERENCES PLAN_ESTUDIO;
commit;

```

Figura N° 4.49: Código SQL para la tabla “Curso” (Elaboración Propia)

```

CREATE TABLE CURSO_MATRICULADO(
    idCursoMatriculado    number NOT NULL,
    idMatricula            number NOT NULL,
    idCurso                number NOT NULL);
commit;

ALTER TABLE CURSO_MATRICULADO
    ADD PRIMARY KEY (idCursoMatriculado);
commit;

ALTER TABLE CURSO_MATRICULADO
    ADD FOREIGN KEY (idCurso)
    REFERENCES CURSO;
commit;

ALTER TABLE CURSO_MATRICULADO
    ADD FOREIGN KEY (idMatricula)
    REFERENCES MATRICULA;
commit;

```

Figura N° 4.50: Código SQL para la tabla “Curso Matriculado” (Elaboración Propia)

```

CREATE TABLE EVALUACION(
    idEvaluacion                number NOT NULL,
    idCursoMatriculado          number NOT NULL,
    nota                        number (2),
    naturalezaExamen            varchar (3),
    fechaActa                   date,
    indiceAcademico             number,
    creditoAcumulado           number,
    numSemPerm                  number);
commit;

ALTER TABLE EVALUACION
    ADD PRIMARY KEY (idEvaluacion);
commit;

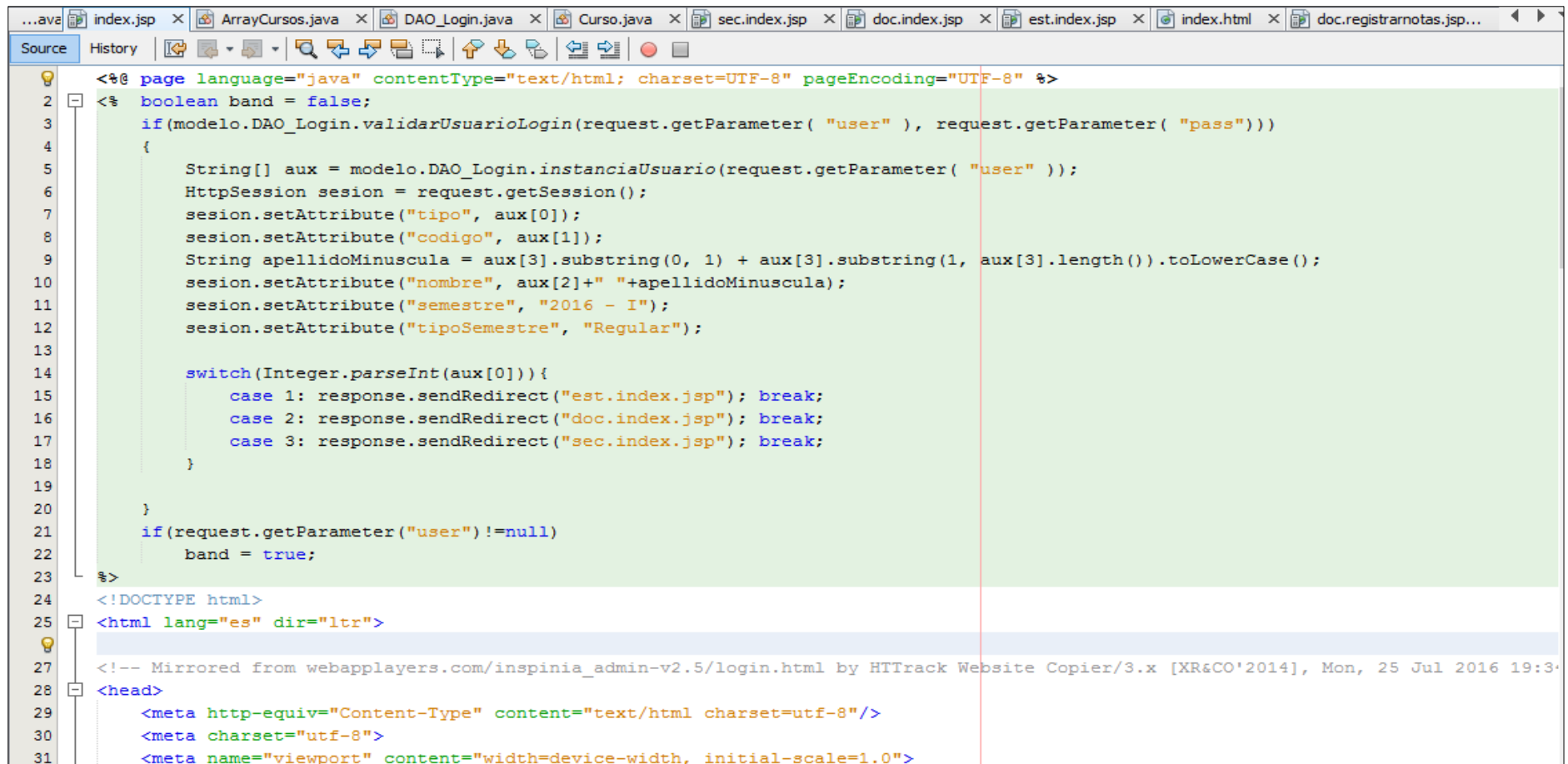
ALTER TABLE EVALUACION
    ADD FOREIGN KEY (idCursoMatriculado)
    REFERENCES CURSO_MATRICULADO;
commit;

```

Figura N° 4.51: Código SQL para la tabla “Evaluacion” (Elaboración Propia)

C. CODIGO FUENTE PARA LAS INTERFACES

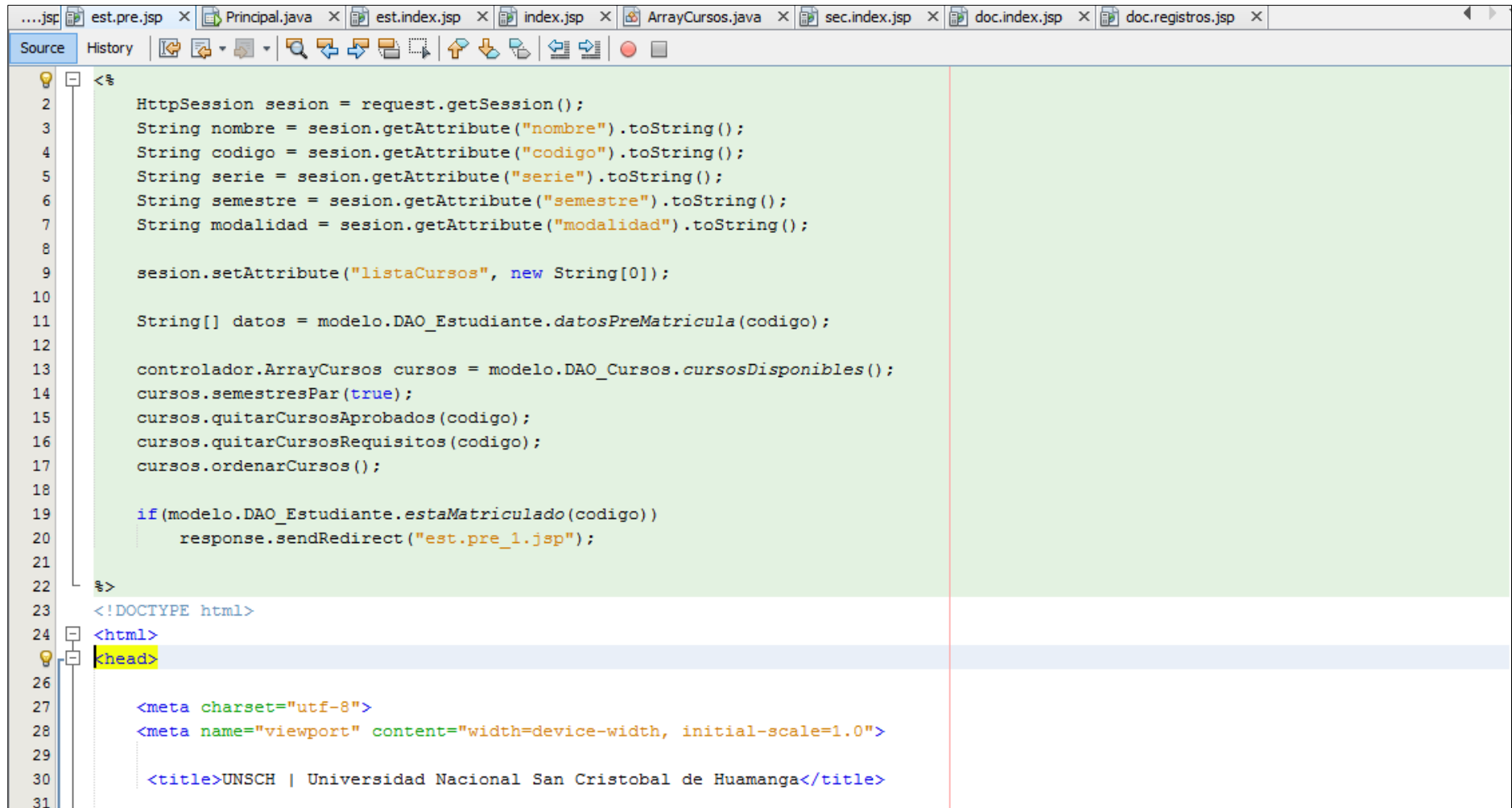
C.1 PÁGINA DE LOGEO



```
...ava index.jsp x ArrayCursos.java x DAO_Login.java x Curso.java x sec.index.jsp x doc.index.jsp x est.index.jsp x index.html x doc.registrarnotas.jsp...
Source History
1 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
2 <% boolean band = false;
3 if(modelo.DAO_Login.validarUsuarioLogin(request.getParameter( "user" ), request.getParameter( "pass")))
4 {
5     String[] aux = modelo.DAO_Login.instanciaUsuario(request.getParameter( "user" ));
6     HttpSession sesion = request.getSession();
7     sesion.setAttribute("tipo", aux[0]);
8     sesion.setAttribute("codigo", aux[1]);
9     String apellidoMinuscula = aux[3].substring(0, 1) + aux[3].substring(1, aux[3].length()).toLowerCase();
10    sesion.setAttribute("nombre", aux[2]+ " "+apellidoMinuscula);
11    sesion.setAttribute("semestre", "2016 - I");
12    sesion.setAttribute("tipoSemestre", "Regular");
13
14    switch(Integer.parseInt(aux[0])) {
15        case 1: response.sendRedirect("est.index.jsp"); break;
16        case 2: response.sendRedirect("doc.index.jsp"); break;
17        case 3: response.sendRedirect("sec.index.jsp"); break;
18    }
19
20 }
21 if(request.getParameter("user")!=null)
22     band = true;
23 %>
24 <!DOCTYPE html>
25 <html lang="es" dir="ltr">
26
27 <!-- Mirrored from webappplayers.com/inspinia_admin-v2.5/login.html by HTTrack Website Copier/3.x [XR&CO'2014], Mon, 25 Jul 2016 19:3:
28 <head>
29     <meta http-equiv="Content-Type" content="text/html charset=utf-8"/>
30     <meta charset="utf-8">
31     <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Figura N° 4.52: Código para la página de Logeo (Elaboración Propia)

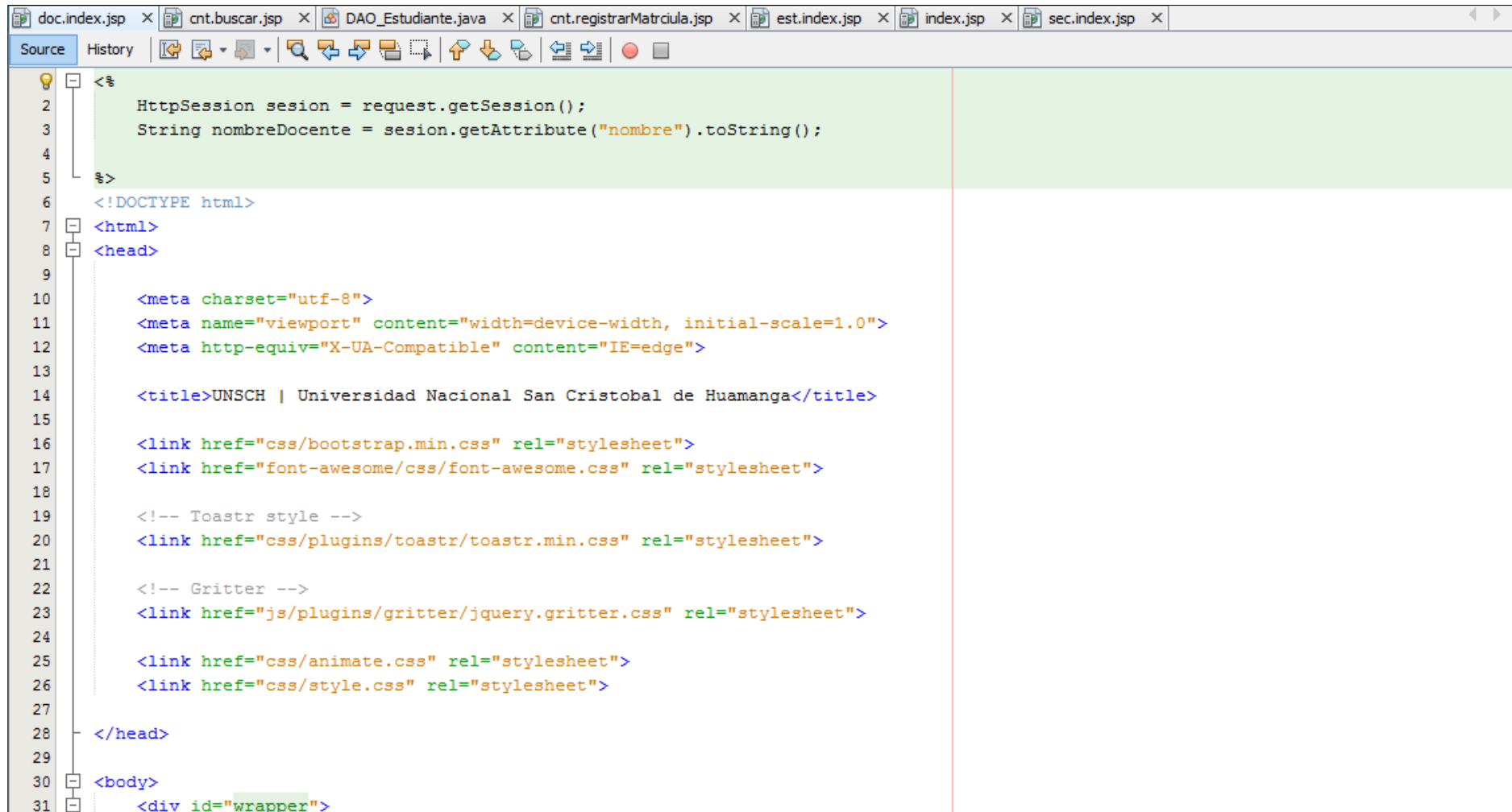
C.2 PÁGINA ESTUDIANTE



```
...jsp x est.pre.jsp x Principal.java x est.index.jsp x index.jsp x ArrayCursos.java x sec.index.jsp x doc.index.jsp x doc.registros.jsp x
Source History
2 HttpSession sesion = request.getSession();
3 String nombre = sesion.getAttribute("nombre").toString();
4 String codigo = sesion.getAttribute("codigo").toString();
5 String serie = sesion.getAttribute("serie").toString();
6 String semestre = sesion.getAttribute("semestre").toString();
7 String modalidad = sesion.getAttribute("modalidad").toString();
8
9 sesion.setAttribute("listaCursos", new String[0]);
10
11 String[] datos = modelo.DAO_Estudiante.datosPreMatricula(codigo);
12
13 controlador.ArrayCursos cursos = modelo.DAO_Cursos.cursosDisponibles();
14 cursos.semestresPar(true);
15 cursos.quitarCursosAprobados(codigo);
16 cursos.quitarCursosRequisitos(codigo);
17 cursos.ordenarCursos();
18
19 if(modelo.DAO_Estudiante.estaMatriculado(codigo))
20     response.sendRedirect("est.pre_1.jsp");
21
22 %>
23 <!DOCTYPE html>
24 <html>
25 <head>
26
27     <meta charset="utf-8">
28     <meta name="viewport" content="width=device-width, initial-scale=1.0">
29
30     <title>UNSCH | Universidad Nacional San Cristobal de Huamanga</title>
31
```

Figura N° 4.53: Código para la página Estudiante (Elaboración Propia)

C.3 PÁGINA DOCENTE

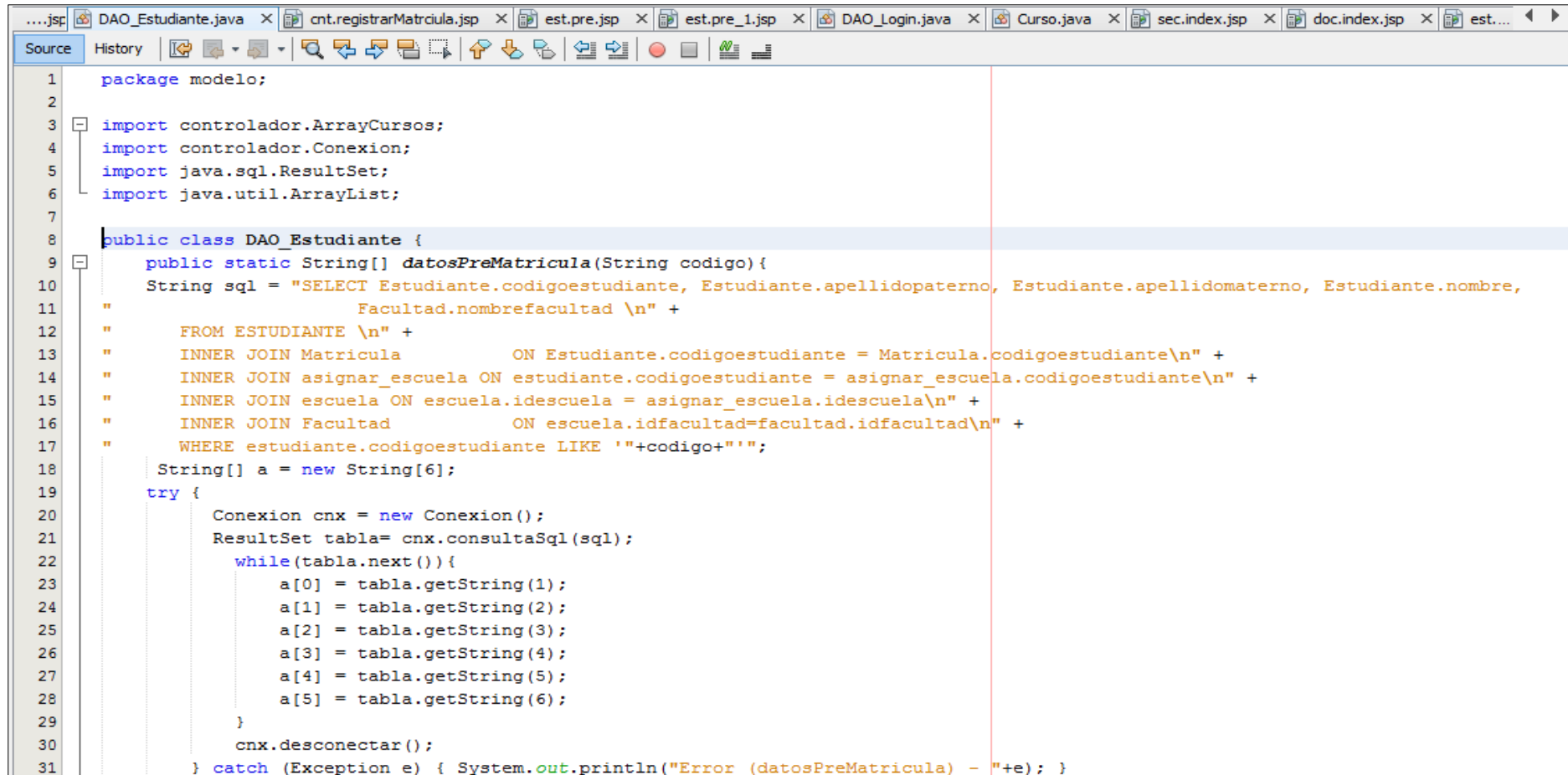


```
1 <%
2     HttpSession sesion = request.getSession();
3     String nombreDocente = sesion.getAttribute("nombre").toString();
4
5 >%
6 <!DOCTYPE html>
7 <html>
8 <head>
9
10     <meta charset="utf-8">
11     <meta name="viewport" content="width=device-width, initial-scale=1.0">
12     <meta http-equiv="X-UA-Compatible" content="IE=edge">
13
14     <title>UNSCH | Universidad Nacional San Cristobal de Huamanga</title>
15
16     <link href="css/bootstrap.min.css" rel="stylesheet">
17     <link href="font-awesome/css/font-awesome.css" rel="stylesheet">
18
19     <!-- Toastr style -->
20     <link href="css/plugins/toastr/toastr.min.css" rel="stylesheet">
21
22     <!-- Gritter -->
23     <link href="js/plugins/gritter/jquery.gritter.css" rel="stylesheet">
24
25     <link href="css/animate.css" rel="stylesheet">
26     <link href="css/style.css" rel="stylesheet">
27
28 </head>
29
30 <body>
31     <div id="wrapper">
```

Figura N° 4.54: Código para la página Docente (Elaboración Propia)

D. CODIGO FUENTE PARA CLASES DAO

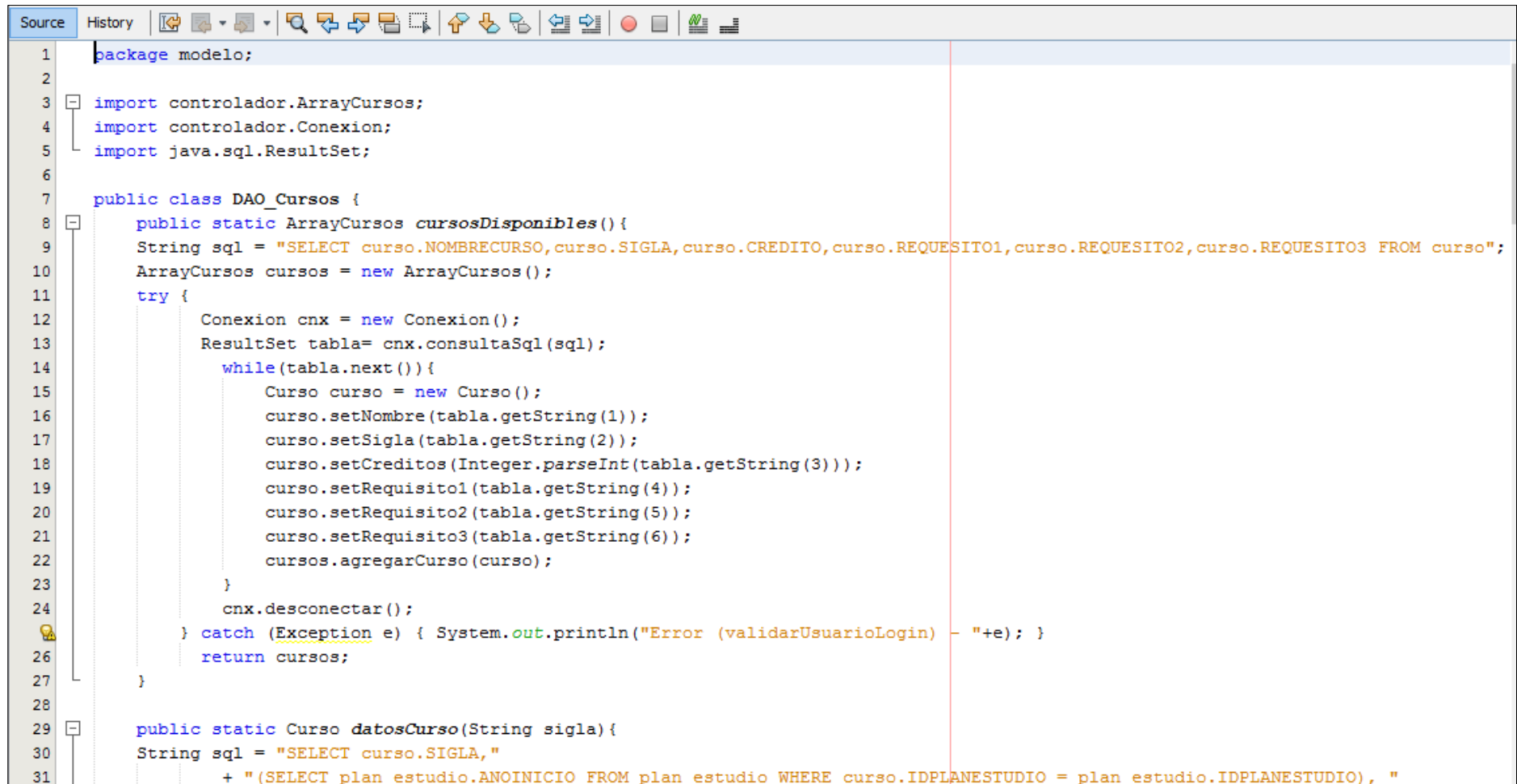
D.1 CLASE DAOESTUDIANTE



```
1 package modelo;
2
3 import controlador.ArrayCursos;
4 import controlador.Conexion;
5 import java.sql.ResultSet;
6 import java.util.ArrayList;
7
8 public class DAO_Estudiante {
9     public static String[] datosPreMatricula(String codigo){
10         String sql = "SELECT Estudiante.codigoestudiante, Estudiante.apellidopaterno, Estudiante.apellidomaterno, Estudiante.nombre,
11 "
12 "             Facultad.nombrefacultad \n" +
13 "             FROM ESTUDIANTE \n" +
14 "             INNER JOIN Matricula ON Estudiante.codigoestudiante = Matricula.codigoestudiante\n" +
15 "             INNER JOIN asignar_escuela ON estudiante.codigoestudiante = asignar_escuela.codigoestudiante\n" +
16 "             INNER JOIN escuela ON escuela.idescuela = asignar_escuela.idescuela\n" +
17 "             INNER JOIN Facultad ON escuela.idfacultad=facultad.idfacultad\n" +
18 "             WHERE estudiante.codigoestudiante LIKE '"+codigo+"'";
19         String[] a = new String[6];
20         try {
21             Conexion cnx = new Conexion();
22             ResultSet tabla= cnx.consultaSql(sql);
23             while(tabla.next()){
24                 a[0] = tabla.getString(1);
25                 a[1] = tabla.getString(2);
26                 a[2] = tabla.getString(3);
27                 a[3] = tabla.getString(4);
28                 a[4] = tabla.getString(5);
29                 a[5] = tabla.getString(6);
30             }
31             cnx.desconectar();
32         } catch (Exception e) { System.out.println("Error (datosPreMatricula) - "+e); }
```

Figura N° 4.55: Código fuente de la Clase DAO_Estudiante (Elaboración Propia)

D.2 CLASE DAOCURSO



```
1 package modelo;
2
3 import controlador.ArrayCursos;
4 import controlador.Conexion;
5 import java.sql.ResultSet;
6
7 public class DAO_Cursos {
8     public static ArrayCursos cursosDisponibles(){
9         String sql = "SELECT curso.NOMBRECURSO, curso.SIGLA, curso.CREDITO, curso.REQUERITO1, curso.REQUERITO2, curso.REQUERITO3 FROM curso";
10        ArrayCursos cursos = new ArrayCursos();
11        try {
12            Conexion cnx = new Conexion();
13            ResultSet tabla= cnx.consultaSql(sql);
14            while(tabla.next()){
15                Curso curso = new Curso();
16                curso.setNombre(tabla.getString(1));
17                curso.setSigla(tabla.getString(2));
18                curso.setCreditos(Integer.parseInt(tabla.getString(3)));
19                curso.setRequisito1(tabla.getString(4));
20                curso.setRequisito2(tabla.getString(5));
21                curso.setRequisito3(tabla.getString(6));
22                cursos.agregarCurso(curso);
23            }
24            cnx.desconectar();
25        } catch (Exception e) { System.out.println("Error (validarUsuarioLogin) - "+e); }
26        return cursos;
27    }
28
29    public static Curso datosCurso(String sigla){
30        String sql = "SELECT curso.SIGLA,"
31        + "(SELECT plan estudio.ANOINICIO FROM plan estudio WHERE curso.IDPLANESTUDIO = plan estudio.IDPLANESTUDIO), "
```

Figura N° 4.56: Código fuente de la Clase DAO_Curso (Elaboración Propia)

E. REPORTES DE PRUEBAS UNITARIAS

N° CU	NOMBRE CLASE CONTROL	NOMBRE DE LOS METODOS	RESULTADO
23	SMostrarPaginaPreMatricula.java	obtenerFacultad()	Satisfactorio
		obtenerEscuela()	Satisfactorio
		obtenerEstudiante()	Satisfactorio
		getSemestreString(String semestre)	Satisfactorio
		datosPreMatricula(int codigo)	Satisfactorio
	SGuardarIncidencia.java	guardarIncidencia()	Satisfactorio
	SMostarPaginaReportarError.java	mostrarPaginaReportarError()	Satisfactorio
	SGuardarDatosMatricula.java	guardarDatosMatricula()	Satisfactorio
	SMostrarDatosParaMatricula.java	getSemestreString(String semestre)	Satisfactorio
		datosCurso(String sigla)	Satisfactorio
		cursosNotas(String codigo, String semestre)	Satisfactorio
		cursosDisponibles()	Satisfactorio
	SRegistrarCursoMatriculado.java	registrarCursosMatricula(String idMatricula, int idCurso)	Satisfactorio

Tabla N° 4.20: Reporte de Prueba Unitaria del caso de uso “Registrar Matrícula” (Elaboración Propia)

F. REPORTE DE PRUEBAS DE ACEPTACIÓN

N° Req.	N° C.P	CASOS DE PRUEBA DE ACEPTACIÓN	RESULTADO
12 13 14	01	Comprobar que los datos de los estudiantes son correctos.	Satisfactorio
	02	Comprobar que los cursos matriculados, los créditos, la serie y el semestre académico en una Ficha de Matrícula de un estudiante son correctos.	Satisfactorio
	03	Verificar que el monto a pagar en una Esquela de Pago de un estudiante es correcto.	Satisfactorio
	04	Imprimir la Esquela de Pago y la Ficha de Matrícula de los estudiantes para un semestre académico.	Satisfactorio
18 19 20	05	Comprobar que los datos del personal administrativo son correctos.	Satisfactorio
	06	Comprobar que los cursos que dictan los docentes son correctos.	Satisfactorio
	07	Comprobar que el Registro de Evaluación, presenta información correcta.	Satisfactorio
	08	Verificar que los datos registrados en el Registro de Evaluación son correctos.	Satisfactorio
	09	Imprimir el Registro de Evaluación para un semestre académico.	Satisfactorio
31	10	Comprobar que la Acta de Evaluación Final, presenta información correcta.	Satisfactorio
32	11	Imprimir el Acta de Evaluación Final para un semestre académico.	Satisfactorio

Tabla N° 4.21: Reporte de Pruebas de Aceptación (Elaboración Propia)

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- a.** Se implementó la aplicación web para optimizar el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, gracias a la metodología ICONIX y tecnologías de Internet; logrando así un control integral del proceso de matrícula y obteniendo un software que automatice dicho proceso, de manera confiable, rápido y seguro.

- b.** Se logró identificar y modelar los artefactos para el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, como se muestra en los diagramas de robustez de las siguientes figuras: Figura N° 4.30: Diagrama de Robustez, caso de uso “Registrar Matrícula”, Figura N° 4.31: Diagrama de Robustez, caso de uso “Emitir Esquela de Pago de matrícula regular”, Figura N° 4.32: Diagrama de Robustez, caso de uso “Emitir Ficha de Matrícula”, Figura N° 4.33: Diagrama de Robustez, caso de uso “Emitir Registro de Evaluación”, Figura N° 4.34: Diagrama de Robustez, caso de uso “Registrar nota del curso del semestre regular” y Figura N° 4.35: Diagrama de Robustez, caso de uso “Emitir Acta de Evaluación Final del semestre regular”; y los diagramas de secuencia de las siguientes figuras: Figura N° 4.39: Diagrama de Secuencia, caso de uso “Registrar Matrícula”, Figura N° 4.40: Diagrama de Secuencia, caso de uso “Emitir Esquela de Pago de matrícula regular”, Figura N° 4.41: Diagrama de Secuencia, caso de uso “Emitir Ficha de Matrícula”, Figura N° 4.42: Diagrama de Secuencia, caso de uso “Emitir Registro de Evaluación”, Figura N° 4.43: Diagrama de Secuencia, caso de uso “Registrar nota del curso del semestre regular” y Figura N° 4.44: Diagrama de Secuencia, caso de uso “Emitir Acta de Evaluación Final del semestre regular”.

- c. Se logró diseñar prototipos amigables para el proceso de matrícula de los estudiantes de la Universidad Nacional de San Cristóbal de Huamanga, como se muestra en la Figura N° 4.23: Prototipo de interfaz gráfica revisada “Registrar Matrícula” [Paso 1], Figura N° 4.24: Prototipo de interfaz gráfica revisada “Registrar Matrícula” [Paso 2], Figura N° 4.25: Prototipo de interfaz gráfica revisada “Emitir Esquela de Pago de matrícula regular”, Figura N° 4.26: Prototipo de interfaz gráfica revisada “Emitir Ficha de Matrícula”, Figura N° 4.27: Prototipo de interfaz gráfica revisada “Emitir Registro de Evaluación”, Figura N° 4.28: Prototipo de interfaz gráfica revisada “Registrar nota del curso del semestre regular” y Figura N° 4.29: Prototipo de interfaz gráfica revisada “Emitir Acta de Evaluación Final del semestre regular”.

- d. Se logró la migración de archivos relacionados de FOXPRO a una base de datos Oracle del proceso de matrícula utilizando el algoritmo que se encuentra en el anexo C.

5.2. RECOMENDACIONES

- a. Se recomienda estandarizar el sistema de evaluación para mejorar la gestión de registro de notas, y así lograr la eficiencia de servicios a los docentes de la Universidad Nacional de San Cristóbal de Huamanga.

- b. Se recomienda implementar la firma digital para la gestión de Ficha de Matrícula, Registro de Evaluaciones y Acta de Evaluación Final, para mejorar el servicio que se brinda a los estudiantes, docentes y personal administrativo de la Universidad Nacional de San Cristóbal de Huamanga.

BIBLIOGRAFÍA

1. Aumaille, B. (2002). J2EE Desarrollo de aplicaciones web. Barcelona, España: Eni.
2. Batini, C., Ceri, S. y Navathe, S. B. (2000). Diseño conceptual de base de datos. Un enfoque de entidades-interrelaciones. Boston, Estados Unidos: Addison-Wesley Iboamericana.
3. Bernal, C. (2006). Metodología de la Investigación para Administración y Economía. Santa Fé de Bogotá. D.C, Colombia: Editorial Nomos S.A.
4. Boletín Estadístico 2011-2012 [archivo de datos]. Ayacucho, Perú: Universidad Nacional de San Cristóbal de Huamanga.
5. Caballero, A. (2008). Innovaciones en las guías metodológicas para planes y tesis de maestría y doctorado. Lima, Perú: Instituto Metodológico ALEN CARO.
6. Ceballos, F. (2000). Programación orientada a objetos con C++. Washington, Estados Unidos: Addison - Wesley Iberoamericana.
7. Díaz, A., Duran, W. y Quijano, R. (2010). Sistema de Información para Gestión Académica. Tecnológica Fitec. Bucaramanga, Colombia.
8. Estatuto Reformulado de la UNSCH [archivo de datos]. Ayacucho, Perú: Universidad Nacional de San Cristóbal de Huamanga.
9. García, V. y Medina, R. (1987). Organización y Gobierno de centros educativos. Madrid, España: Ediciones Rialp.
10. Gómez, M., (2006). Introducción a la Metodología de la investigación científica. Córdoba, Argentina: Editorial Brujas.
11. Hansler, G. y Hansen, J. (2000). Diseño y administración de base de datos (2ª Ed.).Madrid, España: Prentice Hall.
12. Hernández et al. (2010). Metodología de la Investigación. (5ta Ed.). México, D.F., México: McGraw.
13. Jacobson, I., Booch, G. y Rumbaugh, J. (1999). The Unified Software Development Process. Madrid, España: Addison-Wesley Professional.

14. Jarabo, F. y Elortegui, N. (2000). Internet. Conexión desde una PC doméstico a ordenadores de todo el mundo (2da Ed.). Madrid, España: Paraninfo.
15. Joyanes, L. (2008). Fundamentos de programación: Algoritmos, estructura de datos y objetos (4ta Ed.). Madrid, España: McGraw – Hill/Interamericana de España.
16. Kendall, J. (2005). Análisis y diseño de sistemas (6ta Ed.). Atlacomulco, México: Pearson Educación.
17. Lázaro, J. (2006). Sistematización de un proceso de matrícula. Universidad Nacional de Ancash “Santiago Antúnez de Mayolo”. Huaraz, Perú.
18. Lozano, H. (2011). Como desarrollar el proyecto de investigación en carreras de Ingeniería. Perú.
19. Luján, S. (2002). Programación de aplicaciones web: historia, principios básicos y clientes web. Madrid, España: Club Universitario.
20. Mora, N. y Vega, C. (2013). Sistema Informático de control de notas y proceso de matriculación de la Unidad Educativa Wenceslao Ríjavec de la ciudad de Calceta- Cantón Bolívar. Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López. Calceta, Ecuador.
21. Naughton, P. (2000). Manual de Java. España: McGraw – Hill/Interamericana de España.
22. Nevado, V. s.f. Introducción a las Bases de Datos Relacionales. Madrid, España, Visión Libros.
23. Noriega, A. (2007). Programación en Java2. Lima, Perú: Grupo Editorial Megabyte.
24. Pressman, R. (2010). Ingeniería del Software: Un enfoque práctico. (7ma Ed.). México, D. F., México: McGraw – Hill Interamericana.
25. Osorio, L. (2008). Base de datos relacionales, Teoría y práctica. Colombia: Fondo editorial ITM.
26. Reglamento Interno de la UNSCH [archivo de datos]. Ayacucho, Perú: Universidad Nacional de San Cristóbal de Huamanga.
27. Rodríguez, J. (2003). Introducción a la programación: Teoría y práctica. Alicante, España: Editorial Club Universitario.
28. Rosenberg, D y Stephens, M. (2007). Use Case Driven Object Modeling with UML: Theory and Practice. Primera Edición. United States, Apress.

29. Rosenberg et al. (2005). *Agile Development with ICONIX Process: People, Process and Pragmatism*. Primera Edición. United States, Apress.
30. Seoane, E. (2005). *La nueva era del comercio, el comercio electrónico: las TIC al servicio de la Gestión Empresarial*. Madrid, España: Ideaspropias.
31. Soares, F. (2008). ¿Automatizar todo?. Recuperado el 15 de Julio de 2015, de http://www.kentron.com.ve/novedades/automatizar_todo.htm
32. Silberschatz, A., Korth, H. y Sudarshan, S. (2001). *Fundamentos de base de datos* (3ª Ed.). Madrid, España: MsGraw Hill/ Interamericana de España.
33. Stallings, W. (2009). *Comunicaciones y Redes de Computadores* (6ta Ed.). España: PEARSON Prentice Hall.
34. Vásquez, J. y Balta, V. (2008). *SUPER JAVA Estandar Edition for Windows with JCreator IDE* (3ª Ed.). Lima, Perú: La Esperanza. Diseños e Impresión SAC.
35. Villón, G. (2013). *Creación e Implementación de un software para la automatización del registro de matrícula de los estudiantes en el Centro de Educación Básica "Ignacio Alvarado" de la comuna Palmar, provincia de Santa Elena, período lectivo 2012-2013*. Universidad Estatal "Península de Santa Elena". La Libertad, Ecuador.
36. Wu, C. (2008). *Programación en Java. Introducción a la Programación Orientada a Objetos*. México, D.F., México: McGraw-Hill/Interamericana editores.
37. Zapata, M. (2008). *La clave de la automatización en los procesos administrativos*. Recuperado el 15 de Julio de 2015, de <http://www.evaluandosoftware.com/nota-3279-La-clave-de-la-automatizacion-en-los-procesos-administrativos.htm>.

ANEXOS

A. INSTRUMENTO PARA RECOLECTAR INFORMACIÓN DEL PROCESO DE MATRÍCULA MEDIANTE EL ANÁLISIS DOCUMENTAL

Número de Registro:
Autor(es):
Nombre del Documento:
Datos Fuente:
Tipo de Documento:
Resumen:
Palabras Claves:
Clasificación:

Tabla A.1: Ficha de Análisis Documental del Proceso de Matrícula (Elaboración Propia)

B. INSTRUMENTO PARA RECOLECTAR INFORMACIÓN DEL PROCESO DE MATRÍCULA MEDIANTE LA OBSERVACIÓN

TEMA:
Subtema:
Lugar:
Fuente:
Nombre del Investigador:
Descripción de la Observación:
Conclusión de la Observación:
Fecha de la Observación:

Tabla B.1: Ficha de Observación del Proceso de Matrícula (Elaboración Propia)

C. ALGORITMO DE MIGRACIÓN

Sub ejecutar()

Dim i As Integer

Dim j As Integer

Dim k As Integer

Dim aux As String

Dim aux2 As String

Dim idMatricula As Integer

Dim idCursoMatriculado As Integer

Dim idPeriodo As Integer

Dim bandera As Boolean

idMatricula = ? 'ID

idCursoMatriculado = ? 'ID

idPeriodo = ? 'PERIODO

Hoja3.Name = "estudiante"

Hoja4.Name = "Matricula"

Hoja5.Name = "curso_matriculado"

Hoja6.Name = "evaluacion"

Hoja3.Range("A" & 1).Value = "codigoestudiante"

Hoja3.Range("B" & 1).Value = "apellidopaterno"

Hoja3.Range("C" & 1).Value = "apellidomaterno"

Hoja3.Range("D" & 1).Value = "nombre"

Hoja4.Range("A" & 1).Value = "IdMatricula"

Hoja4.Range("B" & 1).Value = "idPersonalAdministrativo"

Hoja4.Range("C" & 1).Value = "CODIGO"

Hoja4.Range("D" & 1).Value = "idPeriodoAcademico"

Hoja4.Range("E" & 1).Value = "idSerie"

Hoja4.Range("F" & 1).Value = "tipoMatricula"

```

Hoja5.Range("A" & 1).Value = "idcursomatriculado"

Hoja5.Range("B" & 1).Value = "idmatricula"

Hoja5.Range("C" & 1).Value = "idcurso"

Hoja6.Range("A" & 1).Value = "idevaluacion"

Hoja6.Range("B" & 1).Value = "idcursomatriculado"

Hoja6.Range("C" & 1).Value = "nota"

Hoja6.Range("D" & 1).Value = "naturalezaexamen"

Hoja6.Range("E" & 1).Value = "fechaacta"

aux = "0"

aux2 = "0"

j = 2

For i = 2 To Hoja1.Range("A" & Rows.Count).End(xlUp).Row

    bandera = False

    If aux <> Hoja1.Range("A" & i).Value Then 'nuevo codigo

        bandera = True

    Else

        If aux2 <> Hoja1.Range("D" & i).Value Then 'nuevo tipo

            bandera = True

        End If

    End If

    If bandera Then

        Worksheets("Matricula").Range("A" & j).Value = idMatricula + j - 1

        Worksheets("Matricula").Range("B" & j).Value = 1 'siempre

        Worksheets("Matricula").Range("C" & j).Value = Hoja1.Range("A" & i).Value

        Worksheets("Matricula").Range("D" & j).Value = idPeriodo

        Worksheets("Matricula").Range("E" & j).Value = Left(Hoja1.Range("J" & i).Value, 1)

        Worksheets("Matricula").Range("F" & j).Value = Hoja1.Range("D" & i).Value

```

```

    j = j + 1

    aux = Hoja1.Range("A" & i).Value

    aux2 = Hoja1.Range("D" & i).Value

End If

Next i

j = 2

aux = "0"

For i = 2 To Hoja1.Range("A" & Rows.Count).End(xlUp).Row

    If aux <> Hoja1.Range("A" & i).Value Then

        Worksheets("estudiante").Range("A" & j).Value = Hoja1.Range("A" & i).Value

        Worksheets("estudiante").Range("B" & j).Value = Left(Hoja1.Range("B" & i).Value,
InStr(Hoja1.Range("B" & i).Value, " "))

        Worksheets("estudiante").Range("C" & j).Value = Right(Left(Hoja1.Range("B" & i).Value,
InStr(InStr(Hoja1.Range("B" & i).Value, " "), Hoja1.Range("B" & i).Value, ",") - 1),
Len(Left(Hoja1.Range("B" & i).Value, InStr(InStr(Hoja1.Range("B" & i).Value, " "), Hoja1.Range("B"
& i).Value, ",") - 1)) - InStr(Hoja1.Range("B" & i).Value, " "))

        Worksheets("estudiante").Range("D" & j).Value = Right(Hoja1.Range("B" & i).Value,
Len(Hoja1.Range("B" & i).Value) - InStr(Hoja1.Range("B" & i).Value, ","))

        aux = Hoja1.Range("A" & i).Value

        j = j + 1

    End If

Next i

j = 1

aux = "0"

aux2 = "0"

For i = 2 To Hoja1.Range("A" & Rows.Count).End(xlUp).Row

    If aux <> Hoja1.Range("A" & i).Value Then 'nuevo codigo

        j = j + 1

        aux = Hoja1.Range("A" & i).Value

        aux2 = Hoja1.Range("D" & i).Value

```

```

Else
    If aux2 <> Hoja1.Range("D" & i).Value Then 'nuevo tipo
        j = j + 1
        aux = Hoja1.Range("A" & i).Value
        aux2 = Hoja1.Range("D" & i).Value
    End If
End If

Worksheets("curso_matriculado").Range("A" & i).Value = idCursoMatriculado + i - 1 'cambiar
por el id

Worksheets("curso_matriculado").Range("B" & i).Value = idMatricula + j - 1
Worksheets("curso_matriculado").Range("C" & i).Value = Hoja1.Range("C" & i).Value

For k = 2 To 74
    If Trim(Worksheets("curso_matriculado").Range("C" & i).Value) = Hoja2.Range("C" &
k).Value Then
        Worksheets("curso_matriculado").Range("C" & i).Value = Hoja2.Range("A" & k).Value
    End If
Next k

Worksheets("evaluacion").Range("A" & i).Value = idCursoMatriculado + i - 1 'cambiar por el id
Worksheets("evaluacion").Range("B" & i).Value = idCursoMatriculado + i - 1 'cambiar por el id
Worksheets("evaluacion").Range("C" & i).Value = Hoja1.Range("F" & i).Value
Worksheets("evaluacion").Range("D" & i).Value = Hoja1.Range("D" & i).Value
Worksheets("evaluacion").Range("E" & i).Value = Hoja1.Range("G" & i).Value

Next i

End Sub
-----
Sub hojas()
    Dim i As Integer

```

```
For i = 1 To 4
    Worksheets.Add
Next i
Hoja2.Name = "aux_curso"

Hoja2.Range("A" & 1).Value = "IDCURSO"
Hoja2.Range("B" & 1).Value = "IDPLANESTUDIO"
Hoja2.Range("C" & 1).Value = "SIGLA"

For i = 2 To 74
    Hoja2.Range("A" & i).Value = i - 1
    Hoja2.Range("B" & i).Value = 1
Next i
Hoja2.Range("C" & 2).Value = "IS141"
Hoja2.Range("C" & 3).Value = "MA141"
Hoja2.Range("C" & 4).Value = "MA143"
Hoja2.Range("C" & 5).Value = "LE141"
Hoja2.Range("C" & 6).Value = "MD141"
Hoja2.Range("C" & 7).Value = "CC121"
Hoja2.Range("C" & 8).Value = "EC141"
Hoja2.Range("C" & 9).Value = "MA241"
Hoja2.Range("C" & 10).Value = "IS545"
Hoja2.Range("C" & 11).Value = "IS543"
Hoja2.Range("C" & 12).Value = "AD541"
Hoja2.Range("C" & 13).Value = "CS341"
Hoja2.Range("C" & 14).Value = "FS241"
Hoja2.Range("C" & 15).Value = "CO341"
Hoja2.Range("C" & 16).Value = "AD551"
Hoja2.Range("C" & 17).Value = "IS241"
Hoja2.Range("C" & 18).Value = "EC241"
Hoja2.Range("C" & 19).Value = "IS347"
```

Hoja2.Range("C" & 20).Value = "EC441"
Hoja2.Range("C" & 21).Value = "AD451"
Hoja2.Range("C" & 22).Value = "MA243"
Hoja2.Range("C" & 23).Value = "ES241"
Hoja2.Range("C" & 24).Value = "IS343"
Hoja2.Range("C" & 25).Value = "IS345"
Hoja2.Range("C" & 26).Value = "IS541"
Hoja2.Range("C" & 27).Value = "IS443"
Hoja2.Range("C" & 28).Value = "IS441"
Hoja2.Range("C" & 29).Value = "IS453"
Hoja2.Range("C" & 30).Value = "AC121"
Hoja2.Range("C" & 31).Value = "CS321"
Hoja2.Range("C" & 32).Value = "AD441"
Hoja2.Range("C" & 33).Value = "IS553"
Hoja2.Range("C" & 34).Value = "IS341"
Hoja2.Range("C" & 35).Value = "IS547"
Hoja2.Range("C" & 36).Value = "IS551"
Hoja2.Range("C" & 37).Value = "IS445"
Hoja2.Range("C" & 38).Value = "IS451"
Hoja2.Range("C" & 39).Value = "MA144"
Hoja2.Range("C" & 40).Value = "AD142"
Hoja2.Range("C" & 41).Value = "IS362"
Hoja2.Range("C" & 42).Value = "IS344"
Hoja2.Range("C" & 43).Value = "MA142"
Hoja2.Range("C" & 44).Value = "IS242"
Hoja2.Range("C" & 45).Value = "IS348"
Hoja2.Range("C" & 46).Value = "IS542"
Hoja2.Range("C" & 47).Value = "ES244"
Hoja2.Range("C" & 48).Value = "AD452"
Hoja2.Range("C" & 49).Value = "AD444"

```
Hoja2.Range("C" & 50).Value = "EC142"  
Hoja2.Range("C" & 51).Value = "IS248"  
Hoja2.Range("C" & 52).Value = "IS442"  
Hoja2.Range("C" & 53).Value = "IS444"  
Hoja2.Range("C" & 54).Value = "IS142"  
Hoja2.Range("C" & 55).Value = "AC222"  
Hoja2.Range("C" & 56).Value = "IS548"  
Hoja2.Range("C" & 57).Value = "FS142"  
Hoja2.Range("C" & 58).Value = "IS546"  
Hoja2.Range("C" & 59).Value = "IS544"  
Hoja2.Range("C" & 60).Value = "IS552"  
Hoja2.Range("C" & 61).Value = "IS452"  
Hoja2.Range("C" & 62).Value = "IS246"  
Hoja2.Range("C" & 63).Value = "CO342"  
Hoja2.Range("C" & 64).Value = "DE542"  
Hoja2.Range("C" & 65).Value = "EF122"  
Hoja2.Range("C" & 66).Value = "IS244"  
Hoja2.Range("C" & 67).Value = "IS346"  
Hoja2.Range("C" & 68).Value = "AD442"  
Hoja2.Range("C" & 69).Value = "IS342"  
Hoja2.Range("C" & 70).Value = "AD552"  
Hoja2.Range("C" & 71).Value = "IS454"  
Hoja2.Range("C" & 72).Value = "IS554"  
Hoja2.Range("C" & 73).Value = "IS262"  
Hoja2.Range("C" & 74).Value = "IS446"  
  
End Sub
```

Tabla C.1: Algoritmo de Migración (Elaboración Propia)

