



Ambiente de AutoEvaluación (AAEV 2.0)

**Canosa No Alejandro Agustín
Galvan Gabriel Nazareno
Di Stefano Lucas Agustín**

**Cátedra Proyecto de Software
Licenciatura en Sistemas**

2016

Ambiente de AutoEvaluación (AAEV 2.0)

Canosa No Alejandro A., Galván Gabriel N, Di Stefano Agustín L
Licenciatura en Sistemas- Departamento de Desarrollo Productivo y Tecnológico
Universidad Nacional de Lanús.

Extracto—En el presente documento se trabajara el proceso de análisis, proyección y desarrollo del proyecto de software orientado a la creación de un Artefacto Web que soporte un Ambiente de AutoEvaluación (AAEV 2.0)

Palabras Clave—AAEV, Proyecto, Gestión, Riesgos, Instalación, Tareas,

I. PROCESO DE SELECCIÓN DE UN CICLO DE VIDA DEL PRODUCTO

El modelo de ciclo de vida elegido para el proyecto fue el Modelo Prototipado, presentando una maqueta primero, presentando la interfaz de usuario y luego un prototipado evolutivo el cual se redefinirá sus requisitos y se modificara hasta el final del mismo. El prototipo se construye en poco tiempo, con pocos recursos y genera nuevos requisitos o los redefine. Este modelo es usado generalmente cuando el cliente tiene los requisitos generales del sistema, pero no conoce los requisitos específicos y sus entradas, procesamientos o salidas. Estos prototipos serán cambiados hasta llegar al producto software final, una vez que el cliente defina los requisitos finales. La maqueta es la presentación de datos estáticos donde se mostrará la interfaz de usuario del sistema y el flujo de información del mismo al cliente para lograr comunicar la solución encontrada por el equipo del proyecto.

El ciclo de vida en prototipado de tipo evolutivo tiene su propio mapa de actividades (ver Figura 4) donde se definen las actividades a realizar en las etapas del diseño del producto software y en qué etapas se realizan. Es normal que haya más de una iteración en una fase de desarrollo o diseño, ya que se reajustan o definen nuevos requisitos del proyecto.

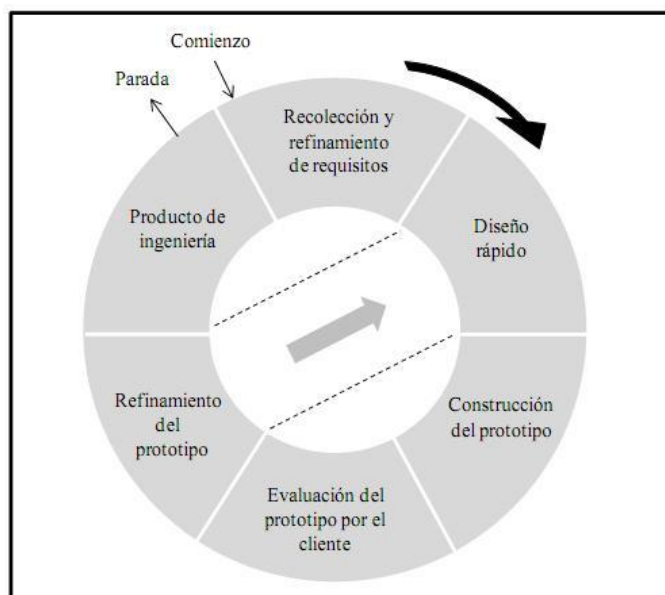


Figura 1. Ciclo de vida en prototipado

II. PROCESOS DE GESTIÓN DEL PROYECTO

A. Proceso de Iniciación del Proyecto

a) Plan de Gestión

Esta sección es el plan de Gestión del proyecto Ambiente de AutoEvaluación 2.0 (AAEV 2.0), pedido por el cliente, con el fin de crear un espacio web donde docentes crearán exámenes para que los alumnos prueben sus conocimientos sobre las materias de sus respectivas carreras.

El proyecto sigue el ciclo de vida en prototipado evolutivo, teniendo el diseño de una maqueta primero de 1 a 3 iteraciones y luego el prototipo donde el usuario refinará los requisitos del sistema.

1) Alcance del proyecto

El objetivo del proyecto es crear un ambiente web de ámbito universitario donde el docente de una o varias materias de una carrera elaborará exámenes a partir del contenido de las mismas para ser luego realizados por los alumnos que se inscriban en ellas mediante el ambiente para así poder lograr una observación de su progreso en la adquisición y entendimiento de conceptos de la materia.

El producto entregado, dependiendo del privilegio asignado, permitirá a los usuarios:

- Gestionar las carreras de la universidad: agregar, modificar, o eliminar una carrera. Acción realizada por el administrador del ambiente.
- Gestionar las materias de cada carrera: agregar, modificar o eliminar una materia. Esta acción es realizada por el administrador del ambiente.
- Gestionar las cuentas del cuerpo docente de cada carrera: agregar, modificar o eliminar un docente. Esta acción es realizada por el administrador del ambiente
- Gestionar los exámenes: agregar exámenes, modificarlos o eliminarlos. Dentro del proceso de gestión de exámenes encontramos la gestión de opciones, donde dependiendo de la pregunta elegida el docente realizará el alta, baja o modificación de las mismas. Por último, el docente asigna un sistema de puntuación al examen agregado.
- Aceptar solicitudes de alumnos para unirse a la materia: los docentes eligen de la lista de solicitudes que alumnos podrán acceder a determinada materia y cuáles no.
- Registrarse en el sistema: el alumno ingresará sus datos en un formulario y se registrará en el sistema con su usuario y clave

- Inscribirse a materia: el alumno elige una de las materias de la carrera y envía una solicitud de inscripción al/los docente/s de la misma, es cuestión del docente aceptarlo en ella o no
- Realizar examen: el alumno, una vez aceptado en la materia, ingresa al examen y contesta las preguntas definidas previamente por el docente, para al final obtener una nota en base a las respuestas correctas e incorrectas del alumno

Para más detalles de las acciones de los usuarios, ver el documento de requisitos del proyecto.

TABLA III. ENTREGABLES DEL PROYECTO.

Identificación del entregable	Detalles del entregable	
	Descripción	Fecha de entrega
Modelo de ciclo de vida seleccionado.	Modelo de ciclo de vida en el que se basan las actividades del proyecto	25/8/16
Plan de Gestión de Proyecto	Es el presente documento	25/8/16
Mapa de actividades	Mapa donde se indica las actividades de los procesos y en qué etapa de desarrollo se realizan, guía al equipo de desarrollo	25/8/16
Análisis de Riesgos	Análisis de los riesgos potenciales del proyecto, su probabilidad de ocurrencia e impacto en el mismo	25/8/16
Plan de Contingencias	Documento donde se planifican las medidas a tomar en caso de que uno de los riesgos analizados se materialice en el desarrollo del proyecto	25/8/16
Plan de Garantía de Calidad del Software	Documento donde se indican los factores que aseguran la calidad del software para satisfacer necesidades dadas	22/9/16
Plan de transición del proyecto	Documento donde se planifica la transición del proyecto del ambiente de desarrollo al ambiente de producción	25/8/16
Modelo de la situación actual	Modelo donde se indica la situación actual del entorno del proyecto (si funciona con otros sistemas, y el funciona-miento de estos)	25/8/16
Modelo del dominio del problema	Modelo conceptual de todos los temas relacionados con un problema específico. En él se describen las distintas entidades, sus atributos, papeles y relaciones, además de las restricciones que rigen el dominio del problema.	25/8/16
Informe preliminar de necesidades	Informe de las necesidades del cliente para el proyecto	25/8/16
Soluciones alternativas posibles	Documento donde se tienen en cuenta las alternativas posibles del desarrollo del software	25/8/16
Soluciones recomendadas	Documento donde se plasman las soluciones recomendadas de acuerdo a las necesidades del cliente.	25/8/16
Documento de requisitos del software	Documento donde se detallan los requisitos del cliente, en el se encuentran integrados los requisitos del software ya mencionados, los requisitos de interfaz con usuario, con otro software, con el hardware y con el	22/9/16

Los privilegios del sistema serán:

- Alto: Administrador
- Medio: Docente
- Bajo: Alumno

2) Entregables del proyecto

En esta sección se definen los documentos incluidos en el presente informe del proyecto, estos documentos establecen la planificación, los riesgos, las tareas a realizar, entre otras funciones.

	sistema físico (si se especifican)	
-Descripción del diseño de software	Auto explicativo	22/9/16
Descripción del flujo de información	Auto explicativo	22/9/16
Descripción de la base de datos	Auto explicativo	22/9/16
Descripción de las interfaces	Auto explicativo	No especificada.
Datos para las pruebas	Datos necesarios para ejecutar las pruebas de software y creación del Plan de Pruebas con sus especificaciones de prueba	No especificada.
Documentación del sistema	Auto explicativo	No especificada.
Documentación de usuario	Manual de usuario	No especificada.
Plan de integración	Plan de integración de los módulos del sistema	No especificada.
Sistema software integrado	“Materialización” del software diseñado a través de lenguajes de programación	No especificada.
Plan de instalación de software	Documento que brinda pasos al usuario de cómo instalar el producto software (también llamado manual de instalación)	No especificada.
Orden de mantenimiento	Detalles de la orden de mantenimiento del producto software	No especificada.
Recomendaciones de mantenimiento	Auto explicativo	No especificada.
Casos de Uso	Documento que muestra las funciones del punto de vista del negocio y los usuarios que las realizan	22/9/16
Especificación de Casos de Uso	Brinda información detallada de cada caso de uso así como los pasos contenidos y la información necesaria para cada uno	22/9/16
Diagrama Entidad Relación	Diagrama que establece la relación entre los datos del sistema y como se distribuyen los mismos	22/9/16
Diagrama de clases	Diagrama que muestra el conjunto de objetos del sistema, las operaciones que puede realizar cada una de ellas y las dependencias y asociaciones entre los mismos	22/9/16.
Plan de verificación y validación	Plan donde se establecen los métodos para verificar y validar el software	22/9/16
Plan de pruebas	Documento donde se planifican las pruebas de software	Sin especificar

TABLA III. ENTREGABLES DEL PROYECTO (II).

Identificación del entregable	Descripción	
	Descripción	Fecha de entrega
Especificación de las pruebas	Documento donde se registran los casos de prueba de software	Sin especificar
Informe resumen de las pruebas	Documento donde se especifican los resultados de las pruebas de software	Sin especificar
Software probado	Software luego de realizar los casos de prueba	Sin especificar

3) Estrategia de evolución del Plan

Por decisión del equipo, Alejandro Canosa es responsable del monitoreo del presente documento, con una frecuencia media prevista de cambios al mismo, en caso de que se presente un posible cambio, el equipo se reunirá y debatirá si el cambio del plan es necesario o no, en caso de que lo sea, los cambios en el documento serán efectuados, se enviará una copia a cada integrante del equipo y se registrará el cambio en la tabla de versiones de documentos, junto con la versión, la fecha, el autor y la descripción del mismo.

Al finalizar el cambio y su registro, el autor del mismo deberá informar por cualquier medio de comunicación la finalización del proceso y esta versión nueva se les deberá ser enviada y subida al repositorio donde se guardan las versiones actuales.

El plan será revisado al inicio de cada fase, modificado de acuerdo a lo necesario, aprobado y distribuido al equipo de proyecto.

4) Organización del proyecto

Por decisión del equipo, Alejandro Canosa es responsable del monitoreo del presente documento, con una frecuencia media prevista de cambios al mismo, en caso de que se presente un posible cambio, el equipo se reunirá y debatirá si el cambio del plan es necesario o no, en caso de que lo sea, los cambios en el documento serán efectuados, se enviará una copia a cada integrante del equipo y se registrará el cambio en la tabla de versiones de documentos, junto con la versión, la fecha, el autor y la descripción del mismo.

5) Tareas del proyecto

Esta sección contiene la definición de las relaciones entre las actividades más relevantes del proyecto (Figura 2). Ver el Diagrama de GANTT del proyecto (Figura 3).











































Id	Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		Ambiente de Autoevaluación	90 días	lun 8/8/16	vie 9/12/16	
2		Análisis preliminar	15 días	lun 8/8/16	vie 26/8/16	
3		Proceso de Selección de un MCVS	1 día	lun 8/8/16	lun 8/8/16	
4		- Identificar los posibles MCVS	1 día	lun 8/8/16	lun 8/8/16	
5		- Seleccionar un modelo para el proyecto.	1 día	lun 8/8/16	lun 8/8/16	
6		Proceso de Iniciación, Planificación y Estimación del Proyecto	5 días	mar 9/8/16	lun 15/8/16	3
7		- Establecer la matriz de actividades para el MCVS	1 día	mar 9/8/16	mar 9/8/16	
8		- Asignar los recursos del proyecto.	1 día	mié 10/8/16	mié 10/8/16	
9		- Definir el entorno del proyecto.	1 día	jue 11/8/16	jue 11/8/16	
10		- Planificar la gestión del proyecto.	2 días	vie 12/8/16	lun 15/8/16	
11		Proceso de Seguimiento y Control del Proyecto	2 días	mar 16/8/16	mié 17/8/16	6
12		- Analizar riesgos.	2 días	mar 16/8/16	mié 17/8/16	
13		Proceso de Exploración de Conceptos	4 días	jue 18/8/16	mar 23/8/16	11
14		- Identificar las ideas o necesidades.	1 día	jue 18/8/16	jue 18/8/16	
15		- Formular las soluciones potenciales.	1 día	jue 18/8/16	jue 18/8/16	
16		- Dirigir los estudios de viabilidad.	1 día	vie 19/8/16	vie 19/8/16	
17		- Planificar la transición del sistema (si se aplica).	1 día	vie 19/8/16	vie 19/8/16	
18		- Refinar y Finalizar la idea o necesidad.	1 día	vie 19/8/16	vie 19/8/16	
19		- Planificar la transición del sistema (si se aplica).	3 días	jue 18/8/16	lun 22/8/16	
20		Hito Nro 1	0 días	vie 26/8/16	vie 26/8/16	
21		Maqueta 1	15 días	lun 29/8/16	vie 16/9/16	20
22		Proceso de Seguimiento y Control del Proyecto	3 días	sáb 27/8/16	mar 30/8/16	
23		- Analizar riesgos.	2 días	sáb 27/8/16	lun 29/8/16	
24		- Realizar la planificación de contingencias.	2 días	sáb 27/8/16	lun 29/8/16	
25		- Archivar registros.	1 día	mar 30/8/16	mar 30/8/16	
26		Proceso de Gestión de Calidad del Software	2 días	mié 31/8/16	jue 1/9/16	
27		- Planificar la garantía de la calidad del software.	2 días	mié 31/8/16	jue 1/9/16	
28		- Desarrollar métricas de calidad.	2 días	mié 31/8/16	jue 1/9/16	
29		Proceso de Exploración de Conceptos	3 días	vie 2/9/16	mar 6/9/16	
30		- Identificar las ideas o necesidades.	1 día	vie 2/9/16	vie 2/9/16	
31		- Formular las soluciones potenciales.	1 día	vie 2/9/16	vie 2/9/16	
32		- Dirigir los estudios de viabilidad.	1 día	lun 5/9/16	lun 5/9/16	
33		- Planificar la transición del sistema (si se aplica).	1 día	lun 5/9/16	lun 5/9/16	
34		- Refinar y Finalizar la idea o necesidad.	1 día	mar 6/9/16	mar 6/9/16	
35		Proceso de Análisis de Requisitos	3 días	mié 7/9/16	vie 9/9/16	
36		- Definir los requisitos de interfaz.	3 días	mié 7/9/16	vie 9/9/16	
37		Proceso de Diseño	2 días	sáb 10/9/16	lun 12/9/16	
38		- Realizar el diseño preliminar.	2 días	sáb 10/9/16	lun 12/9/16	
39		Proceso de Implementación e Integración	2 días	mar 13/9/16	mié 14/9/16	
40		- Crear la documentación de operación.	2 días	mar 13/9/16	mié 14/9/16	
41		Proceso de verificación y validación	1 día	jue 15/9/16	jue 15/9/16	
42		- Planificar la verificación y validación.	1 día	jue 15/9/16	jue 15/9/16	

Figura 2. Tareas del proyecto (I).

























































43		Hito Nro 2	1 día	jue 22/9/16	jue 22/9/16
44		Prototipo Evolutivo 1	20 días	vie 23/9/16	jue 20/10/16 43
45		Proceso de Seguimiento y Control del Proyecto	1 día	vie 23/9/16	vie 23/9/16
46		- Analizar riesgos.	1 día	vie 23/9/16	vie 23/9/16
47		- Realizar la planificación de contingencias.	1 día	vie 23/9/16	vie 23/9/16
48		- Archivar registros.	1 día	vie 23/9/16	vie 23/9/16
49		Proceso de Análisis de Requisitos	2 días	sáb 24/9/16	lun 26/9/16
50		- Definir y Desarrollar los requisitos del software.	1 día	sáb 24/9/16	sáb 24/9/16
51		- Priorizar e Integrar los requisitos del software.	1 día	lun 26/9/16	lun 26/9/16
52		Proceso de Diseño	4 días	mar 27/9/16	vie 30/9/16
53		- Realizar el diseño preliminar.	1 día	mar 27/9/16	mar 27/9/16
54		- Analizar el flujo de información.	1 día	mar 27/9/16	mar 27/9/16
55		- Diseñar la base de datos (si se aplica).	1 día	mié 28/9/16	mié 28/9/16
56		- Diseñar las interfaces.	2 días	jue 29/9/16	vie 30/9/16
57		- Realizar el diseño detallado.	3 días	mié 28/9/16	vie 30/9/16
58		Proceso de Implementación e Integración	9 días	sáb 1/10/16	mié 12/10/16
59		- Crear los datos de prueba.	1 día	sáb 1/10/16	sáb 1/10/16
60		- Crear el código fuente.	6 días	sáb 1/10/16	vie 7/10/16
61		- Generar el código objeto.	6 días	sáb 1/10/16	vie 7/10/16
62		- Crear la documentación de operación.	2 días	sáb 8/10/16	lun 10/10/16
63		- Planificar la integración.	2 días	sáb 8/10/16	lun 10/10/16
64		- Realizar la integración.	2 días	mar 11/10/16	mié 12/10/16
65		Proceso de verificación y validación	4 días	jue 13/10/16	mar 18/10/16
66		- Ejecutar las tareas de verificación y validación.	1 día	jue 13/10/16	jue 13/10/16
67		- Recoger y analizar los datos de las métricas	1 día	vie 14/10/16	vie 14/10/16
68		- Planificar las pruebas.	2 días	jue 13/10/16	vie 14/10/16
69		- Desarrollar las especificaciones de las pruebas.	2 días	sáb 15/10/16	lun 17/10/16
70		- Ejecutar las pruebas.	2 días	lun 17/10/16	mar 18/10/16
71		Hito Nro 3	0 días	jue 20/10/16	jue 20/10/16
72		Prototipo Evolutivo 2	15 días	vie 21/10/16	jue 10/11/16 44
73		Proceso de Seguimiento y Control del Proyecto	1 día	vie 21/10/16	vie 21/10/16
74		- Analizar riesgos.	1 día	vie 21/10/16	vie 21/10/16
75		- Realizar la planificación de contingencias.	1 día	vie 21/10/16	vie 21/10/16
76		- Archivar registros.	1 día	vie 21/10/16	vie 21/10/16
77		Proceso de Análisis de Requisitos	2 días	sáb 22/10/16	lun 24/10/16
78		- Definir y Desarrollar los requisitos del software.	1 día	sáb 22/10/16	sáb 22/10/16
79		- Priorizar e Integrar los requisitos del software.	1 día	lun 24/10/16	lun 24/10/16
80		Proceso de Diseño	4 días	mar 25/10/16	vie 28/10/16
81		- Realizar el diseño preliminar.	1 día	mar 25/10/16	mar 25/10/16
82		- Analizar el flujo de información.	1 día	mar 25/10/16	mar 25/10/16
83		- Diseñar la base de datos (si se aplica).	1 día	mié 26/10/16	mié 26/10/16
84		- Diseñar las interfaces.	2 días	jue 27/10/16	vie 28/10/16
85		- Realizar el diseño detallado.	3 días	mié 26/10/16	vie 28/10/16
86		Proceso de Implementación e Integración	5 días	sáb 29/10/16	jue 3/11/16
87		- Crear los datos de prueba.	1 día	sáb 29/10/16	sáb 29/10/16
88		- Crear el código fuente.	4 días	sáb 29/10/16	mié 2/11/16
89		- Generar el código objeto.	4 días	sáb 29/10/16	mié 2/11/16
90		- Crear la documentación de operación.	2 días	mié 2/11/16	jue 3/11/16
91		- Planificar la integración.	1 día	mié 2/11/16	mié 2/11/16
92		- Realizar la integración.	1 día	jue 3/11/16	jue 3/11/16
93		Proceso de verificación y validación	3 días	vie 4/11/16	mar 8/11/16
94		- Ejecutar las tareas de verificación y validación.	1 día	vie 4/11/16	vie 4/11/16
95		- Recoger y analizar los datos de las métricas	1 día	vie 4/11/16	vie 4/11/16
96		- Planificar las pruebas.	1 día	vie 4/11/16	vie 4/11/16
97		- Desarrollar las especificaciones de las pruebas.	2 días	sáb 5/11/16	lun 7/11/16
98		- Ejecutar las pruebas.	2 días	lun 7/11/16	mar 8/11/16

Figura 2. Tareas del proyecto (II).

99		Hito Nro 4	0 días	jue 10/11/16	jue 10/11/16
100		Instalacion y Operación	15 días	vie 11/11/16	jue 1/12/16 72
101		Proceso de Instalación y Aceptación	5 días	vie 11/11/16	jue 17/11/16
102		- Planificar la instalación.	2 días	vie 11/11/16	lun 14/11/16
103		- Instalar el software.	2 días	mar 15/11/16	mié 16/11/16
104		- Cargar la base de datos (si se aplica).	1 día	mié 16/11/16	mié 16/11/16
105		- Realizar las actualizaciones.	2 días	mié 16/11/16	jue 17/11/16
106		Proceso de Mantenimiento	5 días	vie 18/11/16	jue 24/11/16
107		- Realizar el mantenimiento correctivo.	5 días	vie 18/11/16	jue 24/11/16
108		Proceso de verificación y validación	5 días	jue 24/11/16	mié 30/11/16
109		- Ejecutar las tareas de verificación y validación.	2 días	jue 24/11/16	vie 25/11/16
110		- Ejecutar las pruebas.	3 días	sáb 26/11/16	mar 29/11/16

Figura 2. Tareas del proyecto (III)



Figura 3. Diagrama de Gantt (Sin las tareas contenidas en los procesos).

6) Responsables

Se identifican las actividades más relevantes en el proyecto (ver Figura 4) y los responsables de las mismas (ver tabla II: responsables de las actividades del proyecto).

Figura 4. Mapa de Actividades (I)

Actividades	Prototipo Maqueta (1 iteración)				Prototipo Evolutivo (2 iteraciones)				Instalación y Operación	
	AP	ERI	DEI	VIU	ERS	DAS	C	PP	IF	OM
Proceso de Selección de un MCVS										
- Identificar los posibles MCVS	X									
- Seleccionar un modelo para el proyecto.	X									
Proceso de Iniciación, Planificación y Estimación del Proyecto										
- Establecer la matriz de actividades para el MCVS	X									
- Asignar los recursos del proyecto.	X	X	X	X	X	X	X	X		
- Definir el entorno del proyecto.	X									
- Planificar la gestión del proyecto.	X									
Proceso de Seguimiento y Control del Proyecto										
- Analizar riesgos.	X	X	X	X	X	X	X			
- Realizar la planificación de contingencias.		X	X	X	X	X	X	X		
- Gestionar el proyecto.		X	X	X	X	X	X	X		
- Archivar registros.		X	X	X	X	X	X	X		
Proceso de Gestión de Calidad del Software										
- Desarrollar métricas de calidad.		X	X		X	X	X			
- Gestionar la calidad del software.		X	X	X	X	X	X	X		
Proceso de Exploración de Conceptos										
- Identificar las ideas o necesidades.	X	X			X					
- Formular las soluciones potenciales.	X	X			X					
- Dirigir los estudios de viabilidad.	X	X			X					
- Planificar la transición del sistema (si se aplica).	X	X			X					
- Refinar y Finalizar la idea o necesidad.	X	X			X					
Proceso de Análisis de Requisitos										
- Definir y Desarrollar los requisitos del software.					X					
- Definir los requisitos de interfaz.		X								
- Priorizar e Integrar los requisitos del software.					X					
Proceso de Diseño										
- Realizar el diseño preliminar.			X			X				
- Analizar el flujo de información.						X				

- Diseñar la base de datos (si se aplica).						X				
- Diseñar las interfaces.						X				
- Realizar el diseño detallado.						X				
Proceso de Implementación e Integración										
- Crear los datos de prueba.						X	X			
- Crear el código fuente.							X			
- Generar el código objeto.							X			
- Crear la documentación de operación.			X			X	X			
- Planificar la integración.						X	X			
- Realizar la integración.							X			
Proceso de Instalación y Aceptación										
- Planificar la instalación.									X	
- Instalar el software.									X	
- Cargar la base de datos (si se aplica).									X	
- Realizar las actualizaciones.									X	
Proceso de Mantenimiento										
- Realizar el mantenimiento correctivo.										X
Proceso de verificación y validación										
- Planificar la verificación y validación.		X			X					
- Ejecutar las tareas de verificación y validación.				X				X	X	
- Recoger y analizar los datos de las métricas				X				X		
- Planificar las pruebas.			X			X				
- Desarrollar las especificaciones de las pruebas.			X			X	X			
- Ejecutar las pruebas.				X				X	X	

Figura 4. Mapa de Actividades (II)

TABLA IV. RESPONSABLES DE ACTIVIDADES DEL PROYECTO (I)

Identificación de actividad	Detalles	
	Descripción de actividad	Responsable/s
Identificar posibles ciclos de vida	Se reúne un grupo de MCV (Modelo de Ciclo de Vida) posibles para el proyecto a desarrollar.	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Seleccionar un modelo para el proyecto	Se elige uno de los ciclos de vida posibles recogidos en la actividad anterior	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Establecer matriz de actividades	Elaboración de mapa de actividades del proyecto	Nazareno Galvan, Agustín Di Stefano, Alejandro Canosa
Asignar recursos	Se asignan los recursos necesarios para llevar a cabo el proyecto	Alejandro Canosa

Definir el entorno del proyecto	Definir Hardware y Software necesario para el correcto desarrollo del proyecto	Alejandro Canosa, Agustín Di Stefano, Nazareno Galvan
Planificar la gestión del proyecto	Planificación del control del proyecto	Alejandro Canosa
Analizar riesgos.	Análisis de riesgos potenciales que podrían aparecer durante el desarrollo del proyecto	Alejandro Canosa
Realizar la planificación de contingencias	Generar un plan que contenga las contramedidas en caso de que un riesgo se materialice	Alejandro Canosa
Gestionar el proyecto	Mediante el plan de gestión, controlar el desarrollo del proyecto	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

TABLA IV. RESPONSABLES DE LAS ACTIVIDADES DEL PROYECTO (II)

Identificación de la actividad	Detalles	
	Descripción	Responsables
Analizar riesgos.	Análisis de riesgos potenciales que podrían aparecer durante el desarrollo del proyecto	Alejandro Canosa
Realizar la planificación de contingencias	Generar un plan que contenga las contramedidas en caso de que un riesgo se materialice	Alejandro Canosa
Gestionar el proyecto	Mediante el plan de gestión, controlar el desarrollo del proyecto	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Archivar registros	Auto explicativo	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Desarrollar métricas de calidad.	Métricas usadas por el equipo para medir la calidad del software.	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Gestionar la calidad del software.	Auto explicativo	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Identificar las ideas o necesidades	Identificación de las ideas o necesidades del producto a desarrollar y su documentación	Nazareno Galvan
Formular las soluciones potenciales	Formulación de las posibles soluciones al proyecto en base a la identificación de ideas o necesidades	Nazareno Galvan
Dirigir estudios de viabilidad	Análisis del problema para determinar si el producto software es viable o no	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la transición del sistema	Generar un documento que contenga el plan de instalación	Agustín Di Stefano
Refinar y finalizar la idea o necesidad	Modificar ideas o necesidades en base a las soluciones encontradas	Nazareno Galvan
Definir y Desarrollar los Requisitos de software	Establecer requisitos funcionales, no funcionales, de interfaz, etc. Del proyecto y documentarlos	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Definir los requisitos de interfaz	Definir requisitos de la interfaz de usuario del proyecto	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Priorizar e integrar los requisitos del software	Establecer prioridades a cada uno de los requisitos del software a construir y documentarlos en un Documento de Requisitos	Alejandro Canosa
Realizar el diseño preliminar	Se planifica el proyecto software a través de documentos de diseño según el paradigma a aplicar, en este, se usaran las herramientas UML para el Paradigma Orientado a	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

	Objetos	
Analizar el flujo de información	Análisis del camino que realizara la información a través del sistema	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Diseñar la base de datos	Diseño de las tablas de la Base de datos en base a las relaciones y atributos necesarios para satisfacer los requisitos	Alejandro Canosa
Diseñar las interfaces	Diseño de interfaces de usuario	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar el diseño detallado	Realización de implementaciones de código del producto (clases de implementación) según los datos obtenidos en el diseño preliminar y el documento de requisitos	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Crear los datos de prueba	Se crean los datos que diferencien los casos de prueba	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Crear el código fuente	Codificar el producto software a raíz de los requisitos establecidos	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Generar código objeto		Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Crear la documentación de operación	Se realiza el manual de usuario para el uso del producto software	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la integración.	Planificación de la unión de los módulos del sistema	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar la integración.	Realizar la integración de los módulos del sistema	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la instalación.	Planificar la instalación del software en el entorno	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Instalar el software.	Instalar el software en el entorno tal como se planifico	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Cargar la base de datos (si se aplica).	Cargar la base de datos del software instalado en caso de necesitar datos predefinidos	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

TABLA IV. RESPONSABLES DE ACTIVIDADES DEL PROYECTO (III)

Identificación de actividad	Detalles	
	Descripción	Responsable/s
Realizar las actualizaciones.	Actualizar el producto software agregándole funcionalidades	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar el mantenimiento correctivo	Realizar le mantenimiento correctivo del software corrigiendo bugs encontrados en el producto instalado	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la verificación y validación.	Realizar el Plan de Verificación y Validación, que consiste en la correcta construcción y diseño del software	Alejandro Canosa
Ejecutar las tareas de verificación y validación.	Realizar revisiones de los documentos	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Recoger y analizar los datos de las métricas	A partir de las métricas establecidas, analizar el software y establecer que tan bien se cumplen	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar las pruebas.	Realizar el plan de pruebas	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Desarrollar las especificaciones de las pruebas.	Realizar los casos de prueba	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Ejecutar las pruebas.	Realizar la ejecución de las pruebas y documentar los resultados	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

7) *Objetivos y prioridades de Gestión*

El objetivo de la Gestión es lograr controlar los procesos software que se llevarán a cabo en la realización del proyecto, así como la gestión de calidad, la verificación y validación y los cambios del mismo.

La gestión comprende 4 actividades

- Iniciación
- Planificación
- Ejecución
- Control

El fin de la iniciación es poder definir la propuesta del proyecto, sus objetivos, alcance, calidad y estimar los riesgos del mismo, es una actividad de alta prioridad ya que si este proceso está definido de manera incorrecta llevara a un desarrollo exhaustivo del proyecto, básicamente. En esta actividad se realizarán consultas al cliente por medio de entrevistas sobre los objetivos del proyecto a desarrollar.

La planificación conllevara la realización del diagrama de GANTT y matriz de actividades, la elaboración del plan de Gestión, el análisis de riesgos, el plan de Contingencias y el diseño y codificación del proyecto. La codificación y las pruebas forman parte de esta actividad o etapa de gestión.

La ejecución refiere a la implementación o puesta en marcha del proyecto, consiste en poner en práctica la planificación llevada a cabo previamente. Se realizará mediante el manual de instalación producido durante el proyecto. En esta actividad se llevará a cabo el plan de transición del proyecto, se presentaran los manuales de Canosa No Alejandro A., Galvan Gabriel N., Di Stefano Agustín. L. . 2016. *Ambiente de AutoEvaluación (AAEV 2.0)*. Cátedra Proyecto de Software 2016

instalación y de usuario, se revisaran los prototipos y el cliente aprobará o especificará los cambios necesarios para la instalación del proyecto en el entorno.

El fin de las actividades de control es asegurar que los objetivos sean alcanzados en el tiempo y calidad planificada, realizando una buena supervisión y medición del rendimiento de los resultados, con el objetivo de que se puedan tomar acciones correctivas, esto se hace mediante la comparación entre la planificación realizada y los valores incurridos. En esta actividad se llevara a cabo la Gestión de calidad, de configuración y el proceso de verificación y validación.

8) *Condiciones asumidas, dependencias y restricciones*

En este momento el proyecto no contiene restricciones.

Dependencias: aceptación del cliente.

Condiciones asumidas: 4 hitos de entrega con fechas predefinidas:

- Hito 1. Análisis y diseño inicial.
- Hito 2. Prototipo 1. Maqueta.
- Hito 3. Prototipo 2. Prototipo Funcional
- Hito 4. Presentación Global del Proyecto.

Realización de presentación Técnica que describan las decisiones realizadas durante los hitos.

Cada hito y documentación final deberá ser entregado en el formato de la revista Latinoamericana de Ingeniería de Software.

9) *Gestión de Riesgos*

La identificación de riesgos, su análisis, prevención y mitigación se ven en el Plan de Contingencias.

10) *Mecanismos de control y ajuste*

Consultas al cliente: vía mail, Skype o presencial.

Reportes presenciales al cliente o por medio del aula virtual del campus de la UNLA.

Auditorías: presenciales, teniendo la configuración de los documentos revisados de acuerdo al plan de Gestión de Configuración

11) *Mecanismos para la Gestión de calidad*

Las actividades más importantes de la Gestión de Calidad de este proyecto son:

- Pruebas de Software: reunión de información, creación de casos de prueba, registro de los mismos, ejecución de las pruebas y registro de los resultados

El monitoreo y control de las pruebas será realizado por el equipo. En caso de posible modificación se notificaran al resto del equipo y se evaluara en conjunto para afirmar la necesidad de actualizar los documentos o descartar el posible cambio. Cada Prueba estará documentada, así como cada caso de prueba y la planificación para llevar a cabo las pruebas de software. Se verificarán los resultados obtenidos en las mismas

12) Mecanismos para la Gestión de Proyecto

Las actividades más importantes de la Gestión de Proyecto son:

- Analizar riesgos
- Realizar la planificación de contingencias

La gestión se monitorea con el presente documento, teniendo en cuenta la revisión del mismo al inicio de cada fase, y las posibles modificaciones que este pueda tener.

- la versión de escritorio teniendo cada uno un repositorio local en su pc.

Experiencia:

- Codificación (lógica):
 - Alejandro Canosa: Media/alta
 - Nazareno Galvan: Media/alta
 - Agustín Di Stefano: Media/alta
- Codificación (interfaz de usuario):
 - Alejandro Canosa: Media
 - Nazareno Galvan: Baja
 - Agustín Di Stefano: Alta
- Ingeniería de software:
 - Alejandro Canosa: Media
 - Nazareno Galvan: Media
 - Agustín Di Stefano: Media

14) Procedimientos técnicos, herramientas y tecnologías

Requerimientos técnicos:

- Sistema Operativo: Debían Jessie
- Servidor Web: Apache2
- Gestor de Base de datos: MySQL v5.5
- Lenguajes: Python v2.7
- Framework: Django Instalado 1.7.7

Herramientas de diseño:

- Software Microsoft Visio 2007

Estándar especificado: IEEE 1074/97

15) Documentación de software

La documentación se realizará de acuerdo a las plantillas de documentos definidas en el Modelo de Proceso para desarrollo de Software del estándar IEEE 1074/97

El formato de las mismas será según la Revista Latinoamericana de Ingeniería de Software

b) Estimación de esfuerzo y Puntos de Función por medio de USC COCOMO II (Figuras 5, 6 y 7)

Medida	Formula	Resultados
Esfuerzo	$A * KLOC^b$	Persona-mes
Tiempo	$c * E^d$	Meses
Personas	E/T	

Figura 5. Esfuerzos (usando modelo básico con proyectos medios)

13) Recursos

El equipo de proyecto está compuesto por Alejandro Canosa, Agustín Di Stefano y Nazareno Galvan.

Recursos tecnológicos de los miembros:

- Entornos de desarrollo
- Repositorio Git para control de versiones, control de documentación y gestión. Se usa la pagina web GitHub y

Siendo $a = 3.0$; $b = 1.12$ $c = 2.5$; $d = 0.35$ y $KLOC = 3551$

Esfuerzo = 12,2

Tiempo = 6 meses

Personas = 2,03

Siendo 3 personas, el desarrollo seria en 3 meses aproximadamente, nótese que cada uno de los resultados son aproximaciones, dado que es solo una estimación.

Ver figuras 6 y 7.

B. Proceso de seguimiento y control del proyecto

a) Plan de Contingencias

Este documento registra los posibles riesgos del Ambiente de AutoEvaluación 2.0 (AAEV 2.0), así como su impacto, costo, recursos afectados y consecuencias. Al finalizar el análisis de requisitos se elaborará el Plan de Contingencias que contiene las medidas a realizar en caso de que un riesgo ocurra. El equipo tendrá acceso al documento y lo consultará en caso de que la probabilidad de materialización de algún riesgo sea alta, registrado o no. En caso de que el riesgo no esté registrado, se realizarán las acciones correspondientes a la configuración del documento (detallados en el Plan de Gestión).

1) Componentes del sistema

Se detallan a continuación todos los componentes del sistema que podrían verse afectados por riesgos:

- Base de Datos: Guarda los datos registrados en el sistema, tales como alumnos, docentes, administradores... etc.
- Documentación: Se registran los planes, el diseño y manuales basados en el proyecto software
- Recursos Humanos: equipo al que se le fue asignado el proyecto
- Archivos de código fuente: Archivos que contienen la lógica del producto (conectan con la base de datos, validan, etc.
- Producto software: proyecto instalado en el ambiente pedido

2) Estructura del proyecto y dependencias

Se describe la estructura del proyecto y los procesos basados en el estándar IEEE por las que transitará hasta llegar al producto software, cada proceso es dependiente del anterior, desde el primero al último:

- Procesos de Gestión del Proyecto.

- Pre-Desarrollo
- Desarrollo
- Post-Desarrollo
- Procesos integrales (Algunos procesos contenidos se realizan a lo largo del ciclo de vida del proyecto)

b) Servicios y materiales utilizados en el proyecto

Los servicios descritos a continuación ayudan al desarrollo del sistema, estos servicios son ajenos al equipo y su falta puede impactar en el desarrollo del proyecto, retrasándolo.

Comunicaciones: e-mail, Whatsapp, Skype, Facebook, SMS, usado para dividir las tareas o consultas al equipo o docentes.

Internet: La mayoría de las comunicaciones e investigaciones realizadas se basan en el uso de este servicio, si este cae, las comunicaciones e investigaciones serán menos efectivas.

Project Name: AAEV 2.0

Scale Factor: 18.97

Schedule

Project Notes

Development Model: Early Design

X	Module Name	Module Size	LABOR Rate (\$/month)	EAF	Language	NCM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	AAEV 2.0	F:3551	0.00	0.74	Non-Specified	10.8	8.0	446.6	0.00	0.0	1.2	0.0

Figura 6. Estimación de Puntos de Función obtenida por USC COCOMO II (I)

Total Lines of Code: 3551		Estimated Effort		Sched		PROD		COST		INST		Staff		RISK	
Hours/PM: 152.00		Optimistic	5.3	6.1	666.5	0.00	0.0	0.9							
		Most Likely	8.0	6.9	446.6	0.00	0.0	1.2	0.0						
		Pessimistic	11.9	7.8	297.7	0.00	0.0	1.5							

Figura 7. Estimación de Puntos de Función obtenida por USC COCOMO II (II)

1) Identificación de riesgos

Amenazas:

Las amenazas de un proyecto son los factores externos o internos que impactan de manera negativa sobre el mismo, ya sea un atraso en la realización de tareas, perdida de documentación, entre otras. Las amenazas encontradas son:

- Un miembro del equipo no está disponible ya sea por problemas personales, de salud u otros durante el desarrollo del proyecto.
- Requisitos mal obtenidos o no obtenidos durante la educación de los mismos
- Falta de tiempo
- Mal funcionamiento del producto
- Perdida de documentación o datos
- Corte temporal de servicios de internet o luz
- Documentación mal desarrollada
- Diseño mal desarrollado

2) Riesgos de proyecto

Los riesgos de proyecto son aquellos riesgos que atrasan el desarrollo del mismo y aumenta sus costos, generalmente se encuentran en la planificación, los requisitos obtenidos, la indisponibilidad de recursos, entre otros.

Los posibles riesgos de proyecto son:

- Atraso en la entrega de documentación establecida en los hitos
- Documentación o datos perdidos
- Incomunicación en el equipo

- Falta de recursos humanos en ciertas tareas
- No es posible compartir la documentación entre el equipo

3) Riesgos técnicos

Los riesgos técnicos refieren al diseño del proyecto, básicamente, se diseña algo más pequeño de lo que en verdad es. Están asociados a la planificación, validación, entre otros

Los posibles riesgos técnicos son:

- Planes mal diseñados
- Producto que el cliente no desea

4) Riesgos de negocio

Los riesgos de negocio son los riesgos del mercado que afectan al producto software, tales como la necesidad del mismo en el mercado, competencia con otros software de mejor calidad, etc. Sin embargo, no se encontraron riesgos de negocio en el proyecto.

5) Análisis de riesgos

El análisis de riesgo define el momento en que puede darse cada riesgo identificado, la probabilidad de ocurrencia de los mismos y el impacto que tiene sobre el proyecto. Los riesgos tienen un ID único asignado y se ordenan por prioridad en orden descendente. Una vez realizado el análisis se documentan las medidas de prevención/mitigación en el Plan de Contingencias

6) Matriz de probabilidad de impacto (Ver Figura 8)

Se define la probabilidad de ocurrencia por las siguientes definiciones:

- Muy baja: menos del 10%

- Baja: del 10% al 25%
- Media: de 25% al 50%
- Alta: del 50% al 75%
- Muy alta: más del 75%

Se define el impacto por las siguientes definiciones:

- Insignificante: no merece ser tenido en cuenta
- Tolerable: no comprometen al proyecto ni al producto, están en un marco de aceptación
- Graves: comprometen gravemente el proyecto o el producto
- Catastrófica: Amenazan la supervivencia del proyecto o del producto

Probabilidad	Amenaza				
Muy alta	MB	B	M	A	MA
Alta	MB	B	M	A	MA
Media	MB	B	M	A	MA
Baja	MB	B	M	A	MA
Muy baja	MB	B	M	A	MA

MB: Muy bajo, B: Bajo, M: Medio, A: Alto, MA: Muy alto

Figura 8. Matriz de probabilidad e impacto

El color verde indica que es un riesgo de prioridad baja, el amarillo que es de prioridad media y rojo que es de prioridad alta.

7) Clasificación de riesgos

TABLA V. CLASIFICACIÓN DE RIESGOS

ID de Riesgo (ordenado por prioridad descendente)	Detalles				
	Nombre	Probabilidad	Impacto	Momento	Consecuencias
1	Planes de Gestión pobres	Media	Grave	Inicio/desarrollo	Retraso en desarrollo de proyecto, progreso dificultoso del mismo
2	Requisitos mal comprendidos	Baja	Grave	Inicio/Desarrollo/validación	Retraso en desarrollo del proyecto, producto software no pedido
3	Perdida de documentación	Baja	Grave	Inicio/Desarrollo	Retraso en desarrollo de proyecto, documentación y producto no consistentes
4	Requisitos no obtenidos	Baja	Tolerable	Inicio/Desarrollo/validación	Producto software incompleto

					pleto
5	Falta de tiempo para el hito de entrega	Media	Tolerable	Inicio/Desarrollo/instalación	Falta de documentación pedida en entrega, producto software incompleto
6	Indisponibilidad de RRHH	Baja	Tolerable	Inicio/Desarrollo/instalación	Documentación incompleta, huecos en la división de trabajo
7	Mal funcionamiento del producto	Muy baja	Tolerable	Instalación	Desaprobación del cliente

c) Plan de Contingencias

A continuación se detallaran las medidas a realizar para prevenir/mitigar los riesgos en caso de aparición de los mismos durante el desarrollo del proyecto

1) Plan de respaldo

El plan de emergencia indica las acciones a realizar durante la materialización de una amenaza. A continuación se indica los planes de emergencia de cada uno de los riesgos.

TABLA VI. PLAN DE RESPALDO

ID de Riesgo	Detalles	
	Riesgo	Plan de respaldo
1	Planes mal diseñados	Revisión y corrección del Plan de Gestión antes de empezar los procesos de desarrollo
2	Sistema no válido para el cliente	Confirmación del cliente de los requisitos presentados en el Documento de Requisitos de Usuario
3	Documentación o datos perdidos	Backup en un repositorio online en cada una de las máquinas del equipo
4	Atraso en la entrega de documentación establecida en los hitos	Elaboración de requisitos en un margen de tiempo prudente
5	No es posible compartir la documentación entre el equipo	Subir la documentación al repositorio online apenas se acaba
6	Falta de recursos Humanos en ciertas tareas	Tener una previa división de trabajo establecida en caso de que un miembro del equipo no esté disponible en un determinado momento
7	Incomunicación del equipo	Establecer horarios de reuniones presenciales
8	Mal funcionamiento del producto	Verificar el código fuente y el diseño mediante los procesos de verificación y validación

1) Plan de emergencia

El plan de emergencia indica las acciones a realizar durante la materialización de una amenaza. A continuación se indica los planes de emergencia de cada uno de los riesgos

TABLA VII. PLAN DE EMERGENCIA

ID de Riesgo	Detalles	
	Riesgo	Plan de emergencia
1	Planes mal diseñados	Revisión y corrección de aquellos planes que no están bien establecidos en el proyecto
2	Sistema no válido para el cliente	Planificación de próxima entrevista para la educación de requisitos
3	Documentación o datos perdidos	Recurrir al backup en el repositorio online o a un compañero de equipo que contenga los archivos perdidos
4	Atraso en la entrega de documentación establecida en los hitos	Planificar la entrega para el hito de recuperatorio, avisar al cliente
5	No es posible compartir la documentación entre el equipo	Pasar la documentación por pendrive presencialmente, o por Bluetooth del dispositivo móvil. Otra alternativa es ir hacia una computadora con internet y enviar los documentos desde ahí

TABLA VIII. PLAN DE EMERGENCIA (II)

ID de Riesgo	Detalles	
	Riesgo	Plan de emergencia
6	Falta de recursos Humanos en ciertas tareas	Tener una previa división de trabajo establecida en caso de que un miembro del equipo no esté disponible en un determinado momento
7	Incomunicación del equipo	Contactar presencialmente con los miembros del equipo. En caso de no encontrarlos redividir el trabajo
8	Mal funcionamiento del producto	Anotar errores del sistema y reparar el código fuente en el mantenimiento del mismo

2) Plan de recuperación

Su fin es poder regresar los elementos del sistema al momento anterior a la materialización de la amenaza. A continuación se presenta el Plan de Recuperación los siguientes riesgos:

TABLA IX. PLAN DE RECUPERACIÓN

ID de Riesgo	Detalles	
	Riesgo	Plan de recuperación
3	Documentación o datos perdidos	Copiar contenidos del backup al lugar correspondiente del documento. Copiar los datos al lugar correspondiente del proyecto. Si se pierden los datos del repositorio por una actualización, revertirla

3) Requerimientos y ejecución del plan de recuperación

En caso de que la pérdida sea local, se detallan los pasos a seguir para ejecutar el plan de recuperación:

- Entrar al repositorio
- En la rama maestra, elegir la opción “Clonar o descargar”
- Elegir “Descargar como zip” para descargar el repositorio en ese formato, guardarlo en la carpeta deseada y buscar el documento
- Otra manera es seleccionar el nombre del archivo que se perdió y en la nueva ventana elegir la opción “Descargar”

En caso de pérdida en el repositorio online por un merge (fusión de 2 ramas del repositorio) con un error, seleccionar el nombre de la actualización (o la primera de la lista en el historial de actualizaciones) y elegir la opción “Revertir” esto deshará todos los cambios hechos en el repositorio y lo devolverá a su estado anterior antes de la fusión.

C. Proceso de Gestión de la calidad del software

Este proceso se centra en el control de la calidad del producto software y su manutención.

a) Plan de garantía de calidad del software.

Este plan se centra en la descripción de las actividades del equipo para mantener y mejorar la calidad del software, así como documentar las métricas con las cuales evaluar las mismas.

La garantía de la misma está dada por las revisiones y correcciones continuas a los documentos, así como su elaboración a tiempo para facilitar el desarrollo del proyecto. Las pruebas y el proceso de verificación y validación contribuyen al desarrollo de la calidad del software debido a su constante refinamiento y reducción de errores.

El proyecto usará el patrón de diseño MVC, que consta de la división del programa en Modelo, donde están establecidos los datos que usa el sistema, la Vista, que es el aspecto visual del mismo (la interfaz de usuario) y el Controlador, el cual realiza toda la lógica del sistema y enlaza el Modelo y la Vista, creando un sistema integro.

Las métricas de calidad del software son:

- Cantidad de errores/deficiencias: los errores de código, diseño, etc.
- Amigabilidad con el usuario: Si el sistema provee información importante y evita la confusión del usuario.
- Facilidad de uso: que el sistema sea fácil de usar

III. PROCESOS DE PRE-DESARROLLO

A. Proceso de Exploración de Conceptos

a) Modelo del Dominio del Problema.

Nuestra app está enfocada a que el alumno que esta cursando determinada materia pueda ver su nivel de conocimiento, acerca de los temas dados por el profesor,

mediante exámenes con una forma de resolución de multiple choice, selección multiple o selección de imágenes.

Algunos de los problemas de nuestra app:

- Los docentes no deben saber la cantidad de intentos que hicieron sus alumnos al intentar resolver sus exámenes.
- Los alumnos no pueden alterar las respuestas.
- Solo el administrador puede dar de alta a los docentes.
- No se deben mezclar las respuestas correctas con las incorrectas.
- El alumno no debe ingresar a materias a las que el docente no le dio acceso.

b) *Modelo del Dominio del Problema.*

El proyecto permite a los usuarios un ambiente de aprendizaje mediante tests de auto-evaluación. Los docentes inscriptos por el Administrador al sistema, pueden cargar preguntas con sus correspondientes respuestas y puntuación. Luego el sistema crea exámenes con preguntas aleatorias dependiendo de la materia que el docente haya cargado. Cuando el Alumno ingresa al sistema, se debe registrar en las materias para que el docente a cargo los habilite a realizar los exámenes. Una vez que el docente habilita al alumno, este puede realizar los exámenes que necesite para probar su conocimiento.

c) *Informe Preliminar de Necesidad.*

Entender el objetivo de la aplicación y las necesidades del cliente.

Se necesita establecer una herramienta de hipertexto para un buscador de exámenes donde el alumno pueda decidir como solicitar el examen que necesita mediante palabras clave, ya sea materias, docentes, contenido de la materia, entre otros.

Se necesita un sistema de comunicación vía mail, donde se notifique, por ejemplo, al alumno que el docente lo dio de alta en determinada materia.

Desarrollar la aplicación utilizando el lenguaje Python con la utilización del Framework Django.

d) *Soluciones Alternativas Posibles.*

Para la búsqueda de examen, entre mejor este organizado la jerarquía de Docente-Carrera-Materia-Guía mejor ubicado estará el alumno haciendo mas optima la búsqueda del examen. Otra alternativa es dar opciones completando la palabra, ayudando al usuario en su búsqueda. Se pueden utilizar, o no, patrones de diseño para mejorar la interfaz.

Una forma de desarrollo seria la utilización del paradigma orientado a objetos y al realizarse bajo el lenguaje Python se puede utilizar distintas herramientas y Frameworks correspondientes al lenguaje como Django. Otra herramienta a considerar seria la utilización de Material Design.

e) *Soluciones Recomendadas.*

Establecer un índice donde el alumno filtre su búsqueda mediante palabras establecidas por el sistema.

Se trabajara mediante el paradigma orientado a objetos el cual brinda el encapsulamiento de los datos además de la utilización del Framework Django que nos permitirá desarrollar el proyecto con más facilidad.

f) *Estudio de viabilidad: Análisis coste/beneficio.*

Este análisis compara los costes del desarrollo de proyecto y los beneficios del mismo para determinar si el desarrollo es viable o no. A continuación se detallan los costes:

- Costes técnicos: la capacidad del equipo para poder llevar a cabo el desarrollo del proyecto y el uso de las herramientas adecuadas.
- Costes operativos: si las soluciones a brindar sean útiles al entorno y satisfagan el problema.

Por último, tenemos los beneficios tangibles:

- Aprendizaje de uso de herramientas nuevas de desarrollo

En conclusión, el proyecto software es un proyecto viable para el desarrollo ya que no tiene costes grandes y los beneficios se equiparán con estos.

B. *Plan de transición*

a) *Introducción*

El sistema requiere tres categorías de usuarios (Administrador, Docentes, Alumnos).

El cargo de Administrador de este sistema corresponde al cliente Sebastian Martins, quien tiene la capacidad de indicar cambios en las funcionalidades del sistema para futuras actualizaciones. Sus deberes y responsabilidades son:

- Gestionar el ABM de los Docentes que quieran participar en el sistema.
- ALTA: Con el previo registro del docente en el sistema, el administrador aceptara la solicitud de incorporación al mismo, dándole al docente las funciones que le correspondan
- BAJA: Ya sea por petición expresa del docente o por cese de actividades en la institución universitaria, el administrador deberá (por medio de la interfaz de gestión de Docentes) dar de baja al docente.
- MODIFICACIÓN: El Administrador podrá (por medio de la interfaz de gestión de Docentes) modificar los datos del docente.
- Gestionar el ABM de los departamentos, carreras y materias existentes en el sistema.
- ALTA: Con la previa petición de inscripción por parte del Docente, el administrador deberá (por medio de la interfaz

de gestión de departamentos, carreras y materias) incorporar los datos del departamento, carrera, y/o materia: departamento, carrera, materia;

- **BAJA:** Solo por petición expresa de la institución universitaria, el administrador deberá (por medio de la interfaz de gestión de departamentos, carreras y materias) dar de baja el departamento, la carrera, y/o la materia (teniendo en cuenta que unas pertenecen a otras, por lo que la eliminación se produce en cascada).
- **MODIFICACIÓN:** El Administrador podrá (por medio de la interfaz de gestión de departamentos, carreras y materias) modificar los datos del departamento, la carrera, y/o la materia.
- **Baja de los Alumnos:** Por petición expresa de la institución universitaria el Administrador podrá (por medio de la interfaz de gestión de alumnos) dar de baja al Alumno.

Los Docentes habilitados en el sistema serán todos aquellos docentes abalados por la institución universitaria que deseen utilizar el sistema presente, debiendo estar éstos incorporados previamente a la plataforma por medio de una petición dirigida al Administrador que constará de: apellido y nombre del docente, legajo, departamento, carrera y materia en la que cumple el rol de docente. Sus funciones y responsabilidades son:

- Gestionar el alta de Alumnos que quieran participar en su materia asignada (por medio de la interfaz de gestión de Alumnos).
- Registrarse en el sistema por medio de un formulario de registro similar al del alumno, sin embargo, el docente debe ser aceptado por el administrador para entrar al sistema como un usuario.
- Gestionar el ABM de exámenes:
- **ALTA:** por medio de la interfaz de gestión de Exámenes el docente podrá publicar exámenes correspondientes a unidades que se aborden en la cursada de la materia.
- **BAJA/MODIFICACIÓN:** por medio de la interfaz de gestión de Exámenes el docente podrá modificar y/o dar de baja a exámenes previamente publicados.
- Gestionar el ABM de preguntas y respuestas:
- **ALTA:** por medio de la interfaz de gestión de Preguntas y Respuestas el docente podrá publicar preguntas y respuestas correspondientes a exámenes previamente publicados. Para ello el Docente deberá contar para cada pregunta con por lo menos una respuesta correcta y una valoración numérica para esta. El criterio numérico queda a elección del Docente siendo 1 (uno) el valor más bajo por defecto.
- **BAJA/MODIFICACIÓN:** por medio de la interfaz de gestión de Exámenes el Docente podrá modificar y/o dar de baja a preguntas y/o respuestas previamente publicadas.

Los Alumnos habilitados en el sistema serán todos aquellos alumnos que hayan realizado la previa inscripción vía online. Sus funciones y responsabilidades son:

- **Inscripción a Materias:** A partir de la interfaz de Materias el alumno podrá seleccionar la mater (perteneciente al departamento y carrera previamente habilitada) para inscribirse a la misma, sin embargo éste deberá contar con la posterior habilitación del Docente para tener acceso a los exámenes de la materia en cuestión.
- **Realización de exámenes:** Por medio de la interfaz de Exámenes el alumno podrá seleccionar los exámenes que desee realizar y completar para luego recibir el resultado de su evaluación.

b) Estrategia de transición

El equipo integrado por Di Stefano, Galvan, y Canosa acudirán al encuentro del Administrador Sebastian Martins en las instalaciones de la Universidad Nacional de Lanús con una copia del producto software y la documentación correspondiente con el fin de instalar y configurar el producto.

Se programa la instalación del software para el día viernes 4 de noviembre a las 11:30hs, estimando un total de 4 horas para su la secuencia de pruebas y finalización.

En caso de tiempo de actualización se estipula que el periodo de interrupción del sistema requerirá una semana (6 días).

c) Documentación, comunicación y formación

Con el producto instalado en el servidor el Administrador podrá acceder a la interfaz principal (a partir de una IP que posteriormente será un vínculo hacia el entorno web) donde podrá acceder al sistema con un usuario y contraseña habilitado por los PM, y posteriormente (según el recorrido de la interfaz) realizar cualquier gestión requerida ya mencionada en el primer apartado llamado "deberes y responsabilidades del personal clave".

El administrador contará con un manual de usuario proporcionado por los PM para consultar cualquier funcionalidad puntual.

C. Proceso de Gestión de la Calidad

Este proceso se centra en el control de la calidad del producto software y su manutención.

a) Plan de garantía de calidad del software.

Este plan se centra en la descripción de las actividades del equipo para mantener y mejorar la calidad del software, así como documentar las métricas con las cuales evaluar las mismas.

La garantía de la misma está dada por las revisiones y correcciones continuas a los documentos, así como su elaboración a tiempo para facilitar el desarrollo del proyecto. Las pruebas y el proceso de verificación y validación contribuyen al desarrollo de la calidad del software debido a su constante refinamiento y reducción de errores.

El proyecto usará el patrón de diseño MVC, que consta de la división del programa en Modelo, donde están establecidos los datos que usa el sistema, la Vista, que es el aspecto visual del mismo (la interfaz de usuario) y el Controlador, el cual realiza toda la lógica del sistema y enlaza el Modelo y la Vista, creando un sistema integro.

Las métricas de calidad del software son:

- Cantidad de errores/deficiencias: los errores de código, diseño, etc.
- Amigabilidad con el usuario: Si el sistema provee información importante y evita la confusión del usuario.
- Facilidad de uso: que el sistema sea fácil de usar

IV. PROCESOS DE DESARROLLO

A. Proceso de Requisitos

a) Documento de Requisitos Software

1) Propósito del documento

Este documento recoge la especificación de requisitos del proyecto. Incluye en detalle cada función que debe incluir el sistema para satisfacer las necesidades del cliente en cuanto al dominio del problema.

2) Que hace y que no hace el producto

El producto consta del ingreso de estudiantes de diversas carreras a las materias correspondientes aprobado por los docentes de estas para evaluar su adquisición de conocimientos en las correspondientes cursadas. A su vez el producto permitirá a los docentes ingresar con sus datos y crear nuevos exámenes para estas materias previamente mencionadas. Este ambiente es exclusivo para evaluación y no contendrá los apuntes ni los programas de las materias.

3) Aplicaciones del software: beneficios, objetivos y metas

El objetivo es lograr que el alumno evalúe los conceptos y conocimientos adquiridos en clase para mejorar su rendimiento en la cursada, así como también refinar sus conocimientos a la hora de prepararse para un examen.

Para el lado del docente, permite analizar el rendimiento de los alumnos conforme a los contenidos de la materia que se evalúan y el programa de las mismas, para lograr conseguir una cátedra de mayor calidad.

4) Referencias

[IEEE93] IEEE Standard (Std.) 830-1993 (Revision of IEEE Std. 830-1984). Software Engineering Standards Committee of the IEEE Computer Society.

5) Organización del documento

Estas especificaciones de requisitos software siguen las recomendaciones del estándar [IEEE93] en cuanto a la organización de los distintos apartados así como el contenido de cada uno de ellos.

6) Independencia del producto

El Ambiente de AutoEvaluación es un producto independiente.

7) Interfaces del sistema

El sistema debe interactuar correctamente con el servidor Apache 2 y la base de datos desarrollada en MySQL 5.5, el sistema operativo especificado es Debian Jessie.

8) Limitaciones de memoria

No se establecen limitaciones en cuanto a la cantidad de memoria, tanto secundaria como primaria que el ambiente deba usar

9) Usuarios del sistema y sus privilegios

Hay 3 tipos de usuarios con sus correspondientes privilegios:

Administrador: Gestiona las cuentas de los docentes ,las materias y las carreras y además acepta los docentes que se registraron en el sistema

Docente: Gestiona los exámenes de la materia, así como unidades y aceptar las inscripciones de alumnos a la materia

Alumno: Realiza los exámenes creados previamente por el/los docente/s en el ambiente, para llevar a cabo esta operación debe estar previamente inscripto en la materia

El modo de operación es interactivo con la interfaz de usuario de la página, el resto de las operaciones que no sean interactivas se realizarán automáticamente por la página según la función deseada por el usuario

10) Características y conocimientos de los usuarios

La aplicación va dirigida a docentes y alumnos en el ámbito académico. Por ende, se encamina a un usuario con conocimientos sobre la realización de evaluaciones para evaluar conceptos específicos.

11) Limitaciones generales

El sistema solo debe tener instancias de evaluación, no soportará subida de apuntes ni comunicación entre alumnos y/o docentes. Sin embargo, el docente podrá exportar el examen deseado

El sistema será desarrollado en el framework Django (versión 1.7.7), de Python (versión 2.7), usará MySQL 5.5 para la persistencia de datos y el servidor usado será Apache 2.

12) Supuestos y dependencias

El acceso de los usuarios a las funciones del sistema depende de cada privilegio, ya que los docentes tienen un registro propio y son aceptados en el sistema por el administrador y los alumnos tienen un registro propio que les dará el ambiente.

13) Requisitos futuros

Estadísticas de las notas de los alumnos por examen: para lograr comprender cuales conceptos son los más difíciles de absorber para los alumnos.

Estadísticas de participación de alumnos en exámenes: para ver cómo es la participación de los alumnos en el ambiente y su interés por la refinación de conocimientos adquiridos en la materia.

Estadísticas de notas por materia: para saber qué materia es la más difícil de comprender para los alumnos.

14) *Requisitos funcionales*

Requisito funcional 1: Gestionar Universidades (Actor: Administrador)

Introducción: El administrador de la página realizara el alta, baja y modificación de las universidades.

Datos Entrada: Las entradas de datos serán las siguientes si se realiza el alta:

- Identificador de universidad
- Nombre de la universidad
- Dominio de los mails pertenecientes a esa universidad

En caso de modificación, la entrada será ya sea el nombre de la universidad o su dominio de mail, dependiendo de los atributos que se quieran modificar.

En caso de baja:

- Universidad registrada

Procesamiento: El sistema realizara las siguientes acciones luego de las entradas en caso de que se desee agregar una universidad:

- Registro en la base de datos de la universidad

En caso de modificar la universidad:

- Actualización en la base de datos

En caso de eliminarse la universidad:

- Actualización de la base de datos

Datos salida: El sistema brindará la siguiente información al ingresar, modificar o eliminar la universidad:

- Universidad registrada en el sistema (Alta).
- Universidad modificada (Modificación).
- Universidad y sus dependencias (Carreras, materias, exámenes, unidades, etc.) eliminadas del sistema.

Requisito funcional 2: Gestionar Carreras (Actor: Administrador)

Introducción: El administrador de la página realizara el alta, baja y modificación de las carreras del ámbito académico universitario.

Datos Entrada: Las entradas de datos serán las siguientes si se realiza el alta:

- Identificador de carrera

- Nombre de la carrera
- Cantidad de años

En caso de modificación, la entrada será ya sea el nombre de la carrera, o la cantidad de años, dependiendo de qué atributo se desee modificar. En caso de baja:

- Carrera registrada

Procesamiento: El sistema realizara las siguientes acciones luego de las entradas en caso de que se desee agregar una carrera:

- Registro en la base de datos de la carrera

En caso de modificar la carrera:

- Actualización en la base de datos

En caso de eliminarse la carrera:

- Actualización de la base de datos

Datos salida: El sistema brindará la siguiente información al ingresar, modificar o eliminar la carrera:

- Carrera registrada en el sistema (Alta).
- Carrera modificada (Modificación).
- Carrera y sus dependencias (Materias, exámenes, unidades, etc.) eliminadas del sistema.

Requisito funcional 3: Aceptar docentes (Actor: Administrador)

Introducción: El administrador de la página evaluará los datos ingresados por el supuesto docente y en base a su evaluación aceptará o rechazará la inscripción del docente en la página. Una vez aceptado, el docente podrá ingresar al ambiente y realizar todas las funciones del mismo.

Datos Entrada: En sí, un dato de entrada son los datos que el docente ingreso en el formulario de registro, luego de la revisión de los datos, y el identificador asignado automáticamente por la base de datos (ver datos de entrada del requisito funcional 4), el administrador aceptará o rechazará el docente junto con su registro.

Procesamiento: Si se acepta, se procede a actualizar la base de datos, cambiando el atributo "Habilitado" del registro donde se encuentre el docente en cuestión, una vez que el docente quiera ingresar al sistema con su usuario y contraseña, se realizara una consulta a través de estos mismos datos para verificar si en efecto ha sido aceptado en el administrador o no. Si se rechaza, se le enviará un e-mail avisando que fue rechazado del mismo.

Datos salida: Los datos de salida son:

- Docente habilitado para usar las funciones del sistema acordes a su privilegio (si se acepta)
- Docente eliminado (si se rechaza)

Requisito funcional 4: Gestionar materias (Actor: Administrador)

Introducción: El administrador agrega, modifica o elimina una materia, ingresando los datos de esta en un formulario y asignándole sus docentes. Si la materia se elimina, los exámenes dentro de la misma también serán eliminados

Datos Entrada: Los datos para agregar una materia son:

- Identificador de materia
- Nombre
- Año de carrera
- Profesor/es de la materia

Para modificarla el dato de entrada es la materia a modificar.

Para eliminar una materia, la entrada de datos será la materia misma

Procesamiento: En el caso del alta, el sistema agregará la materia a la base de datos.

En el caso de la modificación, se actualizarán la base de datos con los valores ingresados

En el caso de la baja, se eliminará de la base de datos la materia indicada con sus dependencias (exámenes, unidades, etc.)

Datos salida:

- Materia agregada en el sistema (Alta).
- Materia modificada (Modificación).
- Materia y dependencias (exámenes, unidades, etc.) eliminados y asociación con profesor/es eliminada/s (Baja).

Requisito funcional 5: Registrarse (Actor: Docente)

Introducción: el docente a través de un formulario ingresará sus datos para poder acceder a las funciones que le brinda el sistema, sin embargo, una vez se registre se enviara una solicitud de registro al administrador para que este vea los datos del docente y lo acepte o rechace en el sistema (ver Requisito funcional 2)

Datos Entrada: Para registrarse el docente usará los siguientes datos:

- Nombre
- Apellido
- DNI
- E-mail (debe ser de dominio académico)
- Usuario
- Contraseña
- Nombre de la Universidad
- Nombre de la Carrera

Además, se usara un identificador único asignado automáticamente por la base de datos a cada docente

Procesamiento: Una vez que se ingresan los datos del usuario y se confirma, se procederá a registrar sus datos en la

base de datos, de modo que este registrado como no habilitado.

Datos salida: El dato de salida es el docente registrado en la base de datos (pero no habilitado para usar el sistema, ver Requisito funcional 2)

Requisito funcional 6: Gestionar examen (Actor: Docente)

Introducción: En este caso de uso, el docente realizara el alta, baja o modificación de un examen, en el alta, se elegirán preguntas al azar de una unidad y tipo de preguntas especificadas. El docente indica la cantidad máxima de preguntas que aparecerán en pantalla y el límite de tiempo establecido.

Datos Entrada: Para realizar el alta de un examen, el docente deberá ingresar:

- La cantidad de preguntas totales del examen
- Los tipos de preguntas
- La cantidad de preguntas por cada tipo de pregunta (ejemplo: opción múltiple con 4 preguntas, unión con 8 preguntas, etc.)
- La duración del examen
- La puntuación del examen en base a las preguntas seleccionadas
- La descripción del examen (si se desea)
- La gestión de opciones del examen a crear

Por último, a cada examen se le asignara automáticamente un identificador único por medio de la base de datos

Para la modificación: el dato de entrada ser al examen mismo y un atributo del mismo. El examen en si puede marcarse como no visible para los alumnos, una vez se ha agregado a la base de datos y se publica.

Para la baja: el dato de entrada será el examen a eliminar

Procesamiento: En el caso del alta, se eligen al azar preguntas de las unidades seleccionadas que contendrá el examen, se calculara la puntuación total en base a cuánto vale cada pregunta, por último, se agrega el examen con sus preguntas a la base de datos.

En el caso de la modificación: actualizar la base de datos según si se modifican preguntas.

En el caso de la baja: Eliminar de la base de datos el examen indicado con sus dependencias

Datos salida: Para la adición, modificación o eliminación de exámenes del sistema se encuentran las siguientes salidas:

- Examen dado agregado (Alta)
- Examen modificado (Modificación)
- Examen eliminado con sus dependencias (opciones de las preguntas) (Baja).

Requisito funcional 7: Aceptar Alumnos (Actor: Docente)

Introducción: El docente tiene a su disposición la lista de alumnos que desean ingresar a la materia para realizar los exámenes, es decisión del usuario si rechaza o no a los solicitantes.

Datos Entrada:

- La lista de los alumnos que solicitan sus datos (podrá ver en detalle los datos de ellos)
- La materia a la que desean inscribirse

Procesamiento: El docente tiene 2 caminos, aceptar o rechazar la solicitud de ingreso. Si se acepta, se actualiza la base de datos cambiando el atributo “Habilitado” en la relación entre el alumno y la materia, se lo elimina de la lista de alumnos solicitantes. Si se rechaza, se actualiza la base de datos eliminándolo de la relación entre el alumno y la materia.

En ambos casos, se notifica al alumno.

Datos salida: Los datos de salida serán:

- Alumno registrado en la materia, posibilitándole hacer exámenes de la misma (Aceptar).
- Alumno rechazado y sin acceso a los exámenes de la materia (Rechazar).

Requisito funcional 8: Eliminar Alumno/s de la materia (Actor: Docente)

Introducción: el docente, puede eliminar un alumno de la lista de alumnos de la materia, prohibiéndole el acceso a la misma desde ese momento, puede también eliminar a todos en caso de que lo desee.

Datos entrada: los datos de entrada son:

- El/Los alumno/s a eliminar
- La materia del docente

Procesamiento: una vez que se seleccione al alumno, la base de datos eliminara la relación entre el alumno y la materia.

Datos de salida: los datos de salida son:

- Alumno/s eliminado/s de la materia

Requisito funcional 9: Gestionar preguntas (Actor: Docente)

Introducción: El docente, al hacer la gestión de un examen, realiza la gestión tanto de preguntas como de opciones (ver requisito funcional 9).

Datos Entrada: los datos de entrada para el alta son:

- Identificador de pregunta
- Enunciado de la pregunta.
- Qué tipo de pregunta es

- Valor de la pregunta en caso de ser respondida correctamente
- Cantidad de opciones

Si el tipo de pregunta es Verdadero o Falso, se usará un desplegable en las opciones.

Para la modificación el dato de entrada es la pregunta a modificar. Para la baja es la pregunta a eliminar.

Procesamiento: para el alta, una vez que se agregaron los datos requeridos, se procede a hacer la gestión de opciones para esa pregunta, una vez que se elige la cantidad de opciones debe tener en cuenta que la primera opción a agregar será siempre la correcta (luego se le asignara un identificador aleatorio dentro de los posibles). Para la modificación, se actualizan los datos de la pregunta en la base de datos. Finalmente, para la baja, se elimina la pregunta de la base de datos junto con sus opciones.

Datos salida: los datos de salida son:

- Pregunta (Alta y modificación).
- Pregunta eliminada (Baja).

Requisito funcional 10: Gestionar opciones (Actor: Docente)

Introducción: El docente, al hacer la gestión de un examen, realiza la gestión tanto de pregunta (ver requisito funcional 8) como de opciones. En este requisito se tratará la gestión de opciones para una pregunta

Datos Entrada: los datos de entrada para el alta son:

- Identificador de la opción
- Texto de la opción.
- Si es la opción correcta o no
- Otra opción (en caso de que el tipo de pregunta sea unión)
- Imagen (si la opción pertenece a la pregunta que es del tipo unión con imágenes)
- Descripción de la opción (si la pregunta es de tipo unión con imágenes solamente, este atributo es una descripción de la imagen y es opcional).

Recordar, como ya se mencionó en el requisito funcional anterior, que la primera opción que agregue el docente será la respuesta correcta a la pregunta, luego, en la base de datos se procederá a cambiar el orden de las preguntas

Para la modificación el dato de entrada es la opción a modificar. Para la baja es la opción a eliminar.

Procesamiento: para el alta, una vez que se agregaron los datos requeridos y se finaliza la gestión de examen se la agrega a la base de datos junto con el examen y las demás preguntas y opciones. Para la modificación, se actualizan los datos de la opción en la base de datos. Finalmente, para la baja, se elimina la opción de la base de datos.

Datos salida: los datos de salida son:

- Opción (Alta y modificación).
- Opción eliminada (Baja).

Requisito funcional 11: Exportar examen (Actor: Docente)

Introducción: El docente puede exportar el examen previamente creado a un archivo con formato reutilizable, para usarlo luego en otro entorno.

Datos Entrada: Las entradas de datos son:

- El examen seleccionado
- La opción de exportación

Procesamiento: Se copiará el contenido del examen al tipo de archivo elegido y será descargado en la pc del usuario.

Datos salida: El archivo descargado será la salida del requisito.

Requisito funcional 12: Gestionar unidades (Actor: Docente)

Introducción: El docente creará, modificará o eliminará unidades en las cuales alojará preguntas para los exámenes a crear.

Datos Entrada: el requisito tendrá los siguientes datos de entrada en caso de alta:

- Identificador de la unidad
- Nombre de la unidad
- Materia a la que pertenece

Para la modificación, el nombre de la unidad es el dato de entrada ya que es el único atributo modificable del mismo.

Para la baja, el dato de entrada será la unidad seleccionada a borrar.

Procesamiento: Una vez se ingresaron los datos para el alta, se procede a crear un registro en la base de datos de la nueva unidad que el docente agrego. Para la modificación, se actualiza el campo "Nombre" en la tabla "Unidad" en la base de datos. Para la baja de una unidad, simplemente se elimina el registro en la base de datos de la unidad seleccionada.

Datos salida: los datos de salida para el alta es la unidad creada. Para la modificación es la unidad con su atributo actualizado. Y para la baja es la unidad eliminada de la base de datos

Requisito funcional 13: Registrarse (Actor: Alumno).

Introducción: El alumno registrará sus datos en el sistema completando un simple formulario de registro que le brindara el ambiente.

Datos Entrada: Los datos de entrada serian:

- Nombres
- Apellidos
- Mail
- DNI
- Usuario
- Contraseña

Una vez se hayan ingresado estos datos, la base de datos asigna automáticamente un identificador único al alumno registrado

Procesamiento: Una vez se hayan ingresado los datos, se registrarán en la base de datos.

Datos salida: El único dato de salida del requisito es el alumno registrado en la base de datos.

Requisito funcional 14: Inscribir a materia (Actor: Alumno)

Introducción: El alumno seleccionara una de las materias de la carrera para enviarle una solicitud de ingreso a los/el docente/s de las mismas que ellos aceptaran o rechazaran (ver Requisito funcional 6)

Datos Entrada: Los datos de entrada serán:

- Alumno
- Materia

Procesamiento: Se actualizara la base de datos asociando al alumno con la materia y agregando un atributo en la relación misma que indique si el alumno está habilitado o no, para que este habilitado, uno de los docentes de la materia debe aceptarlo (ver requisito funcional 6).

Datos salida: Las salidas correspondientes al requisito son:

- Registro de los datos de la solicitud en la base de datos
- Alumno agregado a la lista de solicitudes pendientes de la materia

Requisito funcional 15: Realizar examen (Actor: Alumno)

Introducción: El Alumno, una vez aceptada la solicitud en la materia, elige uno de los exámenes realizados por los docentes y comienza su auto evaluación en el sistema, al final de la misma se le indicara su nota y cantidad de preguntas correctas e incorrectas

Datos Entrada: Los datos de entrada serán:

- El alumno que realizará el examen
- El examen propiamente dicho
- Las respuestas del alumno al examen

Procesamiento: Se realizaran los siguientes procesos para satisfacer el requisito:

- El sistema corrige el examen e indica los aciertos y errores del alumno, su nota final y si aprobó o no
- El sistema registra la nota obtenida y actualiza la cantidad de intentos de ese alumno al hacer el examen
- Una vez que se termina, el sistema cambiara el color del link al examen para indicarle al alumno que ya ha realizado al menos una vez ese examen

Datos salida: Las salidas correspondientes al requisito son:

- Examen realizado
- Resultados del examen (nota, condición de aprobación, aciertos y errores)
- Intentos del alumno en ese examen

15) *Requisitos de interfaz de usuario*

General:

La pantalla de ingreso al sistema tendrá un saludo de bienvenida y los campos para ingresar (usuario y clave) y la opción de registrarse).

El registro contendrá un formulario para datos de ingreso al sistema (usuario y clave), así como una opción donde el usuario indicará si es alumno o docente y completará los datos correspondientes al tipo de usuario elegido.

Administrador: Comprenderá las siguientes interfaces:

- Gestión de carreras, donde se realiza el alta, baja o modificación de las mismas
- Docentes pendientes, donde aceptará o rechazará los docentes que se hayan registrado en el sistema
- Gestión de materias, donde se realiza el alta, baja o modificación de las mismas

Docente: Comprenderá las siguientes interfaces:

- Registro al sistema, donde el usuario deberá ingresar sus datos
- El inicio del docente contendrá la lista de exámenes pendientes por publicar, a su vez como la lista de materias por universidad, por último, tendrá también una previa vista a la lista de solicitudes de alumnos. Los exámenes publicados pueden ser marcados como no visibles.
- Consulta y aceptación de solicitudes, donde se ve la lista de alumnos en espera de la aceptación y se decide si aceptarlas o no, se pueden aceptar todos los alumnos o rechazar a todos
- Expulsión de alumnos de la materia, donde se ve la lista de alumnos en la materia y se decide eliminarlos o no
- Gestión de exámenes, donde se realiza el alta, baja o modificación de los mismos. La modificación y alta son la misma interfaz de usuario. Dentro del alta puede aparecer una opción para guardar el examen como borrador. Los exámenes publicados, no publicados y borradores se ven dentro de esta interfaz

Alumno: Comprenderá las siguientes interfaces:

- Registro al sistema, donde el usuario deberá ingresar sus datos

- El inicio del alumno contendrá las universidades y carreras que se han registrado en el sistema con sus respectivas materias, contendrá una tabla donde se indiquen las materias a las que ya está inscripto y las materias a las que se inscribió pero que aún no lo aceptaron para acceder a las mismas, por último, se verá también los exámenes de las materias a las que ya esté inscripto, apareciendo de distinto color los nombres de los exámenes que ya ha realizado previamente.
- Búsqueda de exámenes por materia, donde aparecen los exámenes según la materia ingresada en el buscador
- Interfaz de autoevaluación, donde aparecerá el examen con sus respectivas preguntas, las preguntas aparecerán todas en un mismo formulario, los tipos de preguntas que aparecen son los definidos por el profesor
- Recuperación de contraseña, donde según el método elegido por el equipo, se realizará un procedimiento para informarle al alumno la contraseña nueva

16) *Interfaces de hardware*

Ninguna especificada.

17) *Interfaces de software*

Ninguna especificada.

18) *Comunicaciones o notificaciones*

Se usará el Email para notificaciones del sistema

19) *Requisitos de ejecución*

Tener instalado el servidor Apache 2 y la base de datos MySQL versión 5.5 en la máquina que administre el servidor

20) *Requisitos no funcionales*

Requisitos no funcionales del requisito funcional 14 (Realizar examen): el único requisito no funcional es el cual se especifica que el examen tiene un límite de tiempo de realización establecido por el docente.

Requisitos no funcionales del requisito funcional 8 (Gestionar opciones):

- La primera opción a agregar, como se ha dicho anteriormente, debe ser la correcta.
- Las opciones del tipo de pregunta Verdadero o Falso deben mostrarse como un desplegable en la página.

Requisitos no funcionales del requisito funcional Gestionar exámenes: El docente indica el límite de preguntas a agregar.

Requisitos no funcionales del requisito funcional Aceptar alumnos: se puede aceptar a todos o rechazar a todos los alumnos seleccionando la opción correspondiente

Requisitos no funcionales del requisito funcional Expulsar alumnos: se puede expulsar a todos los alumnos seleccionando la opción correspondiente.

Requisitos no funcionales del requisito funcional (Registrarse, Alumno y Docente): el registro no puede hacerse más de una vez con el mismo DNI.

Requisitos no funcionales del requisito funcional Gestionar exámenes: el docente puede indicar si un examen publicado es visible o no para los alumnos.

21) Acatamiento de estándares

Especificación de Requisitos Software: IEEE Std. 830/1983

Proyecto: IEEE Std. 1074/1989

B. Proceso de Diseño

a) Descripción de la arquitectura del sistema

Al ser un sistema web, la arquitectura del mismo será cliente-servidor, en la cual las tareas se distribuyen entre los proveedores de servicios (servidor) y los consumidores (cliente). En esta arquitectura el cliente realiza una petición al programa del servidor y este le envía una respuesta. Un ejemplo de cliente-servidor es el correo electrónico.

1) Ventajas

- Centralización del control: los recursos y datos son controlados por el servidor de forma que un cliente defectuoso no afecte al sistema
- Escalabilidad: la capacidad de los clientes y servidores se pueden amplificar por separado y se pueden añadir nuevos nodos a la red ya sean clientes o servidores
- Fácil de mantener: al estar distribuidas las funciones y operaciones en ordenadores independientes es fácil reemplazar, arreglar, actualizar o trasladar un servidor mientras que los clientes no se verán afectados o son afectados mínimamente por estos cambios.
- Amplio rango de tecnologías: hay varias herramientas para la arquitectura que aseguran la seguridad de las

transacciones, la amigabilidad de la interfaz y la facilidad de empleo.

2) Desventajas

- Problema con congestiones de tráfico: el servidor puede tener problemas al atender peticiones simultáneas de varios clientes
- No es tan robusto como una red P2P (redes de pares o peer-to-peer, donde cada nodo tiene su propio servidor y los recursos están distribuidos): si un servidor de la arquitectura cliente-servidor cae, no pueden ser satisfechas ninguna de las peticiones de los clientes, en cambio, en una red P2P, debido a la distribución de los recursos en varios nodos de red, es posible completar las peticiones.
- El software y hardware de un servidor son determinantes: Para servir de servidor, se necesita hardware y software específico para atender una cierta cantidad de clientes, esto genera un coste mayor a la hora de armar la arquitectura

a) Descripción del diseño del software

1) Casos de uso

Los casos de uso describen las funciones del sistema del punto de vista del negocio y que actores realizan dichas acciones (ver figuras 9, 10 y 11).

2) Diagrama Entidad-Relación

El diagrama entidad-relación es un diagrama donde se especifican las entidades del sistema y sus conjuntos de atributos, sirve para ver los datos del punto de vista del negocio, además de las relaciones entre estas, sirve para diseñar la base de datos del sistema (ver Figura 12).

3) Diagrama de Clases

El diagrama de clases muestra las clases que se usaran en la codificación, y sus atributos comprendidos, también muestra las operaciones que realiza cada acción dentro del sistema, es una visión interna a los datos del mismo (ver Figura 13).

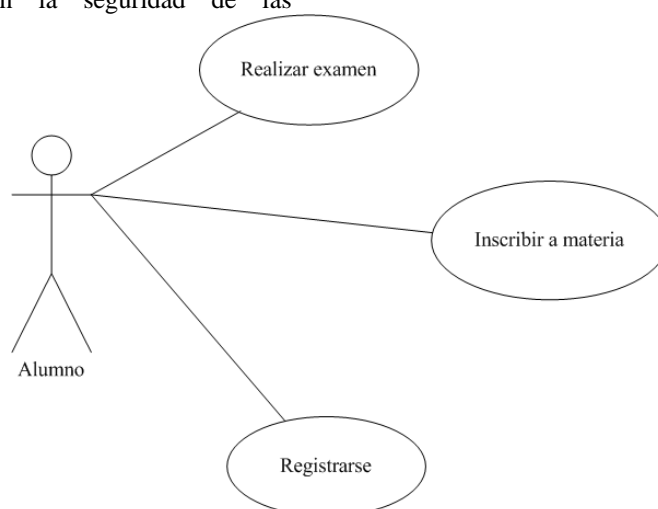


Figura 9. Casos de uso del alumno

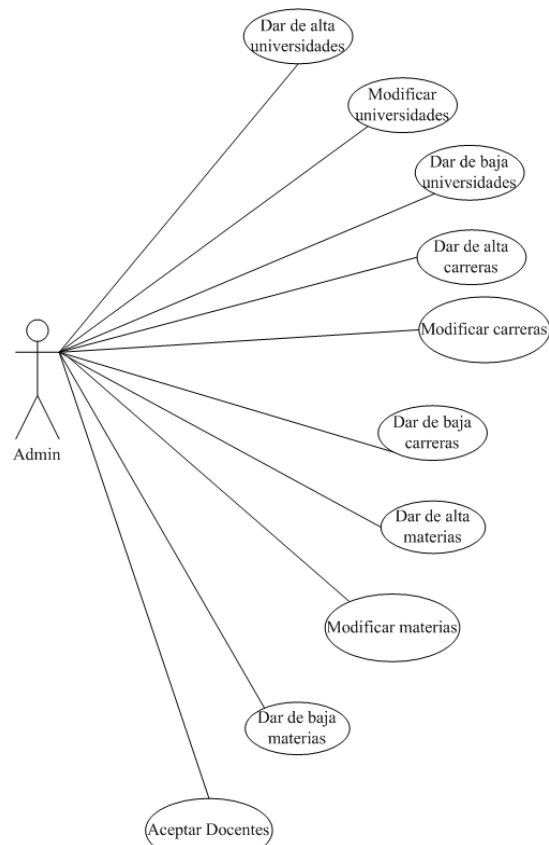


Figura 10. Casos de uso del administrador

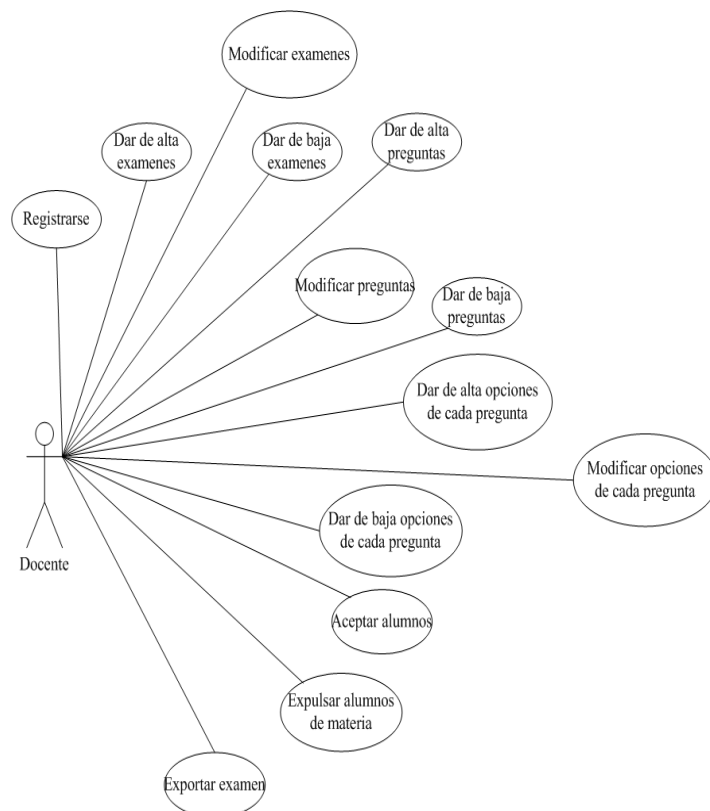


Figura 11. Casos de uso del Docente

Nota: en los casos de uso, aquellos que contengan alta, baja o modificación son vistos como un requisito funcional solo,

por ejemplo, agregar, modificar o eliminar un examen se ven reflejados en el requisito funcional Gestionar exámenes

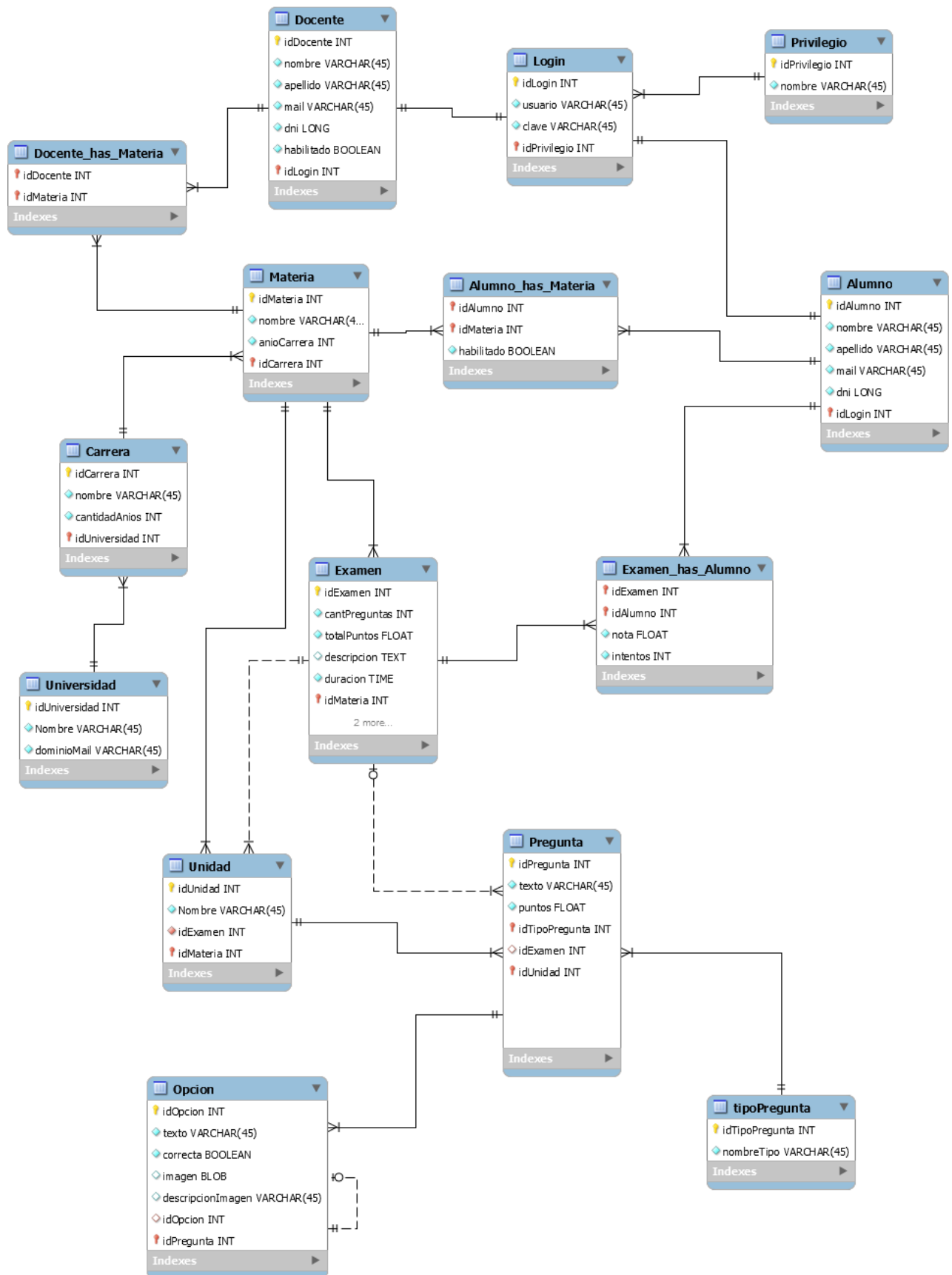


Figura 12. Diagrama Entidad-Relación (DER)

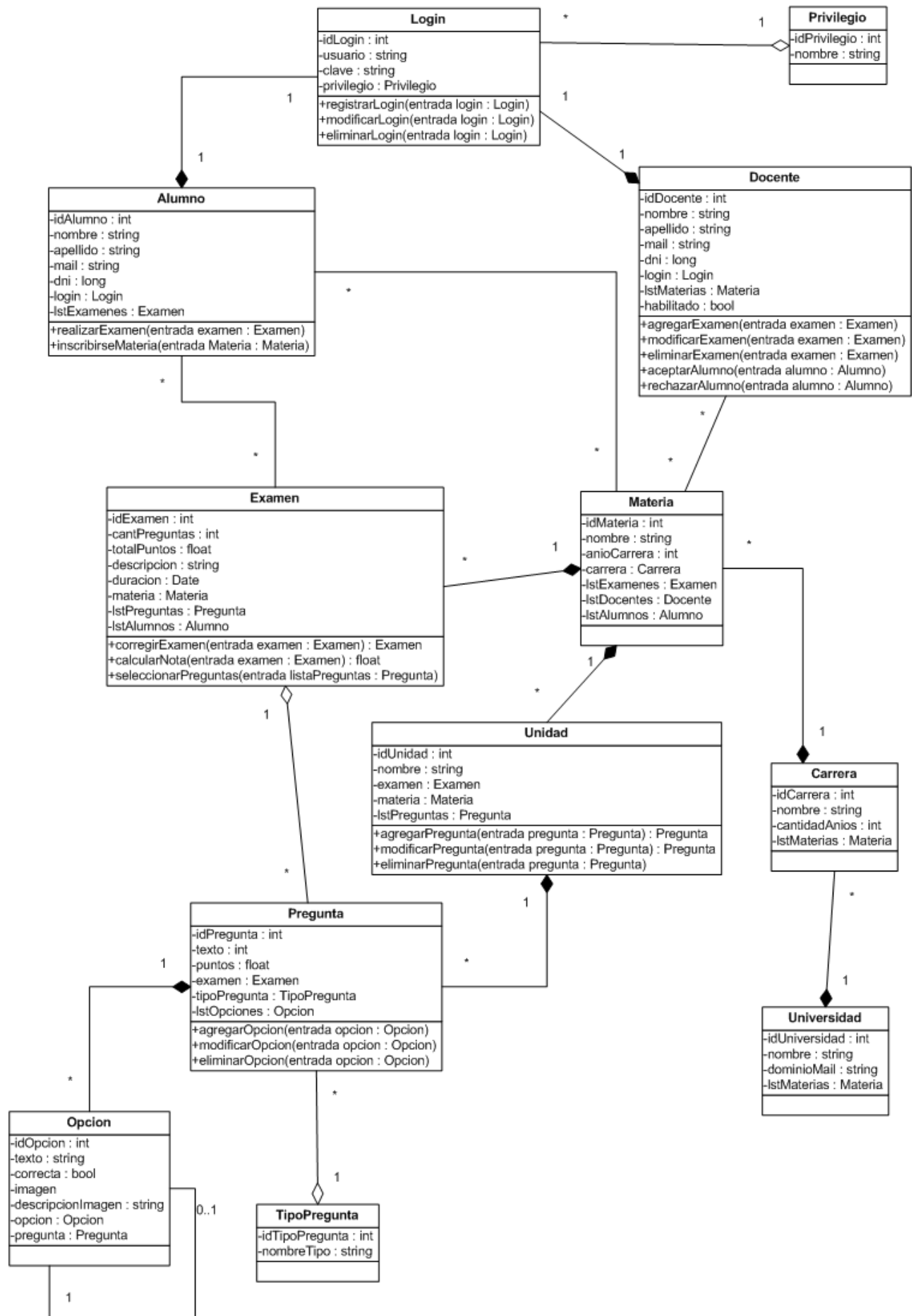


Figura 13. Diagrama de clases

4) Escenarios de Caso de Uso

Los escenarios de caso de uso son una descripción detallada de casos de uso determinados, estos contienen los pasos que realiza cada paso y la información de cada uno de estos.

Administrador:

Nombre del caso de uso:	Aceptar docentes (Aprobar)	ID: 001
Área:	Administración de la página	
Actor(es):	Administrador	
Descripción:	El administrador acepta los docentes que se registraron en la pagina para desempeñar sus funciones	
Activar evento:	El administrador accede a la vista de docentes pendientes y elige la opción de “Aceptar docente” o “Aceptar todos”	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. Se procede a habilitar el docente en la base de datos	Docente	
2. Se elimina al docente de la lista de docentes pendientes	Docente	
Precondiciones:	Docente no debe haber sido aceptado previamente, debe haber sido registrado en el sistema previamente	
Postcondiciones:	Docente con funciones e ingreso habilitados al sistema	
Suposiciones:	Se supone que el administrador verificará el mail del docente para ver si coinciden con el dominio universitario	
Reunir requerimientos:	No aplica	
Aspectos sobresalientes:	No aplica	
Prioridad:	Alta	
Riesgo:	Bajo	

Nombre del caso de uso:	Aceptar docentes (Rechazar)	ID: 002
Área:	Administración de la página	

Actor(es):	Administrador
Descripción:	El administrador rechaza los docentes que se registraron en la página para desempeñar sus funciones
Activar evento:	El administrador accede a la vista de docentes pendientes y elige la opción de “Rechazar docente” o “Rechazar todos”
Tipo de señal:	Externa
Pasos desempeñados (ruta principal):	Información para los pasos
1. Se elimina al docente de la base de datos	Docente
2. Se elimina al docente de la lista de docentes pendientes	Docente
Precondiciones:	Docente no debe haber sido aceptado previamente, debe haber sido registrado en el sistema previamente
Postcondiciones:	Docente rechazado y su registro eliminado en la base de datos
Suposiciones:	Se supone que el administrador verificará el mail del docente para ver si coinciden con el dominio universitario
Reunir requerimientos:	No aplica
Aspectos sobresalientes:	No aplica
Prioridad:	Alta
Riesgo:	Bajo

Docente:

Nombre del caso de uso:	Registrarse	ID: 003
Área:	Sistema del docente	
Actor(es):	Docente	
Descripción:	El docente ingresa sus datos en un formulario para enviar la solicitud de ingreso a la página al administrador	
Activar evento:	El docente selecciona la opción de registrarse y confirma el envío de datos del formulario	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El usuario ingresa el usuario en el formulario	Nombre, apellido, DNI, mail, universidad, carrera	
2. El usuario es devuelto a la página de inicio del sistema	Ninguna	

Precondiciones:	Docente no debe haber sido registrado en el sistema previamente, ni debe haber un docente con el mismo DNI registrado
Postcondiciones:	Docente con solicitud enviada al administrador
Suposiciones:	Se supone que el docente se registra con un mail de dominio académico
Reunir requerimientos:	No aplica
Aspectos sobresalientes:	No aplica
Prioridad:	Alta
Riesgo:	Bajo

Nombre del caso de uso:	Aceptar alumnos (Aprobación de solicitud)	ID: 004
Área:	Sistema del docente	
Actor(es):	Docente	
Descripción:	El docente acepta (para el rechazo ver el escenario 005) a los alumnos que solicitan el ingreso a la materia	
Activar evento:	El docente accede a la pantalla que muestra el total de alumnos pendientes en la materia, y selecciona "Aceptar"	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El docente selecciona el alumno a aceptar	Alumno	
2. El docente acepta al alumno	Alumno	
3. El sistema habilita al alumno dentro de la materia, permitiéndole realizar los exámenes dentro de la misma	Alumno, Materia	
Precondiciones:	Alumno no debe haber sido aceptado previamente	
Postcondiciones:	Alumno aceptado y con los exámenes de la materia disponibles	
Suposiciones:	Se supone que el alumno ha enviado previamente la solicitud	
Reunir requerimientos:	No aplica	
Aspectos sobresalientes:	No aplica	
Prioridad:	Alta	
Riesgo:	Bajo	

Nombre del caso de uso:	Aceptar alumnos (Rechazo de solicitud)	ID: 005
Área:	Sistema del docente	
Actor(es):	Docente	
Descripción:	El docente rechaza (para la aprobación ver el escenario 004) a los alumnos que solicitan el ingreso a la materia	
Activar evento:	El docente accede a la pantalla que muestra el total de alumnos pendientes en la materia	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El docente selecciona el alumno a rechazar	Alumno	
2. El docente rechaza al alumno seleccionando la opción "Rechazar"	Alumno	
3. El alumno es eliminado de la lista de pendientes	Alumno	
Precondiciones:	Alumno no debe haber sido aceptado previamente	
Postcondiciones:	Alumno rechazado y sin acceso a los exámenes de la materia	
Suposiciones:	Se supone que el alumno ha enviado previamente la solicitud	
Reunir requerimientos:	No aplica	
Aspectos sobresalientes:	No aplica	
Prioridad:	Alta	
Riesgo:	Bajo	

Nombre del caso de uso:	Exportar examen	ID: 006
Área:	Sistema del docente	
Actor(es):	Docente	
Descripción:	Exporta el examen y sus preguntas a un formato de archivo que él desee dentro de los disponibles	
Activar evento:	El docente selecciona la opción "Exportar examen" dentro de la vista de exámenes	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El docente selecciona el examen a exportar	Examen	
2. El docente selecciona el	Examen, tipo de archivo	

tipo de archivo en el que desea formatear el examen	
3. El examen es formateado a ese archivo y descargado por el docente	Examen, tipo de archivo
Precondiciones:	El examen no ha sido convertido ni descargado por el docente
Postcondiciones:	Examen convertido y descargado por el docente
Suposiciones:	Ninguna
Reunir requerimientos:	No aplica
Aspectos sobresalientes:	No aplica
Prioridad:	Alta
Riesgo:	Bajo

Nombre del caso de uso:	Expulsar alumnos de la materia	ID: 007
Área:	Sistema del docente	
Actor(es):	Docente	
Descripción:	El docente expulsa a los alumnos que tienen acceso a la materia, puede expulsar a todos si se desea	
Activar evento:	El docente accede a la vista de alumnos inscriptos en la materia y selecciona alumnos para borrar	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El docente selecciona el/los alumno/s a eliminar	Alumno(s)	
2. El docente selecciona la opción correspondiente ("Eliminar" o "Eliminar todos")	Alumno(s)	
3. Los alumnos son expulsados de la materia y pierden el acceso a la mismas	Alumno(s), Materia	
Precondiciones:	Los alumnos a eliminar no fueron expulsados previamente	
Postcondiciones:	Alumno/s expulsado/s de la materia	
Suposiciones:	Ninguna	
Reunir requerimientos:	No aplica	
Aspectos sobresalientes:	No aplica	
Prioridad:	Alta	
Riesgo:	Bajo	

Alumno:

Nombre del caso de uso:	Registrarse	ID: 008
Área:	Sistema del alumno	
Actor(es):	Alumno	
Descripción:	El alumno ingresa sus datos en un formulario para registrarse en el sistema	
Activar evento:	El alumno selecciona la opción de registrarse y confirma el envío de datos del formulario	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El usuario ingresa los datos en el formulario	Nombre, apellido, DNI, mail, usuario, clave	
2. El usuario es devuelto a la página de inicio del sistema	Ninguna	
Precondiciones:	Alumno no debe haber sido registrado en el sistema previamente, ni debe haber un alumno con el mismo DNI registrado	
Postcondiciones:	Alumno registrado en la base de datos y con ingreso al sistema disponible	
Suposiciones:	Ninguna	
Reunir requerimientos:	No aplica	
Aspectos sobresalientes:	No aplica	
Prioridad:	Alta	
Riesgo:	Bajo	

Nombre del caso de uso:	Realizar examen	ID: 009
Área:	Sistema del alumno	
Actor(es):	Alumno	
Descripción:	El alumno selecciona un examen de las materias en las que fue aceptado y contesta sus preguntas, al final, obtiene el resultado del examen	
Activar evento:	El alumno selecciona el examen a realizar	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El alumno selecciona el examen deseado	Examen	
2. El alumno contesta las preguntas del examen	Ninguna	

3. El sistema calcula la puntuación del examen en base a lo que el alumno respondió	Examen, Pregunta
4. El sistema le informa al alumno los resultados del examen	Examen
Precondiciones:	Ninguna
Postcondiciones:	Examen realizado
Suposiciones:	Ninguna
Reunir requerimientos:	No aplica
Aspectos sobresalientes:	No aplica
Prioridad:	Alta
Riesgo:	Bajo

Nombre del caso de uso:	Inscribir a materia	ID: 010
Área:	Sistema del alumno	
Actor(es):	Alumno	
Descripción:	El alumno selecciona una materia de las disponibles para enviar la solicitud al docente de la misma	
Activar evento:	El alumno selecciona la materia a Inscribirse	
Tipo de señal:	Externa	
Pasos desempeñados (ruta principal):	Información para los pasos	
1. El alumno selecciona la materia deseada	Materia	
2. El alumno envía la solicitud de inscripción a los docentes de la materia	Materia, Alumno	
Precondiciones:	El alumno no debe haber enviado una solicitud que aún no ha sido aceptada o rechazada	
Postcondiciones:	Solicitud de inscripción enviada	
Suposiciones:	Ninguna	
Reunir requerimientos:	No aplica	
Aspectos sobresalientes:	No aplica	
Prioridad:	Alta	
Riesgo:	Bajo	

El proposito del proceso de Verificacion y Validacion es lograr localizar los errores que se encuentran dentro del diseño o codigo del proyecto, asi como lograr evaluar si la funcionalidad del mismo es la requerida.

Para llevar a cabo este proceso, se establecerá un Plan de Pruebas de Software para cumplir con el proposito establecido, ademas de revisiones periodicas del equipo en los documentos del proyecto, el Plan de Pruebas contendrá los siguientes casos de prueba:

- Todos los ABM
- La realizacion de un examen
- La correccion de un examen
- La exportación de un examen
- Aceptar/rechazar solicitudes de alumnos y docentes
- Expulsar alumnos de una material
- Enviar las solicitudes de inscripcion y registro.

V. PROCESOS DE VERIFICACIÓN Y VALIDACIÓN

A. Plan de verificación y validación

Se entiende por verificación a la comprobación de que el software este propiamente hecho (en cuestiones de código). La validación es la comprobación de que el producto software realiza las funciones previamente acordadas, es decir, hace lo que se quiere que haga.

ANEXO 1

I. MAQUETA DEL PROYECTO

Este anexo contiene la maqueta presentada al cliente para definir los requisitos de interfaz del proyecto, téngase en cuenta que la maqueta presentada no es la interfaz de usuario final, es usada para definir requisitos de interfaz del proyecto y la estructura de la interfaz propiamente dicha. Considere

también que es posible que haya cambios entre la maqueta y la interfaz de usuario final, o bien los requisitos funcionales y de interfaz del proyecto.

Pantalla de inicio: login:



The login screen features a green header with the text "AAEV 2.0" and "Ambiente de Auto Evaluación 2.0". Below the header, there is a section titled "Iniciá sesión:" with a user icon. It contains two input fields for "E-mail" and "Contraseña". Below these fields are two buttons: "RECUPERAR CONTRASEÑA" and "ENTRAR". At the bottom, there is a section titled "¿Sos nuevo en esto?" with a user icon and a button labeled "REGISTRARTE".

Pantalla de registro: alumno:



The registration screen features a green header with the text "AAEV 2.0" and "Ambiente de Auto Evaluación 2.0". Below the header, there is a section titled "Registro de usuario nuevo:" with a user icon. It contains several input fields: "Nombre", "Apellido", "DU/DNI", "E-mail", "Contraseña", and "Contraseña nuevamente". There are also two radio buttons for "Soy alumno" (selected) and "Soy docente". At the bottom right, there is a button labeled "REGISTRARME".

Pantalla de registro: docente:



AAEV 2.0
Ambiente de Auto Evaluación 2.0

 **Registro de usuario nuevo:**

Nombre

Apellido

DU/DNI

Soy alumno ☐ Soy docente ☒

E-mail institucional

Contraseña

Contraseña nuevamente

Facultad/Universidad donde trabajo

Departamento Universitario (opcional)

Carrera en la que trabajo

Materia que imparto

REGISTRARME

Pantalla de registro exitoso:



AAEV 2.0
Ambiente de Auto Evaluación 2.0

 **Registro de docente realizado con éxito!**

En los próximos días recibirá en su correo la confirmación de la cuenta por parte del Administrador, quien habrá vinculado su usuario con su materia correspondiente. A partir de ese momento podrá utilizar el sitio en su totalidad.



Pantalla de Inicio del alumno: Materias pendientes, activas y exámenes:

AAEV 2.0

Ambiente de Auto Evaluación 2.0

NAZARENO GALVAN

CURSOS ACTIVOS

Ninguno...

CURSOS PENDIENTES

Ninguno...

EXAMENES PUBLICADOS

EXAMENES PENDIENTES

Ninguno...

EXAMENES REALIZADOS

Ninguno...

OFERTAS ACADEMICAS

MATERIAS HABILITADAS

Universidad Nacional de Lanus

Desarrollo productivo y tecnológico

Lic. Sistemas

- Matemática 1 Vasquez
- Matemática 2 Diddio
- Ing. Software 1 Garcia Martinez
- Ing. Software 2 Amatriain
- Org. de Computadora Garcia

Humanidades y Artes

Lic. D. y C. Visual

- Diseño 1 Fraccia

Universidad Nacional de Lomas de Zamora

Ingeniería

General

Mecatrónica

- Mecánica de los fluidos Lopez

Pantalla de inicio del alumno, solicitudes de materia enviadas:

AAEV 2.0

Ambiente de Auto Evaluación 2.0

NAZARENO GALVAN

CURSOS ACTIVOS

Ninguno...

CURSOS PENDIENTES

Universidad Nacional de Lanus / Lic. Sistemas / Matemática 1
Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1
Universidad Nacional de Lomas de Zamora / Ingeniería / Mecán..

EXAMENES PUBLICADOS

EXAMENES PENDIENTES

Ninguno...

EXAMENES REALIZADOS

Ninguno...

OFERTAS ACADEMICAS

MATERIAS HABILITADAS

Universidad Nacional de Lanus

Desarrollo productivo y tecnológico

Lic. Sistemas

- Matemática 1 Vasquez
- Matemática 2 Diddio
- Ing. Software 1 Garcia Martinez
- Ing. Software 2 Amatriain
- Org. de Computadora Garcia

Humanidades y Artes

Lic. D. y C. Visual

- Diseño 1 Fraccia

Universidad Nacional de Lomas de Zamora

Ingeniería

Ing. General


Ing. Mecatrónica

- Mecánica de los fluidos Lopez

Inicio de alumno: aceptado en 2 materias, puede ver los exámenes de las mismas:


AAEV 2.0

Ambiente de Auto Evaluación 2.0

 **NAZARENO GALVAN**


CURSOS ACTIVOS
Universidad Nacional de Lanus / Lic. Sistemas / Matemática 1
Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1

CURSOS PENDIENTES
Universidad Nacional de Lomas de Zamora / Ingenieria/ Mecán..


 **EXAMENES PUBLICADOS**


EXAMENES PENDIENTES
Lic. Sistemas / Ing. Software 1 / GUIA 1 / 10 preguntas
Lic. Sistemas / Ing. Software 1 / GUIA 2 / 6 preguntas
Lic. Sistemas / Ing. Software 1 / GUIA 3 / 15 preguntas


EXAMENES REALIZADOS CON EXITO
Ninguno...

 **OFERTAS ACADEMICAS**


MATERIAS HABILITADAS


 **Universidad Nacional de Lanus**

 **Desarrollo productivo y tecnológico**


 Lic. Sistemas


- Matemática 1 Vasquez
- Matemática 2 Diddio
- Ing. Software 1 Garcia Martinez
- Ing. Software 2 Amatriain
- Org. de Computadora Garcia


 Humanidades y Artes


 Lic. D. y C. Visual

- Diseño 1 Fraccia

 **Universidad Nacional de Lomas de Zamora**

 **Ingeniería**

 Ing. General

 Ing. Mecatrónica

- Mecánica de los fluidos Lopez

Canosa No Alejandro A., Galvan Gabriel N., Di Stefano Agustín. L. . 2016. *Ambiente de AutoEvaluación (AAEV 2.0)*.
Cátedra Proyecto de Software 2016

36

Pantalla de realización de examen:

The screenshot displays the AAEV 2.0 exam environment with a green header. It shows six question cards arranged in a 3x2 grid. Each card has a green checkmark icon and the text 'Lic. Sistemas/ Ing. Software 1/ GUIA 2'. The questions are numbered 1 through 6. Questions 1, 3, and 5 are multiple-choice with radio buttons. Questions 2 and 4 are matching exercises with arrows and labels. Question 6 is a text-based question asking if the user has finished, with a button to submit results.

AAEV 2.0
Ambiente de Auto Evaluación 2.0

1. PREGUNTA UNO
Lic. Sistemas/ Ing. Software 1/ GUIA 2
☒ RESPUESTA UNO
☐ RESPUESTA DOS
☐ RESPUESTA TRES

2. PREGUNTA DOS
Lic. Sistemas/ Ing. Software 1/ GUIA 2
A. PUNTO UNO
B. RESPUESTA DOS
C. RESPUESTA TRES
A RESPUESTA UNO
A RESPUESTA DOS
A RESPUESTA TRES

3. PREGUNTA TRES
Lic. Sistemas/ Ing. Software 1/ GUIA 2
☒ RESPUESTA UNO
☐ RESPUESTA DOS
☐ RESPUESTA TRES
☐ RESPUESTA CUATRO
☐ RESPUESTA CINCO

4. PREGUNTA CUATRO
Lic. Sistemas/ Ing. Software 1/ GUIA 2
RESPUESTA UNO
RESPUESTA DOS
RESPUESTA TRES

5. PREGUNTA UNO
Lic. Sistemas/ Ing. Software 1/ GUIA 2
☒ RESPUESTA UNO
☐ RESPUESTA DOS
☐ RESPUESTA TRES

5. PREGUNTA SEIS
Lic. Sistemas/ Ing. Software 1/ GUIA 2
☒ RESPUESTA UNO
☐ RESPUESTA DOS

Si crees haber terminado puedes enviar tus 6 respuestas
ENVIAR RESULTADOS

Pantalla de resultado del examen realizado (nótese que estos 2 resultados van en vistas distintas):

The first screenshot shows a green header and a green checkmark icon with the text 'Examen superado!' and 'Tu rendimiento fue satisfactorio.' The second screenshot shows a green header and a yellow exclamation mark icon with the text 'Examen no superado!' and 'Tu rendimiento fue insuficiente. Intenta realizar este examen nuevamente.'

AAEV 2.0
Ambiente de Auto Evaluación 2.0
Examen superado!
Tu rendimiento fue satisfactorio.

AAEV 2.0
Ambiente de Auto Evaluación 2.0
Examen no superado!
Tu rendimiento fue insuficiente. Intenta realizar este examen nuevamente.

Vuelta a inicio del alumno, donde el examen aparece en la lista de exámenes realizados:

AAEV 2.0

Ambiente de Auto Evaluación 2.0

NAZARENO GALVAN

CURSOS ACTIVOS

Universidad Nacional de Lanus / Lic. Sistemas / Matemática 1
Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1

CURSOS PENDIENTES

Universidad Nacional de Lomas de Zamora / Ingeniería/ Mecán..

EXAMENES PUBLICADOS

EXAMENES PENDIENTES

Lic. Sistemas / Ing. Software 1 / GUIA 1 / 10 preguntas
Lic. Sistemas / Ing. Software 1 / GUIA 3 / 15 preguntas

EXAMENES REALIZADOS CON EXITO

Lic. Sistemas / Ing. Software 1 / GUIA 2 / 6 preguntas

OFERTAS ACADÉMICAS

MATERIAS HABILITADAS

Universidad Nacional de Lanus

Desarrollo productivo y tecnológico

Lic. Sistemas

Matemática 1 Vasquez
Matemática 2 Diddio
Ing. Software 1 García Martínez
Ing. Software 2 Amatriain
Org. de Computadora García

 Humanidades y Artes

Lic. D. y C. Visual

Diseño 1 Fraccia

Universidad Nacional de Lomas de Zamora

Ingeniería

Ing. General
 Ing. Mecatrónica

Mecánica de los fluidos Lopez

Inicio del docente, donde se ven exámenes publicados, sus materias, y los exámenes no publicados:

AAEV 2.0

Ambiente de Auto Evaluación 2.0

RAMON GARCIA MARTINEZ

ALUMNOS PENDIENTES

Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1

LISTA DE EXAMENES PENDIENTES

EXAMENES SIN PUBLICAR

Ninguno...

CREAR UN NUEVO EXAMEN

LISTA DE MATERIAS

MATERIAS ASIGNADAS

Universidad Nacional de Lanus

Desarrollo productivo y tecnológico

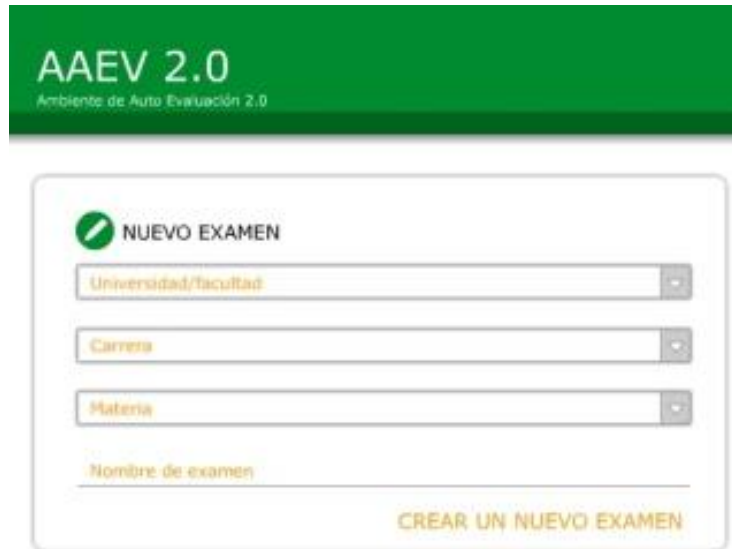
Lic. Sistemas

Ing. Software 1 García Martínez

GUIA 1

GUIA 2

Alta de exámen:



The screenshot shows the 'NUEVO EXAMEN' (New Exam) form in the AAEV 2.0 interface. It features a green header with the text 'AAEV 2.0' and 'Ambiente de Auto Evaluación 2.0'. The form itself is white with a green pencil icon and the title 'NUEVO EXAMEN'. It contains four input fields: 'Universidad/facultad', 'Carrera', 'Materia', and 'Nombre de examen'. At the bottom right of the form is a yellow button labeled 'CREAR UN NUEVO EXAMEN'.

Inicio del examen, con examen creado sin publicar, estos se pueden editar para agregar preguntas/opciones:



The screenshot displays the AAEV 2.0 dashboard with a green header. It is divided into two main sections. The left section, titled 'RAMON GARCIA MARTINEZ', contains two sub-sections: 'ALUMNOS PENDIENTES' showing 'Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1', and 'LISTA DE EXAMENES PENDIENTES' showing 'EXAMENES SIN PUBLICAR' with a list of 'Lic. Sistemas / Ing. Software 1 / GUIA 3 / 0 PREG' and a 'CREAR UN NUEVO EXAMEN' button. The right section, titled 'LISTA DE MATERIAS', shows 'MATERIAS ASIGNADAS' for 'Universidad Nacional de Lanus', including 'Desarrollo productivo y tecnológico' and 'Lic. Sistemas', with a detailed list of exams like 'Ing. Software 1' and 'GUIA 1' and 'GUIA 2'.

Alta o edición de examen (alta de preguntas I):

AAEV 2.0
Ambiente de Auto Evaluación 2.0

 Lic. Sistemas / Ing. Software 1 / GUIA 3 / 0 PREG

Nombre de examen

PREGUNTAS

Ninguna...

GUARDAR EDICIONAGREGAR PREGUNTA

Alta de preguntas II:

AAEV 2.0
Ambiente de Auto Evaluación 2.0

 AGREGAR PREGUNTA (1) - GUIA 3

Mult. Choice texto ☒ Mult. Choice imagen ☐ Unir con flechas ☐

Pregunta

Valor numérico de la respuesta

SIGUIENTE PASO

Alta de Opciones (cada panel es para un tipo de pregunta distinto, siendo el primero arriba a la izquierda un multiple choice, a la derecha una unión de conceptos, y abajo a la izquierda selección de imágenes):

AAEV 2.0
Ambiente de Auto Evaluación 2.0

AGREGAR PREGUNTA (1) - GUIA 3

Respuesta correcta

Respuesta incorrecta

2da respuesta incorrecta

GUARDAR RESPUESTAS

AGREGAR PREGUNTA (1) - GUIA 3

Pregunta A

Respuesta A

Pregunta B

Respuesta B

Pregunta C

Respuesta C

GUARDAR RESPUESTAS

AGREGAR PREGUNTA (1) - GUIA 3

Respuesta correcta

IMAGEN respuesta correcta

EXAMINAR

Respuesta incorrecta

IMAGEN respuesta incorrecta

EXAMINAR

2da respuesta incorrecta

IMAGEN 2da respuesta incorrecta

EXAMINAR

GUARDAR RESPUESTAS

Alta o edición de examen: vemos la pregunta agregada:

AAEV 2.0
Ambiente de Auto Evaluación 2.0

Lic. Sistemas / Ing. Software 1 / GUIA 3 / 1 PREG

Nombre de examen

PREGUNTAS

1. PREGUNTA UNO / 3 RESPUESTAS

GUARDAR EDICION

AGREGAR PREGUNTA

Edición de preguntas:

AAEV 2.0
Entorno de Auto Evaluación 2.0

 **EDITAR PREGUNTA (1) - GUIA 3 / 3 RESPUESTAS**

Pregunta

Valor numérico de la respuesta

AGREGAR RESPUESTA

GUARDAR EDICION

 **EDITAR RESPUESTA INCORRECTA (2)** 

Respuesta incorrecta

 **EDITAR RESPUESTA CORRECTA**

Respuesta correcta

 **EDITAR RESPUESTA INCORRECTA (1)** 

Respuesta incorrecta

Eliminación de una opción:

AAEV 2.0
Entorno de Auto Evaluación 2.0

 **EDITAR PREGUNTA (1) - GUIA 3 / 3 RESPUESTAS**

Pregunta

Valor numérico de la respuesta

AGREGAR RESPUESTA

GUARDAR EDICION

 **EDITAR RESPUESTA INCORRECTA (2)** 

Respuesta incorrecta

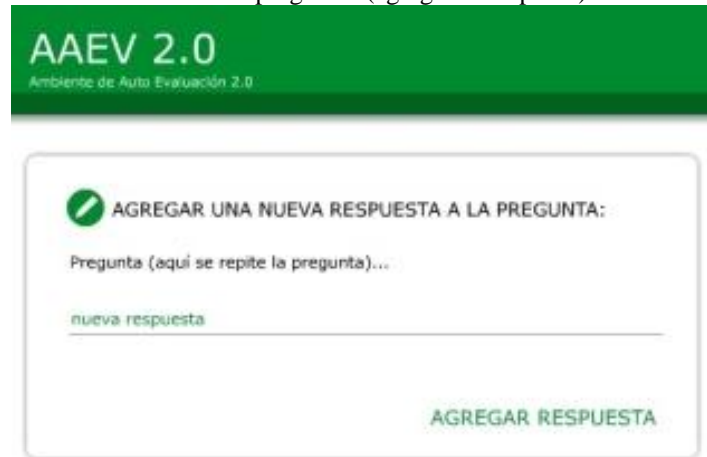
 **EDITAR RESPUESTA CORRECTA**

Respuesta correcta

 **EDITAR RESPUESTA INCORRECTA (1)** 

Respuesta incorrecta

Edición de preguntas (agregar una opción):



Edición de preguntas: opción agregada:



Edición o alta de exámenes: se ve el examen que se agrego al principio, con todas sus preguntas a la derecha:



Inicio del docente: donde se ven sus exámenes agregados:

AAEV 2.0
Ambiente de Auto Evaluación 2.0

RAMON GARCIA MARTINEZ

ALUMNOS PENDIENTES

Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1

LISTA DE EXAMENES PENDIENTES

EXAMENES SIN PUBLICAR

Ninguno...

CREAR UN NUEVO EXAMEN

LISTA DE MATERIAS

MATERIAS ASIGNADAS

Universidad Nacional de Lanus

Desarrollo productivo y tecnológico

Lic. Sistemas

- Ing. Software 1 Garcia Martinez
 - GUIA 1 ✖ ✔
 - GUIA 2 ✖ ✔
 - GUIA 3 ✖ ✔

Aceptación o Rechazo de alumnos en una materia (Docente):

AAED2 x

www.AAED2.com.ar

AAEV 2.0
Ambiente de Auto Evaluación 2.0

LISTA DE ALUMNOS PENDIENTES

Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1

DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔
DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔
DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔
DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔
DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔
DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔
DNI	Apellido	Nombre	fecha_de_solicitud	✖	✔

CANCELAR TODOS HABILITAR TODOS

Expulsión de alumnos de una materia (Docente):

AAEV 2.0
Ambiente de Auto Evaluación 2.0

 **LISTA DE ALUMNOS EN LA MATERIA**

Universidad Nacional de Lanus / Lic. Sistemas / Ing. Software 1

DNI	Apellido	Nombre	fecha_de_solicitud	
				
				
				
				
				
				

CANCELAR TODOS

Inicio del administrador:

AAEV 2.0
Ambiente de Auto Evaluación 2.0

 **SEBASTIAN MARTINS**

DOCENTES PENDIENTES

Di Dio Edgardo / Didioedgardo@unla.com.ar /matematica 1

 **LISTA DE MATERIAS**

MATERIAS ASIGNADAS

 **Universidad Nacional de Lanus**

 **Desarrollo productivo y tecnológico**

 Lic. Sistemas

• Ing. Software 1 García Martínez