

RESUMEN

DESARROLLO DE UNA APLICACIÓN WEB QUE AUTOMATICE
LA ACTIVIDAD DE EVALUACIÓN DOCENTE, EN LA
UNIVERSIDAD DE NAVOJOA, SONORA

por

Víctor Fragoso Vázquez

Asesor principal: Saulo Hernández Osoria

RESUMEN DE TESIS DE MAESTRÍA

Universidad de Morelos

Facultad de Ingeniería y Tecnología

Título: DESARROLLO DE UNA APLICACIÓN WEB QUE AUTOMATICE LA ACTIVIDAD DE EVALUACIÓN DOCENTE, EN LA UNIVERSIDAD DE NAVOJOA, SONORA

Nombre del investigador: Víctor Fragoso Vázquez

Nombre y título del asesor principal: Saúl Hernández Osoria, Maestría en Tecnologías de la Información

Fecha de terminación: Octubre de 2018

Problema

El presente estudio procuró desarrollar una aplicación web que automatice la actividad de evaluación docente, aprovechando la estructura digital y de procesos que maneja la Universidad de Navojoa, Sonora.

Método

En la recolección de los requerimientos en el desarrollo del sistema web para evaluación docente se utilizó el siguiente procedimiento: (a) observación directa del proceso de evaluación y el sistema académico, (b) revisión de la documentación y

legislación de la evaluación docente y (c) una entrevista con el director de investigación y calidad educativa, juntamente con el administrador del sistema académico institucional.

Resultados

Los resultados de la evaluación son automáticos, por lo que el docente puede conocer los puntos que necesita mejorar para una mayor satisfacción del alumno: planificación de sus clases y saber a qué necesita dar mayor hincapié en el desarrollo de su clase. El tener los resultados en una base de datos ayuda a poder realizar análisis más profundos y detallados de los mismos para tomar decisiones y sacar conclusiones de las decisiones previamente tomadas.

Conclusiones

1. La tecnología ayuda a que los procesos de las organizaciones puedan agilizarse y reducir los tiempos de aplicación y ejecución. En el presente proyecto, se mejoró el proceso de evaluación.

2. El uso de una metodología ayuda a una mejor documentación y comprensión del funcionamiento de la aplicación web, facilitando el crecimiento y la adaptación y reduciendo los errores y costos de modificaciones de los requerimientos.

3. Los diagramas de la base de datos dan la posibilidad de migrar o crecer con Mysql ya que, gracias a la reducción en el tiempo de procesamiento de la ejecución de diferentes procesos, se logra obtener mejores resultados.

4. El uso de tecnologías abiertas ayuda a reducir los costos sin sacrificar el rendimiento, ya que se usaron en la implementación herramientas comprobadas y utilizadas en el mercado, sin limitar el desarrollo de la aplicación.

5. El desarrollarlo en una aplicación web abre la posibilidad de poder realizar la evaluación en cualquier equipo con acceso a internet, garantizando la libertad y anonimato del alumno.

6. Los resultados se pueden consultar de manera instantánea, ya que el proceso de consulta es automático, reduciendo y optimizando el proceso de la evaluación dirigida por el departamento de calidad educativa.

7. El PHP es un lenguaje que está en constante evolución y documentación, garantizando que en futuras ampliaciones se facilite el crecimiento de la aplicación de evaluación e integración de nuevos módulos.

8. El modelo MVC ayuda a la seguridad de los datos y a la flexibilidad de la aplicación.

Universidad de Morelos
Facultad de Ingeniería y Tecnología

DESARROLLO DE UNA APLICACIÓN WEB QUE AUTOMATICE
LA ACTIVIDAD DE EVALUACIÓN DOCENTE, EN LA
UNIVERSIDAD DE NAVOJOA, SONORA

Tesis
presentada en cumplimiento parcial
de los requisitos para el grado de
Maestría en Ciencias Computacionales

por

Víctor Fragoso Vázquez

Octubre de 2018

DESARROLLO DE UNA APLICACIÓN WEB QUE AUTOMATICE
LA ACTIVIDAD DE EVALUACIÓN DOCENTE, EN LA
UNIVERSIDAD DE NAVOJOA, SONORA

Proyecto

presentado en cumplimiento parcial de
los requisitos para el grado de
Maestría en Ciencias
Computacionales

por

Amado Víctor Fragoso Vázquez

APROBADO POR LA COMISIÓN:



M.C. Saulo Hernández Osoria
Asesor principal




M.C. Benjamín Tass Herrera
Asesor externo



M.C. Alejandro Walterio García Mendoza
Miembro



Dr. Ramón Andrés Díaz Valladares
Director de Posgrado e Investigación



Dr. Ramón Andrés Díaz Valladares
Miembro

26-Nov-18
Fecha de aprobación

TABLA DE CONTENIDO

LISTA DE TABLAS	vi
RECONOCIMIENTOS	vii
Capítulo	
I. DIMENSIÓN DEL PROBLEMA	1
Introducción.....	1
Antecedentes.....	2
Planteamiento del problema	3
Declaración del problema	4
Justificación.....	5
Objetivos.....	5
Limitaciones	5
Delimitaciones.....	6
Defiición de términos.....	6
Marco filosófico.....	7
II. MARCO TEÓRICO.....	8
La evaluación.....	8
La evaluación del docente	8
La sociedad de la información.....	11
Crecimiento de la web.....	12
Lenguajes.....	14
Paradigma orientado a objetos.....	15
Metodología ICONIX	16
MVC.....	20
Arquitectura	22
PHP (Hypertext Pre-processor).....	22
III. MARCO METODOLÓGICO	25
Análisis de requisitos	25
Recolección de requerimientos	25
Personal involucrado.....	25

Conclusiones de la entrevista y la recolección de la documentación	26
Procesos del sistema	26
Proceso de evaluación docente.....	26
Proceso de entrega de resultados.....	26
Procesos de administración.....	27
Proceso de traspaso de datos	29
Restricciones	31
Requerimientos técnicos	31
Objetos del dominio	31
Elaboración de prototipos.....	32
Prototipos de validación del usuario	33
Prototipo de evaluación al docente.....	33
Prototipo de reporte de evaluación.....	34
Prototipo de reporte del administrador y el director.....	34
Prototipo de reporte de asignación docente – escuela.....	35
Prototipo de estatus docente.....	36
Prototipo de asignación de derechos.....	37
Prototipo de editar facultad.....	37
Prototipo de relación escuela-facultad.....	38
Prototipo de asignar evaluación.....	38
Modelo de casos de uso.....	39
Actores	39
Diagrama de caso de uso general.....	41
Análisis y diseño preliminar.....	41
Descripción de casos de uso.....	41
Validar usuario.....	43
Aplicar evaluación.....	43
Consultar reporte personal.....	44
Consultar reporte de escuela.....	44
Consultar reportes personalizados.....	44
Editar docentes.....	45
Modificar estatus.....	45
Editar facultad.....	45
Asignar facultad-escuela.....	46
Asignar derechos.....	46
Asignar evaluación.....	46
Diagrama de robustez.....	46
Implementación.....	47
Herramientas.....	49
Base de datos.....	49
Editor de código.....	50
Servidor http apache.....	51
Escribir y generar código.....	52

IV. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS	54
Introducción	54
Acceso.....	54
Automatización	55
Satisfacción	55
Resultados	55
V. RESUMEN, DISCUSIÓN, CONCLUSIONES Y RECOMENDACIONES	56
Introducción	56
Conclusiones.....	56
Trabajos futuros.....	57
Apéndice	
A. EVALUACIÓN DEL DESEMPEÑO DOCENTE	59
B. RESULTADOS DE EVALUACIÓN.....	63
C. DESCRIPCIÓN DE CASOS DE USO.....	67
D. CASO DE USO POR ACTOR Y MÓDULO.....	74
E. DIAGRAMA DE ROBUSTEZ.....	78
F. CÓDIGO FUENTE.....	86
REFERENCIAS.....	118

LISTA DE FIGURAS

1. Usuarios de Internet en el mundo por región geográfica	14
2. Cronología de lenguajes de programación 1950-2013.....	16
3. La herencia en la programación orientada a objetos.....	17
4. Fases de iconix	18
5. Ejemplos de diagramas UML	19
6. Modelo-vista-controlador	21
7. Modelo MVC en PHP	21
8. Arquitectura cliente/servidor	23
9. PHP de un servidor web	24
10. Diagrama de clases del dominio	32
11. Validar usuario	33
12. Responder evaluación	34
13. Resultado evaluación	35
14. Buscar docente	35
15. Asignar docente-escuela	36
16. Estatus del docente	36
17. Estatus del docente	37
18. Editar facultad	38
19. Asignar facultad-escuela	39
20. Asignar evaluación	40
21. Diagrama de casos de uso	42
22. Elementos del diagrama de robustez	47
23. Diagrama ER de la base de datos	48
24. Base de datos en Mysql	49
25. Entorno de desarrollo en Sublime Text	50
26. Cuota de mercado sitios web de mayor actividad	51
27. Estructura de la aplicación web.....	52
28. GitHub de AdminLTE.....	53

LISTA DE TABLAS

1. Requerimientos técnicos	31
2. Actores del sistema	40
3. Descripción de casos de uso	43

RECONOCIMIENTOS

Agradezco a quienes hicieron posible y colaboraron en la realización de este proyecto de investigación.

A la Universidad de Navojoa, por la información y apoyo en el cumplimiento de este proyecto.

A mis asesores de tesis, por ayudarme en la realización de este proyecto con su tiempo, recomendaciones y asesorías.

A mis alumnos y compañeros de trabajo, quienes me han brindado su ayuda, para realizar esta investigación

Al ingeniero Daniel Lozano, quien me acompañó en la realización de este proyecto.

A mi familia, por su constante apoyo incondicional, consejos y oraciones constantes.

A Dios, que me da la vida, la sabiduría y la oportunidad de servicio en este proyecto.

CAPÍTULO I

DIMENSIÓN DEL PROBLEMA

Introducción

La sociedad de la información ha desarrollado nuevas tecnologías cambiando tanto la forma de pensar como la manera de recibir información. Esto ha originado la necesidad de espacios donde todos pueden opinar y la educación debe enfrentarse a estos nuevos desafíos.

La educación en esta sociedad de la información busca todos los instrumentos, acciones, medios de información y comunicación para generar un proceso educativo que llegue a ser colaborativo. El apoyarse de la tecnología ayuda a ser digital, interactivo e innovador.

Valencia Rodríguez y Vallejo Cardona (2015) afirman que

la evaluación es un proceso permanente, desde una perspectiva ética, ha de ser eminentemente formativa; de tal manera que tanto el maestro como el estudiante se reconozcan y dialoguen, avanzando de manera organizada en la construcción de conocimientos, saberes, habilidades, destrezas y valores. (p. 211)

White (1971) dice que actualmente se requiere de personas creadoras, aportando ideas que ejerzan un liderazgo. La verdadera educación de calidad significa “más que seguir cierto curso de estudios. Es amplia, incluye el desarrollo armonioso de todas las facultades físicas, mentales y espirituales” (p. 13).

Antecedentes

La calidad educativa y la evaluación constituyen una realidad compleja (Bertoni, Poggi y Teobaldo, 1993). Dependen de algunos factores que son usados por la administración educativa como el medio que analiza cada una de las acciones a realizar, encaminadas a una educación dinámica integral.

Al respecto, Canales Sánchez (2007) dice que

la evaluación es una dimensión fundamental del campo educativo y requiere mayor atención si asumimos que en buena medida de ella depende tanto la actividad de los profesores como el aprendizaje de los alumnos, la gestión de las autoridades y el sistema en su conjunto. El proceso educativo, al igual que el resto de las actividades, se desenvuelve en un nuevo escenario, en donde las tecnologías de la información y la comunicación, los adelantos científicos y tecnológicos, y al mismo tiempo la formación de bloques regionales, la búsqueda de competitividad, los rezagos sociales, las desigualdades, las asimetrías entre y al interior de las naciones, desafían el papel que puede cumplir la educación. (p. 48)

El mismo autor señala que “la evaluación no ha permanecido al margen de este debate y hoy es parte de las reflexiones sobre cómo enfrentar los retos del nuevo entorno, qué perspectivas adoptar y cómo mejorar tanto los sistemas como el aprendizaje de las personas” (p. 48).

En el proceso de lograr la mejora continua, las instituciones de educación han desarrollado aplicaciones web que pueden contener elementos que permiten una comunicación activa entre el usuario y la información, llevando a implementar estas soluciones en la evaluación docente. Instituciones como la Universidad Nacional Autónoma de México (<http://www.arquitectura.unam.mx/evaluacion>), la Universidad de Morelos (<https://apps.um.edu.mx>), la Universidad Autónoma de Yucatán (<https://intranet.matematicas.uady.mx/evaluacion/index.php>), el Instituto Tecnológico

de San Luis Potosí (<http://sii.itslp.edu.mx>), los Colegios de Bachilleres del Estado de Baja California (<http://cobachbc.edu.mx/evadoc>), entre otras, incluyen aplicaciones web en la evaluación docente como parte del análisis para la toma de decisiones y lograr una mejora continua.

Planteamiento del problema

En la experiencia del investigador, las empresas que ofrecen algún servicio o producto se mantienen en constante proceso de evaluación para ser más eficientes y dar un mejor servicio, ya que la competitividad y la globalización las mantiene en constante búsqueda de la calidad.

Capelleti (1999, citado en Horbath y Gracia, 2014) señala que

la evaluación debe contemplar un proceso comprensivo de análisis del desempeño del alumno, dinámico, crítico, creativo, cooperativo, que presupone el acompañamiento constante y que toma en cuenta las diversas dimensiones de la actuación del alumno. En este sentido, contribuye a la toma de decisiones y al mejoramiento de la calidad de la enseñanza, y enfatiza un aspecto de diagnóstico procesal, informando a los protagonistas de la acción para su perfeccionamiento constante. (p. 61)

Horbath y Gracia (2014) dicen que

como medio o instrumento, la educación es el vehículo por medio del cual los individuos de una sociedad no sólo se aproximan al conocimiento, sino que acceden a nuevas formas y dimensiones de su vida en sociedad que contribuyen a mejorar la calidad de esta, tanto a nivel individual como en sociedad. (p. 62)

Las universidades mexicanas evalúan al docente como parte de los procesos de planeación. “La evaluación como actividad integrada a los procesos sociales que tienen lugar en las instituciones escolares se ha desarrollado con tal velocidad que cada día se le reconoce como una actividad profesional altamente especializada” (Rueda Beltrán, Elizalde Lora y Torquemada González, 2003, p. 72).

La Universidad de Navojoa, en busca de lograr esta realimentación, aplica esta evaluación cada semestre, donde el alumno contesta un instrumento que maneja el departamento de calidad educativa. El proceso tiene algunas deficiencias, ya que se aplica a los alumnos repartiéndoles hojas sin verificar las materias y los docente que les dan esas materias; posteriormente, se capturan los datos para darle al docente, al coordinador y a la administración el resultado de la evaluación sin saber la evolución del docente o de la materia en algunas etapas anteriores. El costo tiempo-trabajo es muy alto y los resultados no son tan integrales como pueden llegar a serlo.

Es necesario que la evaluación de la actividad docente en la Universidad de Navojoa sea eficiente para lograr mejorar su práctica. La opinión de los estudiantes sobre el desempeño de cada uno de sus profesores es crucial, ya que los resultados pueden repercutir en la planeación de las actividades institucionales, como la formación de profesores, la calidad en los programas educativos y el estímulo al desempeño académico.

El proceso que actualmente se lleva en la Universidad de Navojoa para esta importante actividad es largo y no muy confiable, porque se aplica de forma manual, convirtiéndose en un proceso lento y tedioso.

El propósito principal de este proyecto es analizar, diseñar, desarrollar e implementar una aplicación web que permita al estudiante agilizar y mejorar el proceso de evaluación en la Universidad de Navojoa.

Declaración del problema

De lo dicho anteriormente, se desprende la necesidad de una aplicación web

que automatice la actividad de evaluación docente aprovechando la estructura digital y de procesos que maneja la Universidad de Navojoa.

Justificación

Las instituciones de educación superior enfrentan estándares de calidad que las obligan a ser competitivas e innovadoras en el desarrollo de su tarea, evaluando su labor con el fin de que sus alumnos tengan calidad y competitividad para el mercado laboral.

La percepción, velocidad y confiabilidad de la actividad de evaluación demanda a la Universidad de Navojoa modernizar el proceso, para lograr un mejor avance en la realimentación de sus profesores y, por tanto, el de sus programas académicos.

Objetivos

Para el presente estudio se plantean los siguientes objetivos:

1. Desarrollar una aplicación Web para la automatización de la evaluación docente en la Universidad de Navojoa.
2. Aplicar la evaluación docente de manera confiable para que aumente la satisfacción del cliente.
3. Agilizar y mejorar el proceso de evaluación de dicha universidad.
4. Mejorar los resultados obtenidos de las evaluaciones docentes que se realizan.

Limitaciones

Algunas limitaciones de esta investigación son las siguientes:

1. La falta de cooperación por parte de las diferentes escuela y/o de la dirección de calidad educativa, no dando el uso y la divulgación necesaria para el uso de la aplicación.

2. La disponibilidad de recursos para realizar la presente investigación.

Delimitaciones

A continuación, se presentan algunas delimitaciones en esta investigación:

Se desarrolló un sistema web automatizado en la Universidad de Navojoa que permitió agilizar y modernizar el proceso de evaluación docente a través de la web.

Dicha implementación involucró el estudio y aplicación de las tecnologías utilizadas en el desarrollo web; por tal motivo, la aplicación se adaptó a las políticas y procedimientos que el departamento de calidad educativa tiene establecidos respecto del proceso e instrumento de evaluación docente. La aplicación web fue realizada solo para ser utilizada por los usuarios (alumnos, docentes y administrativos) de la Universidad de Navojoa.

Definición de términos

A continuación, se definen algunos términos utilizados en esta investigación:

Aplicación web: conjunto de recursos web que participan en el funcionamiento de la aplicación web (Aumaille, 2002).

Evaluación docente: proceso donde el departamento de calidad educativa aplica un instrumento de evaluación por medio de los alumnos.

Instrumento de evaluación: cuestionario que se entrega para conocer la opinión de los estudiantes.

Calidad: conjunto de propiedades y características de un producto o servicio que le confiere su aptitud para satisfacer unas necesidades expresadas o implícitas (International Organization for Standardization, ISO, 2011).

Marco filosófico

Las instituciones deben ofrecer un servicio que glorifique a Dios y que haga que los estudiantes se sientan satisfechos con el servicio recibido. La Biblia, en Génesis 1:31, declara que cuando Dios vio el fruto de su trabajo, lo llamó muy bueno. Dios examinó y evaluó la calidad de su trabajo y, cuando él determinó que había hecho un buen trabajo, se regocijó con el resultado.

White (2005) señala que “todo alumno puede tener la satisfacción de saber que ha estado de parte de Cristo, manifestando respeto por el orden, la diligencia y la obediencia” (p.127).

El trabajo se debe hacer con regocijo. Al respecto, 1Corintios 15:58 declara lo siguiente: “por lo tanto, mis queridos hermanos, manténganse firmes e incommovibles, progresando siempre en la obra del Señor, conscientes de que su trabajo en el Señor no es en vano”. Marcos 9:35 dice “entonces Jesús se sentó, llamó a los doce y les dijo: —Si alguno quiere ser el primero, que sea el último de todos y el servidor de todos”.

White (1995) afirma que “solo mediante un mejoramiento continuo de las facultades intelectuales, tanto como de las morales, se puede esperar satisfacer el propósito del Creador y de los semejantes” (p. 253).

CAPITULO II

MARCO TEORICO

La evaluación

La evaluación ayuda a determinar de manera sistemática cuál es el mérito, el valor y el significado de algo o alguien en función de un conjunto de criterios que son determinados por un conjunto de normas (Real Academia Española, 2014). Se utiliza en la actualidad en una amplia gama de empresas orientadas a prestar servicios o producir bienes para evaluar al personal, a los procesos y productos, incluso a los gobiernos, las fundaciones y las organizaciones sin fines de lucro.

Las organizaciones evalúan para generar criterios objetivos de crecimiento, calidad y productividad. Además, tienen la tarea de examinar a las personas para determinar si reúnen las condiciones para determinadas tareas y condiciones.

La norma ISO (2011) comenta que la evaluación de la conformidad puede ser elaborada por terceros independientes o el proveedor, dependiendo del comprador o requisitos de la autoridad. Incluye inspección, ensayo, certificación del producto o cualquier combinación de estas actividades que sea pertinente.

La evaluación docente

En la educación, la evaluación de la docencia es importante para el desarrollo de cualquier institución educativa, debido a que el profesor es uno de los principales

agentes del proceso de enseñanza-aprendizaje (Arce Orozco , 2010).

Márquez Jiménez (2016) afirma que “la evaluación de los docentes en México constituye uno de los ejes fundamentales de las políticas educativas, dada la convicción de que de ello depende la posibilidad de mejorar la “calidad” de la educación en el país” (p. 3).

Para la Secretaría de Educación Pública (2014), la evaluación del docente

significa evaluar, por un lado, el grado de cumplimiento de sus funciones y responsabilidades establecidos por la institución de pertenencia y por otro, la calidad con que se lleva a cabo la función en términos de rendimiento y los logros obtenidos en un tiempo determinado que permitan realizar una valoración y dar a conocer las fortalezas y áreas de oportunidad para la mejora de la función. (p. 11)

La Asociación Nacional de Universidades e Instituciones de Educación Superior (2000, citada en Osés Bargas, Duarte Briceño y Esquivel Alcocer (2007) afirma que

el mejoramiento de la calidad está íntimamente ligado, además, con la continua innovación tanto en el ámbito académico como en el de los procesos de gestión. Uno de los aspectos que puede constituirse en uno de los pilares de la calidad de la educación superior es la cultura de la evaluación, la cual es un componente estructural de cada proyecto, de cada programa, de cada acción que se emprende. En el proceso educativo la evaluación debe estar siempre presente. (pp. 13-14)

Para el caso de México, Canales Sánchez (2008) afirma que

la actividad docente constituye una de las principales funciones de las instituciones de educación superior, y prácticamente la única en la mayoría, no se ha visto reflejada en las principales iniciativas de mejora para el sector puestas en marcha en las dos últimas décadas. En particular, se destaca que a partir de la amplia política de evaluación instaurada en el sistema educativo al final de los años ochenta, la docencia no figuró en las preocupaciones de los mecanismos ni de las recompensas del sistema de evaluación. Las medidas que posteriormente se instrumentaron, intentaron remediar la situación principalmente mediante la opción de optimizar el tiempo de dedicación y las calificaciones de quienes desempeñaban tal función; no se ocuparon de mejorar la actividad docente en estricto sentido, ni de diseñar y aplicar un esquema de evaluación que la alentara y fortaleciera. (p. 1)

Crispín (1998, citado en Loredo Enríquez, Romero Lara e Inda Icaza (2008) dice que

lo que las instituciones educativas esperan del docente ha cambiado; los objetivos educativos se transformaron. Antes, se esperaba que el profesor fuera capaz de impartir una buena cátedra y transmitir sus conocimientos de una manera clara; hoy en día se espera que diseñe diferentes estrategias que permitan a los alumnos aprender a plantear y resolver problemas, a pensar en forma crítica y a ser creativos. (p. 4)

Por lo tanto, se recomienda atender las necesidades detectadas en busca de una formación docente que permita la mejora continua.

De Vicente (1996, citado en Catalán y González (2009) señala que

el desarrollo profesional del profesor necesita apoyarse en los resultados de la evaluación de su acción educativa, tanto a nivel específico como genérico; así, deberá conocerse su capacidad para dominar y aplicar las habilidades técnicas para la práctica; habrá de saberse hasta qué punto es capaz de incrementar la claridad, el significado y la coherencia a través de la práctica reflexiva; será necesario averiguar si está dotado del conocimiento suficiente y de la habilidad necesaria para la investigación y la exploración; si posee, en fin, la adecuada disposición a recibir y dar ideas, a prestar asistencia y a recibirla de otros, a la colaboración. (p. 99)

Ante tales exigencias, Rueda Beltrán, Elizalde Lora y Torquemada González (2003) señalan la importancia de obtener información detallada de cómo se está realizando la actividad docente, afirmando que cada día son más los países e instituciones de nivel superior que se preocupan por adoptar algún sistema para evaluar a los docentes.

Los propósitos de la evaluación del profesorado pueden definirse como la responsabilidad educativa y el desarrollo profesional; por lo tanto, resulta muy importante que la evaluación sea válida y fiable para lograr el propósito por el que se aplica (Fuentes Medina y Herrero Sánchez, 1999).

Ahora el problema se centra en cuál es su propósito, cuáles son las fuentes de información, quién la realiza y cuál es la metodología apropiada para su realización (Arce Orozco, 2010).

Por todo esto, el proceso de evaluación en cualquiera de sus vertientes puede tener riesgos, deficiencias y un uso inapropiado si no se toman en cuenta las herramientas adecuadas (Santos Guerra, 1993).

La Organización para la Cooperación y el Desarrollo Económicos (2009) recomienda que

los gobiernos modelen sus sistemas de evaluación y reconocimiento de forma congruente con elementos comunes que emergen de otras experiencias más exitosas revisadas, entre los elementos está que la Información sea robusta y el sistema de datos pueda relacionar el desempeño de los estudiantes a los docentes y escuelas a través del tiempo. (p. 77)

El manejo de la información y la calidad de sus servicios hacen importante tener sistemas de gestión que satisfagan los requisitos exigidos. Las organizaciones que no controlen sus procesos no podrán conseguir buenos resultados (Van Bon et al., 2008).

La tecnología ha venido a satisfacer estos requisitos, ya que el uso del software y el almacenamiento inteligente de los datos puede ampliar la utilidad de estos datos acumulados (Daft, 2007).

La sociedad de la información

Uno de los primeros autores que utiliza el término sociedad de la información (SI) es Fritz Machlup, aunque una mejor retrospectiva del concepto fue desarrollada por Yoneji Masuda en 1980 (Briggs y Burke, 2005).

Este término se puede definir como una sociedad interesada en crear, consultar, utilizar y compartir información que atraviesa todas las actividades. En este tipo de

organización social, la información ocupa un lugar sustantivo y se convierte en fuente de riqueza (Crovi Druetta, 2002).

Estudillo García (2001) dice que

el término sociedad de la información se ha utilizado para describir sistemas socioeconómicos que emplean extensivamente la información en sus áreas de actividad. El uso extensivo de tecnologías de información ha provocado que se identifique a éste como el elemento que impulsa el desarrollo hacia una sociedad de la información. (p. 84)

El mismo autor afirma que

otros elementos que han sido considerados para explicar la conformación de una sociedad de la información han sido la educación, la economía y el elemento global donde se relaciona el espacio y tiempo. Ninguno de estos elementos por sí solo han podido explicar la transición de una sociedad industrial a una sociedad de la información, pero considero que los cinco elementos en conjunto nos orientan a interpretar en qué consiste. (p. 85)

Actualmente existe mucha información que cada vez se multiplica en menor tiempo, así que, el paso siguiente sería el acceso al conocimiento; sería importante saber cómo se procesa y se analiza esa información.

El crecimiento rápido de las tecnologías de información, comunicación y la digitalización es una de las claves técnicas de la sociedad de la información, proceso que ha dado lugar a nuevos medios, formas de producir, almacenar y difundir la información (Peres Núñez e Hilbert, 2009).

Crecimiento de la web

Con respecto al crecimiento de la web, Luján Mora (2002) comenta que

en 1989 Tim Berners-Lee y Robert Cailliau presentan el proyecto World Wide Web mientras trabajaban en el CERN en Ginebra, Suiza, con las características básicas que perduran hoy. Este estudio Tim Berners-Lee lo completó en 1990, mostrando el primer servidor web y el primer cliente. En base a esto, se deben tres elementos claves del nacimiento de la Web: HTML (Lenguaje de Marcado de Hipertexto), HTTP (Protocolo de Transferencia de Hipertexto) y

URL (Identificador de Recurso Uniforme). Esto permitió poder con facilidad, a través de URL, saltar de un lugar a otro del mundo leyendo texto e imágenes. (p. 6)

En este inicio, los contenidos eran estáticos y la navegación lenta; las computadoras tenían memoria con procesadores de poca potencia, la velocidad de las redes era baja; la productividad en el uso de la red era muy pobre.

Posteriormente, la interfaz Common Gateway Interface, para la generación dinámica de páginas web, consigue que la información pueda ser actualizada en tiempo real al poderse enlazar con una, tomando en cuenta la entrada de datos por parte del usuario (Redondo Duque y Bravo Santos, 2005).

Los mismos autores señalan que, en 1994, se creó Mosaic por Netscape, un navegador o browser gráfico que ayudó a impulsar la propagación de la web.

Como dice Pérez González (s.f),

la Web facilita la interacción de los usuarios que participan generando contenidos y valorando aplicaciones y servicios, fundamentalmente en las organizaciones que poseen grandes cantidades de información, y aportar conocimientos que pueden resultar una herramienta muy útil para innovar todo tipo de productos y servicios. Permite además, crear contenidos (textos, fotos, audio, vídeo) sin tener conocimientos de programación. (p. 3)

En la actualidad existen millones de personas conectadas a internet y a sitios web, además de montañas de información disponible, como lo muestra la Figura 1.

Las diferentes herramientas informáticas han ido evolucionando, permitiendo al desarrollador incorporar conceptos multimedia, y al usuario disfrutar con sitios web de mayor calidad estética, interactiva y a nivel de sus interfaces.

Internet Users in the World by Geographic Regions - 2012 Q2

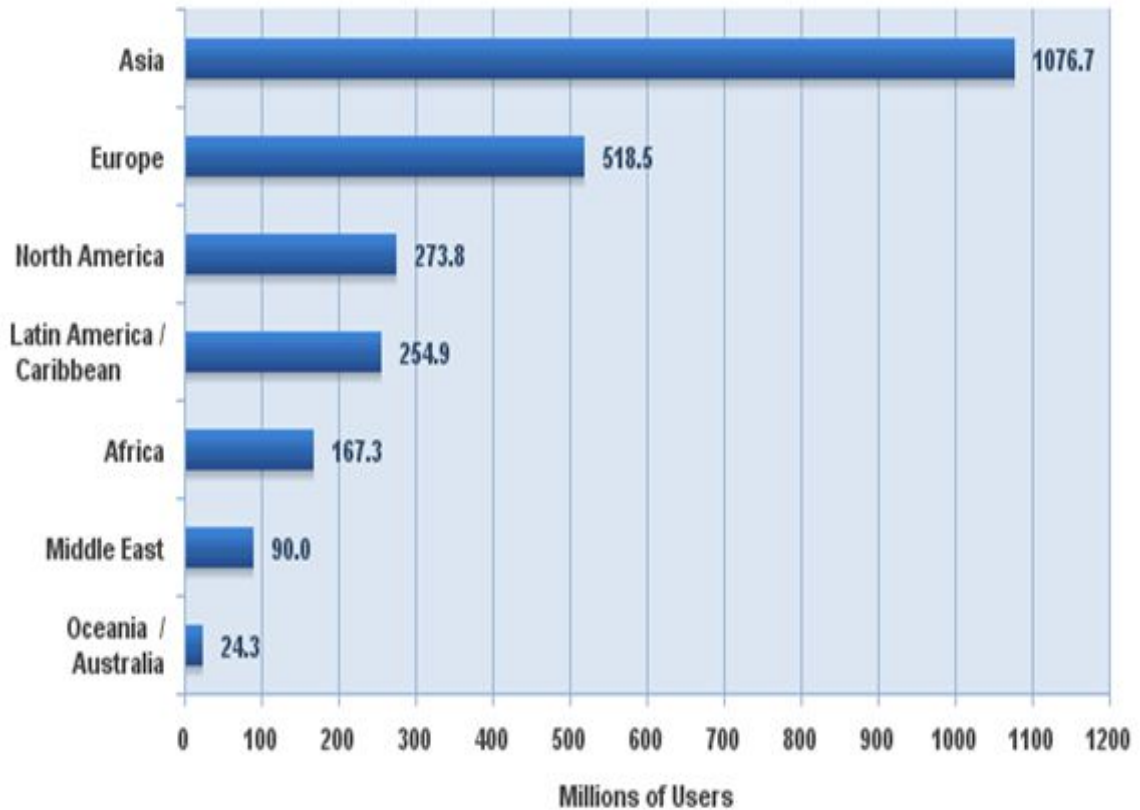


Figura1. Usuarios de Internet en el mundo por región geográfica.

Lenguajes

Morales (2014) señala que

el lenguaje de programación es básicamente un sistema estructurado de comunicación, similar al humano, el cual permite comunicar por medio de signos, ya sean palabras, sonidos o gestos. Refiriéndonos a los aparatos, este sistema está organizado para que se entiendan entre sí y a su vez interprete las instrucciones que debe ejecutar. (p. 1)

La comunicación con las computadoras se basa en lenguaje de programación,

que ayuda a darle instrucciones que controlen el hardware y el software. Las instrucciones se llaman código fuente, que genera un código objeto. Lo interpreta la máquina con base en el sistema operativo (Lutz, 2009).

En la actualidad existen alrededor de 2500 lenguajes de programación separados en diferentes paradigmas, debido a su forma de conceptualizar, al nivel de abstracción y a la manera en que organizan las tareas que se lleven a cabo para su ejecución.

Los lenguajes de programación se separan en paradigmas; cada paradigma transforma los problemas en diferentes soluciones, donde se tiene los lenguajes imperativos, funcionales, lógicos, demostrativos, de transformación, basado en formularios, flujo de datos, basado en restricciones y orientado a objetos (Amo, Martínez Normand y Segovia Pérez, 2005).

Cada uno de los lenguajes de programación intenta dar solución a los desafíos que se tienen al momento de resolver los problemas; esto genera la necesidad de determinar el lenguaje de programación ideal donde se intenta utilizar el de mejor calidad (Cancher Castro, Ponce de León Amador y Saquete Boró, 2006).

Paradigma orientado a objetos

El paradigma orientado a objetos es un método de implementación en el cual los programas se organizan como colecciones cooperativas de objetos (ver Figura 2). Según Grady Booch (Sommerville, 2005), cada uno de los cuales representan una instancia de alguna clase y cuyas clases son todas miembros de una jerarquía de clases unidas mediante relaciones de herencia.

Mi Perro Doggy salta



Figura 2. Ejemplo de paradigma orientado a objetos.

Izquierdo (s. f.) dice que

la programación orientada a objetos es otra forma de descomponer problemas. Este nuevo método de descomposición es la descomposición en objetos; vamos a fijarnos no en lo que hay que hacer en el problema, sino en cuál es el escenario real del mismo, y vamos a intentar simular ese escenario en nuestro programa. (p. 2)

El mismo autor afirma que

la programación orientada a objetos va más allá, permitiéndonos definir clases a partir de otras clases ya construidas. Por ejemplo, las bicicletas de montaña, las de carretera y los tándems son todos, en definitiva, bicicletas. En términos de programación orientada a objetos, son subclases o clases derivadas de la clase bicicleta. Análogamente, la clase bicicleta es la clase base o superclase de las bicicletas de montaña, las de carretera y los tándems. Esta relación se muestra en la Figura 3.

Metodología ICONIX

De San Martín Oliva (s. f.) dice que “el desarrollo de aplicaciones va cambiando por innovaciones tecnológicas, estrategias de mercado en la industria de la informática; esto lleva a los desarrolladores de aplicaciones a evolucionar para obtener aplicaciones en menor tiempo, más vistosas y de menor costo” (p. 4).

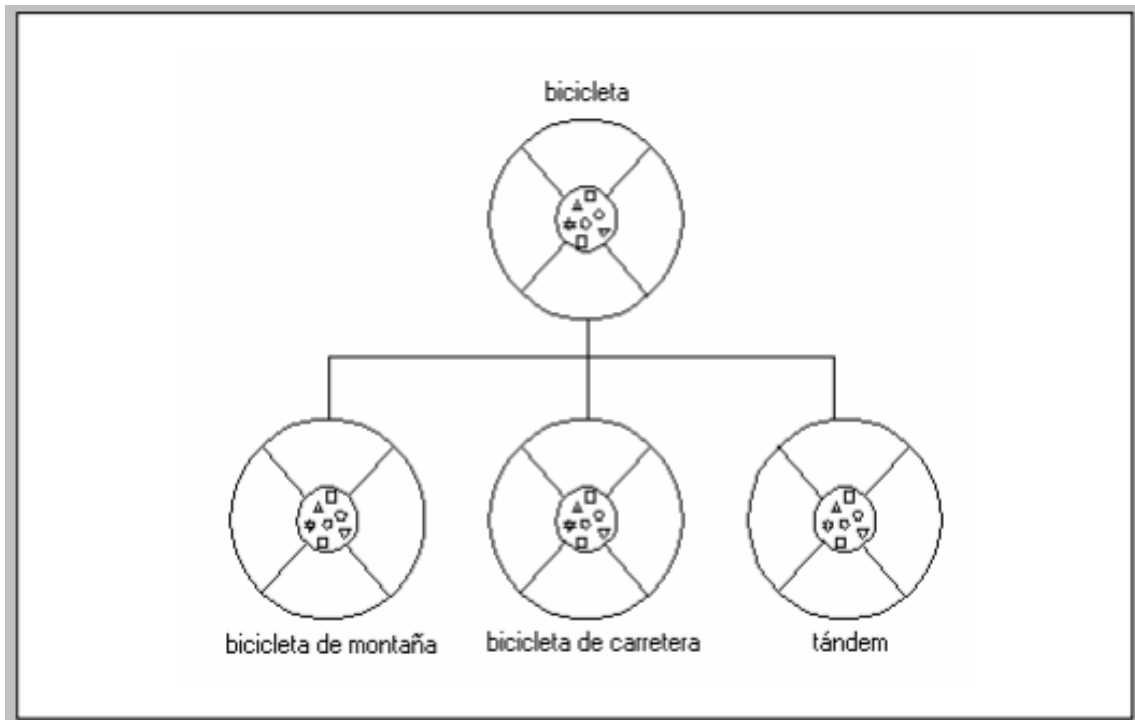


Figura 3. La herencia en la programación orientada a objetos.

Amavizca Valdez, García Ruíz, Jiménez López, Duarte Guerrero y Vásquez

Brindis (2014, citados en Rosales Romero, 2017) comenta que

la metodología ICONIX, elaborada por Doug Rosenberg y Kendall Scott en 1993, se define como un proceso de desarrollo de software práctico; ubicándose entre la complejidad del RUP (Rational Unified Processes) y la simplicidad y pragmatismo del XP (Extreme Programming) sin descartar las tareas de análisis y diseño que XP no contempla. (p. 38)

De acuerdo con De San Martín Oliva (s. f.), las tres características fundamentales de ICONIX son las siguientes:

1. Es iterativo e incremental: varias iteraciones ocurren entre el desarrollo del modelo del dominio y la identificación de los casos de uso. El modelo estático es incrementalmente refinado por los modelos dinámicos.

2. Tiene trazabilidad: cada paso está referenciado por algún requisito. Se define trazabilidad como la capacidad de seguir una relación entre los diferentes artefactos producidos.

3. Posee la dinámica del UML: la metodología ofrece un uso “dinámico del UML” como los diagramas del caso de uso y los diagramas de secuencia y de colaboración” (p. 5).

Benítez Chica (2014, citado en Rosales Romero, 2017) señala que ICONIX se compone de las tres fases siguientes: (a) análisis de requisitos, (b) análisis y diseño preliminar y (c) diseño e implementación. La Figura 4 muestra cada una las fases mencionadas.

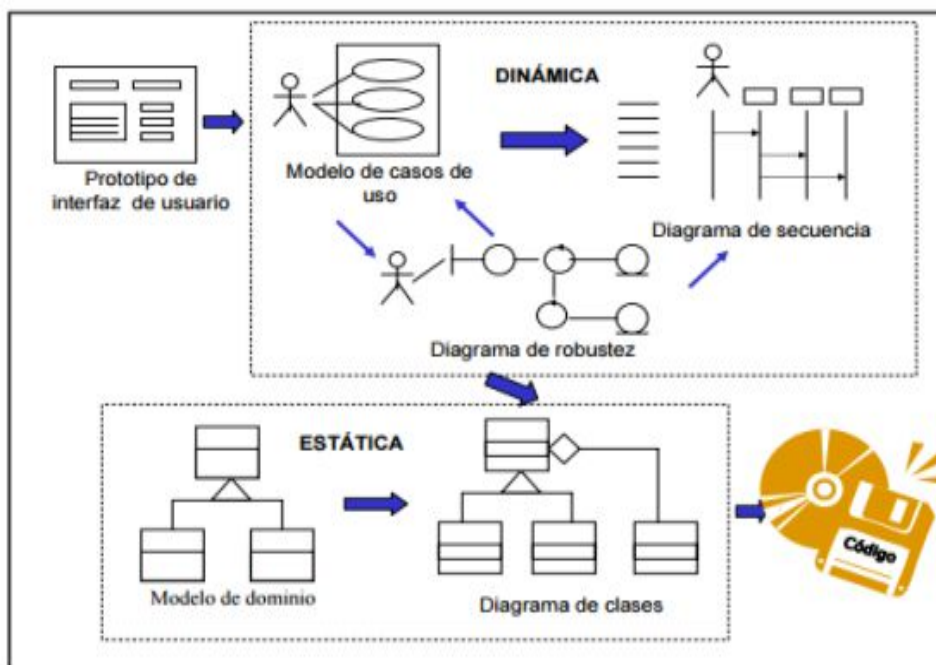


Figura 4. Fases de iconix (Benítez Chica, 2014).

Dentro del marco de modelado que respalda la Object Management Group (OMG), se encuentra la especificación UML (Unified Modeling Language), un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software (ver Figura 5) (Booch, Rumbaugh y Jacobson, 2005).

Esta nació con la unión de las metodologías y notaciones de Grady Booch y James Rumbaugh, proporcionando un plano desde varios ángulos, cubriendo los aspectos conceptuales, los procesos del negocio y las funciones del sistema.

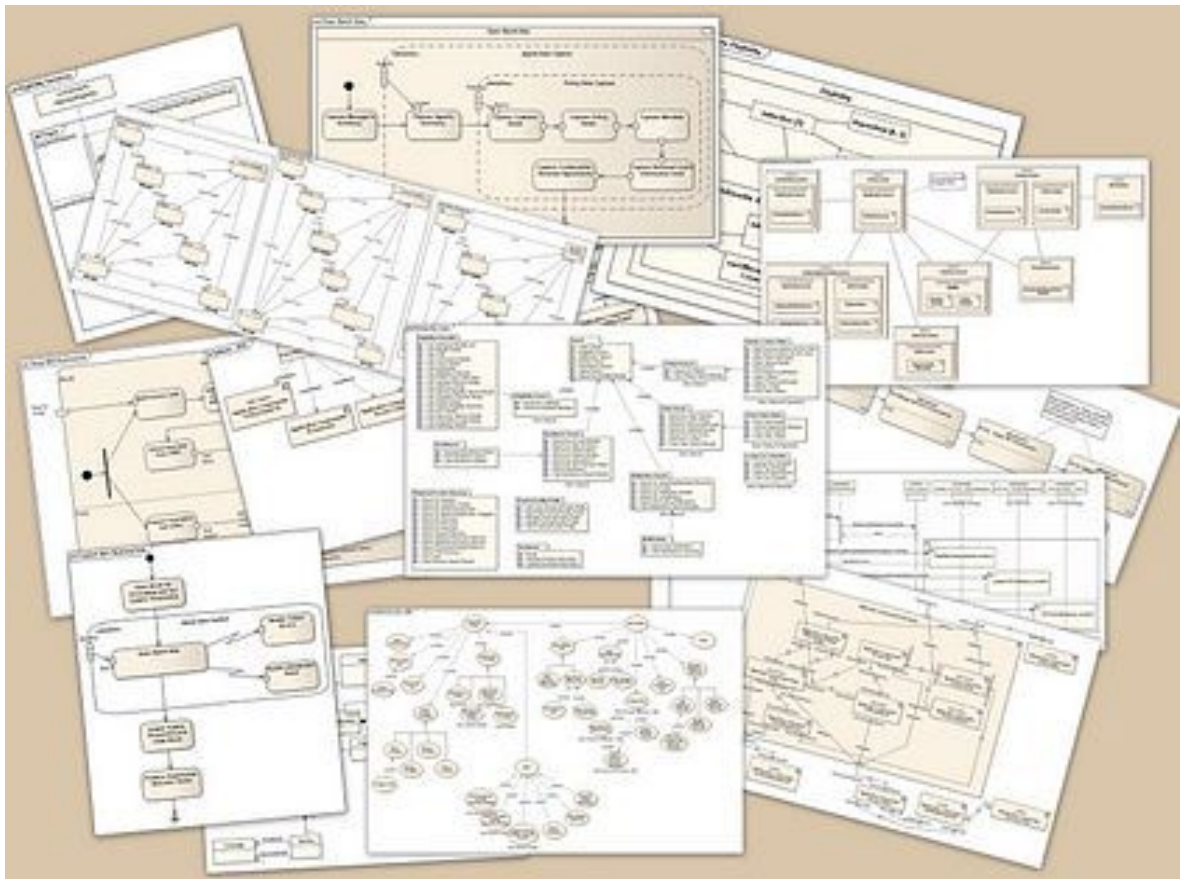


Figura 5. Ejemplos de diagramas UML.

De acuerdo con de San Martín Oliva (s. f.),

ICONIX es un proceso simplificado en comparación con otros procesos más tradicionales, que unifica un conjunto de métodos de orientación a objetos, con el objetivo de abarcar todo el ciclo de vida de un proyecto. Fue elaborado por Doug Rosenberg y Kendall Scott a partir de una síntesis del proceso unificado de los “tres amigos”: Booch, Rumbaugh y Jacobson y que ha dado soporte y conocimiento a la metodología ICONIX desde 1993. Presenta claramente las actividades de cada fase y exhibe una secuencia de pasos que deben ser seguidos. Además, ICONIX está adaptado a los patrones y ofrece el soporte de UML, dirigido por casos de uso y es un proceso iterativo e incremental. (p. 5)

MVC

Existen varios patrones de diseño que proporcionan beneficios considerables cuando se aplican. Un patrón es el paradigma MVC (modelo-vista-controlador) que divide la aplicación en tres componentes interoperables (Sommerville, 2005), como se observa en la Figura 6.

El modelo representa los datos del negocio y la lógica empresarial que regulan el acceso y la modificación de los datos. Se limita a la vista y su controlador facilita las presentaciones visuales complejas, como acceso al controlador de funcionalidad de la aplicación.

El diseño de una aplicación con PHP de nivel web puede cumplir con el patrón de diseño MVC (ver Figura 7), ya que sirve peticiones HTTP al más alto nivel. El nivel de web hace cuatro cosas básicas en el orden específico siguiente: interpreta las solicitudes del cliente, envía esas peticiones a la lógica del negocio, selecciona la siguiente vista de la pantalla y genera y entrega la siguiente vista.

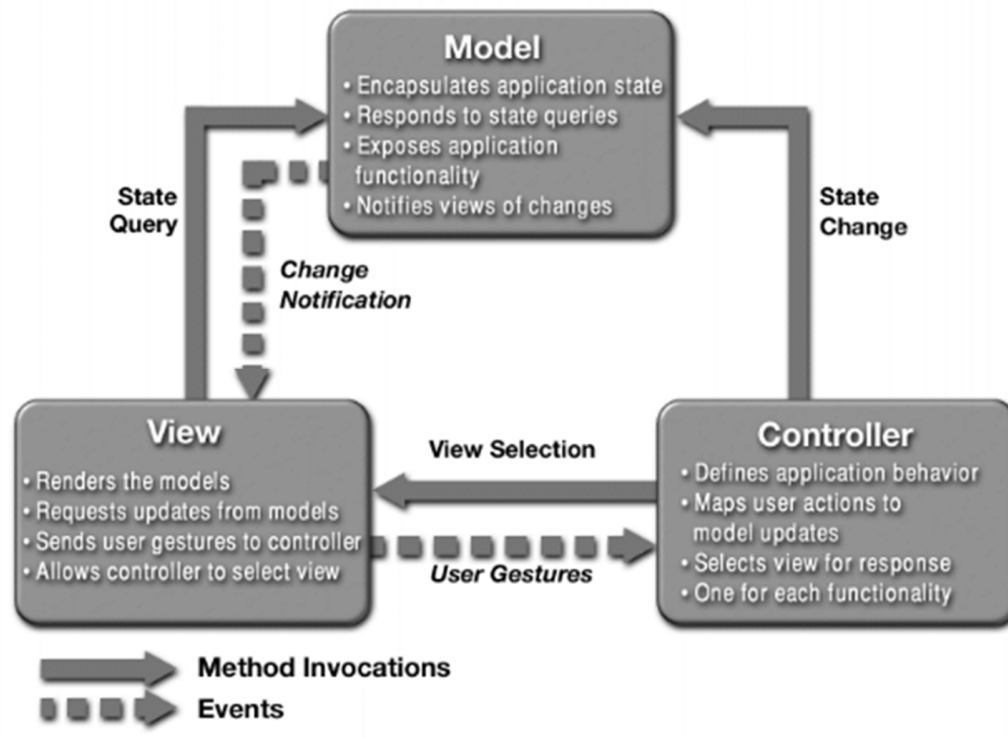


Figura 6. Modelo-vista-controlador.

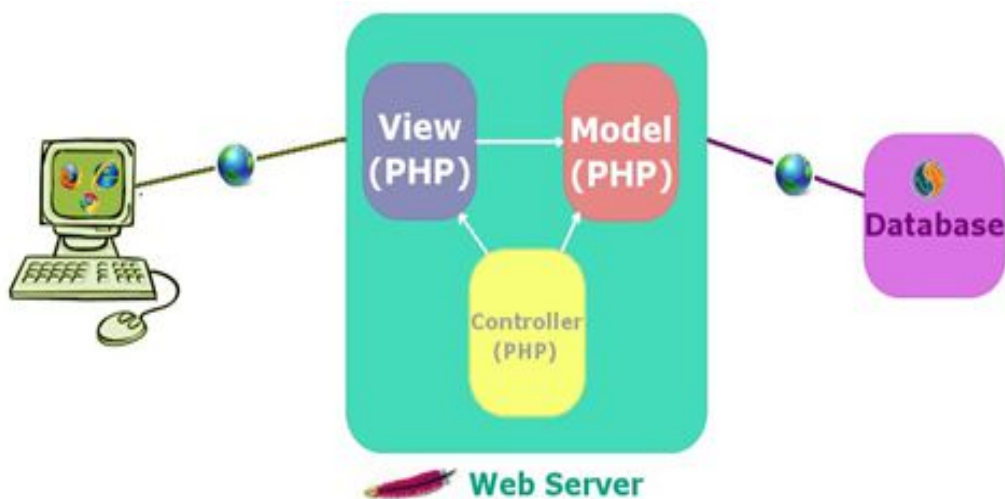


Figura 7. Modelo MVC en PHP.

Arquitectura

La complejidad del desarrollo del software ha llevado a desarrollar formas y guías que estructuran una aplicación, como los planos en un edificio; a esto se le llama arquitectura del software, un paso esencial en el desarrollo de cualquier aplicación o sistema de software.

En palabras de Bass, Clements y Kazman (2003), la arquitectura de software es la estructura o estructuras del sistema, lo que comprende los componentes del software, sus propiedades externas visibles y las relaciones entre ellos.

La arquitectura propuesta en este proyecto busca potencializar la principal característica de la aplicación, ya que es necesario que soporte el desarrollo de aplicaciones web.

En la arquitectura de red cliente/servidor, cada proceso o computadora en la red es cliente o servidor (Luján Mora, 2002). Está compuesta por los tres elementos básicos siguientes: el cliente, el servidor y el middleware, como se muestra en la Figura 8. Esto abarca todo el software distribuido necesario para el soporte de interacciones entre clientes y servidores; es decir, es el enlace que permite que un cliente obtenga un servicio del servidor (Perdomo Vargas y Ochoa Guevara, 2013, citados en Merino Balladares y Aguirre Ponce, 2015).

PHP: Hypertext Preprocessor)

Palomo Duarte (s. f.) señala que

el lenguaje PHP (cuyo nombre es acrónimo de PHP: Hipertext Preprocessor) es un lenguaje interpretado con una sintaxis similar a la de C++ o JAVA. Aunque el lenguaje se puede usar para realizar cualquier tipo de programa, es en la generación dinámica de páginas web donde ha alcanzado su máxima popularidad. En

concreto, suele incluirse incrustado en páginas HTML (o XHTML), siendo el servidor web el encargado de ejecutarlo. (p. 2)

Los lenguajes modernos se orientan mayormente en la web donde primeramente surgió un lenguaje de marcado HTML (HyperText Markup Language) que cada vez se amplió, emergiendo nuevas tecnologías en torno al objetivo de desarrollar aplicaciones en sitios web como actualmente se conoce.

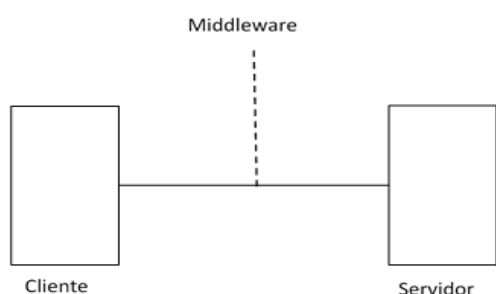


Figura 8. Arquitectura cliente/servidor.

En la actualidad existen diferentes lenguajes de programación para desarrollar aplicaciones web; estos se van presentando debido a las tendencias y necesidades de la sociedad informática y de los diversos sistemas operativos.

Este crecimiento condicionó a los lenguajes de programación a que interactuaran con las bases de datos, exigiéndoles que pasaran de estáticos a dinámicos, independientes del sistema operativo del hardware y del navegador web, manteniendo una navegación amigable y segura y fortaleciendo el concepto de cliente/servidor.

Otra de sus virtudes es su código abierto que le permite generar soluciones y adecuaciones a las necesidades del desarrollador. En la Figura 9 se observa un ejemplo del PHP de un servidor web.

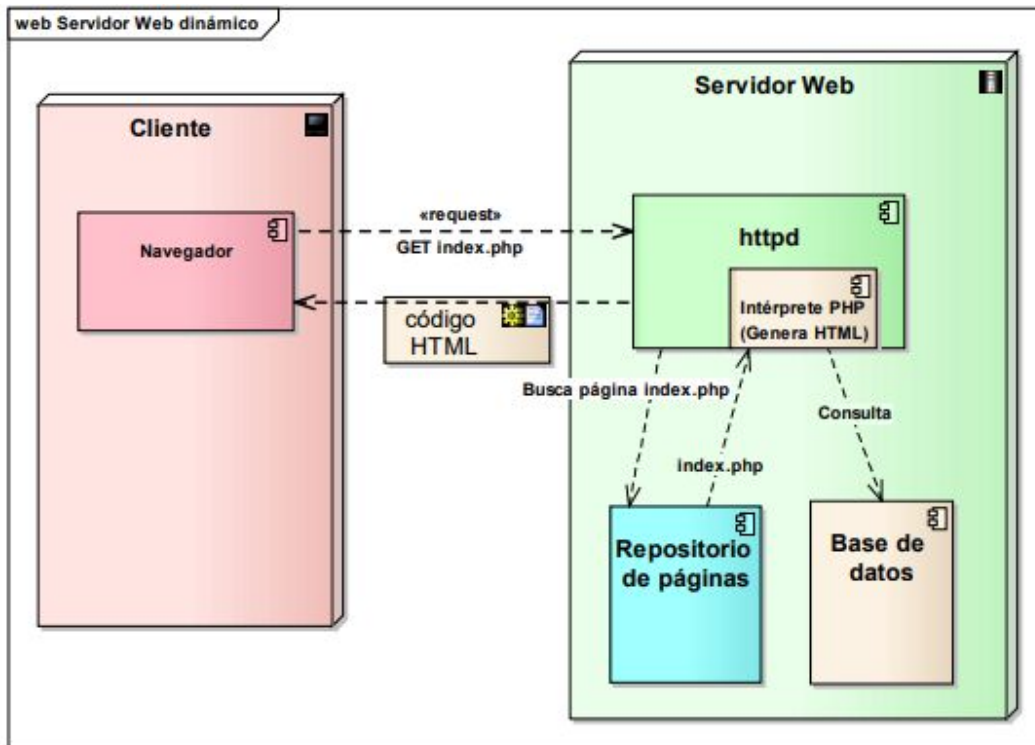


Figura 9. PHP de un servidor web.

CAPITULO III

MARCO METODOLÓGICO

Análisis de requisitos

En el análisis de requisitos se dice que el sistema debe ser capaz de hacer y entender el espacio del problema en términos inequívocos. El usuario y el sistema interactuarán con los siguientes pasos: (a) recolección de requerimientos, (b) personal involucrado, (c) conclusiones después de la entrevista y la recolección de la documentación, (d) procesos del sistema, (e) restricciones, (f) requerimientos técnicos, (g) objetos del dominio y (h) elaboración de prototipos. A continuación se presentan cada uno de ellos.

Recolección de requerimientos

En la recolección de los requerimientos en el desarrollo del sistema web para evaluación docente se utilizó lo siguiente: (a) la observación directa del proceso de evaluación y el sistema académico, (b) la revisión de la documentación y la legislación de la evaluación docente y (c) una entrevista con el director de investigación y calidad educativa, juntamente con el administrador del sistema académico institucional.

Personal involucrado

Dentro del proceso de elaboración del documento de requerimientos intervino

el siguiente personal: la doctora Sara López, directora de calidad educativa y el ingeniero Fernando López, administrador del sistema académico institucional.

Como resultado de la entrevista, se entregó el instrumento de evaluación docente (ver Apéndice A) y el formato donde se entregan los resultados a los docentes (ver Apéndice B).

Conclusiones de la entrevista y la recolección de la documentación

Posteriormente a la observación, la entrevista y la revisión de la documentación del proceso de la evaluación docente, se identificaron los siguientes procesos.

Procesos del sistema

Proceso de evaluación docente

El evaluador asignado por la dirección de calidad educativa entregó al alumno el instrumento de evaluación docente. En una hoja, el alumno apunta la materia que desea evaluar y el valor de su evaluación; al terminar, entrega la evaluación al evaluador. Este proceso se desarrolló manteniendo el anonimato del alumno, pero se tenía que saber quién ya había contestado la evaluación y quién no la había contestado. Fue necesario automatizar, manteniendo y conservando los requerimientos del proceso.

Proceso de entrega de resultados

El proceso de entrega de resultados se describe a continuación:

Entrega de resultados al director de la facultad

Una vez que la dirección de calidad educativa registrara y calculara las evaluaciones, se entregaron al director de cada facultad los resultados de la evaluación de las escuelas que tenían bajo su dirección. El sistema de evaluación distinguió al director de facultad, al director de calidad educativa y al vicerrector académico, quienes tuvieron acceso a estos reportes para analizar los resultados.

Entrega de resultados al director de la escuela

Igualmente, la dirección de calidad educativa entregó al director de la escuela los resultados de la evaluación a los docentes, en un consejo técnico. El sistema de evaluación distinguió al director de facultad, al director de calidad educativa y al vicerrector académico, quienes tenían acceso a estos reportes para analizar los resultados.

Entrega de resultados al docente

El director de la carrera entregó los resultados de la evaluación a los docentes, en un consejo técnico. El sistema de evaluación distinguió al director de facultad, al director de calidad educativa y al vicerrector académico, quienes tienen acceso a estos reportes para analizar los resultados.

Resultados de la evaluación docente

El rector, el vicerrector académico y la dirección de calidad educativa tenían acceso a las evaluaciones aplicadas a todos los docentes en los diferentes periodos y cargas. El sistema de evaluación docente tenía un módulo donde el usuario podía ver los reportes con base en el periodo, la carga y el docente.

Procesos de administración

Dentro de las actividades que se realizaron para organizar la información, se llevaron a cabo las siguientes actividades

Asignación docente-escuela

Cada docente debía pertenecer a una escuela y/o consejo técnico; por lo tanto, se ocupó un proceso donde se le asignó al docente la escuela y/o el consejo técnico al que pertenecía, ya que el sistema institucional no contiene esta relación que se ocupó al momento de mostrar los resultados de las evaluaciones. El sistema tuvo un proceso de asignación docente-escuela exclusivo para el vicerrector académico y/o para la dirección de calidad educativa.

Estatus docente

Cada periodo de clases tuvo docentes que, por motivos de cambio de responsabilidad, de escuela, de renuncia, etc., dejaron de pertenecer a una escuela y/o consejo técnico; por lo cual fue necesario poner el estatus y activar o desactivar a los docentes; esto ayudó a controlar quiénes podían o no entrar al sistema. Este proceso fue exclusivo del vicerrector académico y/o de la dirección de calidad educativa. El sistema tuvo un proceso de asignación del estatus del docente.

Asignar derechos

Se identificó si un usuario era alumno o maestro en el sistema institucional por la matrícula; sin embargo, fue necesario tener un proceso donde se pudiera diferenciar al docentes, al director de escuela, al director de facultad, al director de calidad educativa, al vicerrector académico y al administrador del sistema. Este proceso fue

exclusivo del vicerrector académico y/o de la dirección de calidad educativa.

Editar facultad

Debido a que en el sistema institucional no estaban determinadas las relaciones de facultades, fue necesario tener un proceso donde se pudieran dar de alta y/o modificar las facultades. Este proceso fue exclusivo del vicerrector académico y/o de la dirección de calidad educativa.

Relación escuela-facultad

El proceso de la relación escuela-facultad se asignó a cada escuela y a la facultad a la que pertenecía para la relación de entrega de resultados. Este proceso fue exclusivo del vicerrector académico y/o de la dirección de calidad educativa.

Asignar evaluación

El proceso de asignar evaluación determina las fechas en que se realiza la evaluación docente, en un periodo y ciclo determinados. Este proceso es exclusivo del vicerrector académico y/o de la dirección de calidad educativa.

Proceso de traspaso de datos

Alumnos

El sistema institucional contiene la lista de alumnos, por lo cual es necesario traspasarlos al sistema de evaluación docente.

Maestros

El sistema institucional contiene la lista de maestros, por lo cual es necesario

traspasarlos al sistema de evaluación docente.

Escuelas

El sistema institucional contiene la lista de escuelas, por lo cual es necesario traspasarlos al sistema de evaluación docente.

Materias

El sistema institucional contiene la lista de materias. Este proceso tiene mayor cuidado, ya que una materia pertenece a un plan de estudios y un plan de estudios pertenece a una escuela; por lo cual es necesario traspasarlos al sistema de evaluación docente, respetando estas relaciones escuela-plan de estudios y plan de estudios-materia.

Periodos

El sistema institucional contiene la lista de periodos y carga donde se asignan las materias. Esta relación del sistema de evaluación se exporta para determinar el periodo en que se va a evaluar y se observa la carga del docente.

Asignación de materia-docente

La asignación de materia-docente consiste en una clase que se imparte por el docente, donde se le asignan alumnos, horarios y aula. Esta relación ya se realiza en el sistema institucional, por lo cual solo se debe generar un “script” para el traspaso de la información al modelo que se va a manejar en el sistema de evaluación docente. Esto implica traspasar la lista de docentes, la lista de alumnos y la lista de materias donde se requiere mantener la relación del docente con la escuela a la que pertenece.

Restricciones

En la determinación de los requerimientos, se ha decidido restringir el acceso al sistema de acuerdo con el tipo de usuario. En el proceso de evaluación docente, solo tienen acceso los alumnos; en el proceso de entrega de resultados, tienen acceso los docentes, el director de escuela, el director de facultad, el director de investigación y calidad educativa y el vicerrector académico, cada uno con el alcance y de acuerdo con su responsabilidad. El proceso de administración es exclusivo del vicerrector académico y/o de la dirección de calidad educativa.

Requerimientos técnicos

En la Tabla 1 se observan los requerimientos técnicos para la realización del proyecto, que incluyen las herramientas y las aplicaciones utilizadas.

Tabla 1

Requerimientos técnicos

Herramientas	Aplicaciones
Base de datos	Mysql
Lenguaje de programación	PHP 5
Servidor web	HTTP Apache
Editor de texto y código fuente	Sublime Text 3
Herramienta de modelado	Magic Draw UML
Dominio de la aplicación	edocente.unav.edu.mx
Diseño de base de datos	MySQL Workbench

Objetos del dominio

El modelo del dominio se basa en descubrir los objetos que intervienen en los procesos del sistema que se está desarrollando; estos representan cosas y conceptos del mundo real. En la Figura 10 se puede ver el modelo del dominio con base en la

entrevista, donde se detectan los conceptos principales: docente, alumno, escuela, carga, materia, pregunta, respuesta, periodos, etc.

Elaboración de prototipos

La elaboración de prototipos implica la creación de un modelo operativo de trabajo con un diseño rápido de la interfaz gráfica de un sistema. Los prototipos permiten que los usuarios participen en el desarrollo del sistema, logrando que la definición de requerimientos se simplifique, ayudando a que la probabilidad de rechazo a la implantación disminuya, disminuyendo el tiempo de desarrollo y mejorando la fase de análisis y diseño, lo que lo vuelve un proceso más ágil.

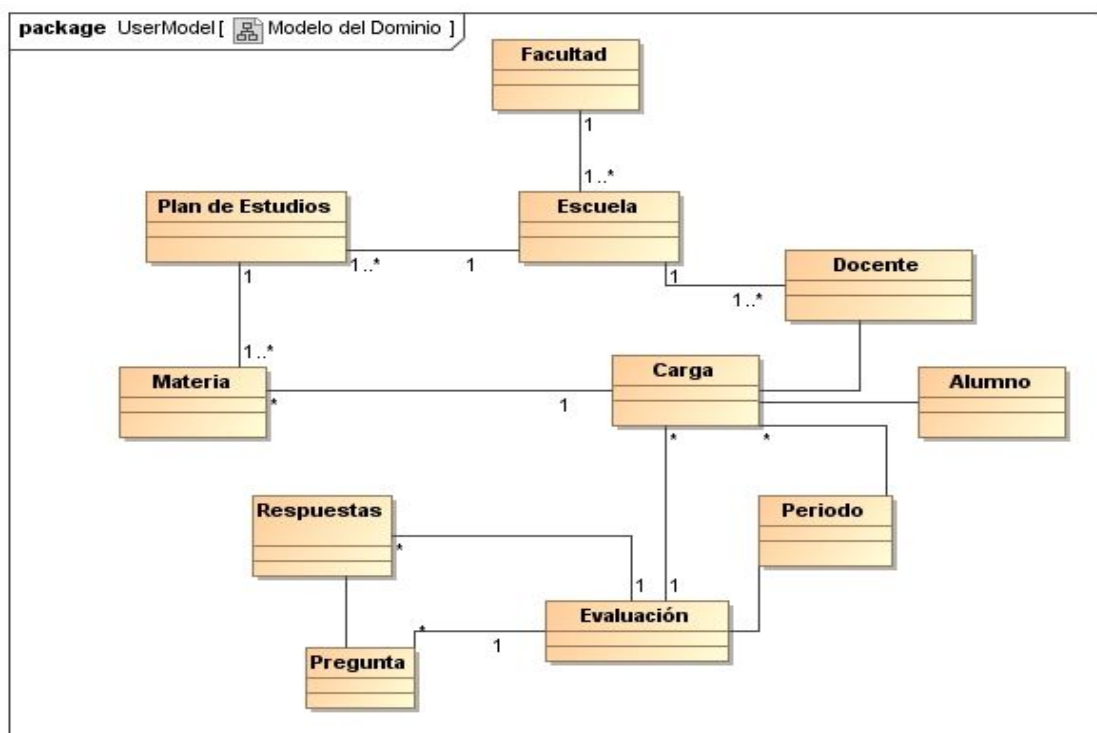


Figura 10. Diagrama de clases del dominio.

Prototipo de validación del usuario

En la Figura 11 se muestra el prototipo de la validación del usuario donde el docente utiliza la matrícula y utilización de una contraseña para poder acceder.

Prototipo de evaluación al docente

En la Figura 12 se muestra la forma como se contesta la evaluación: en cada fila, las preguntas y en las columnas, las materias con sus respectivas opciones para cada respuesta.

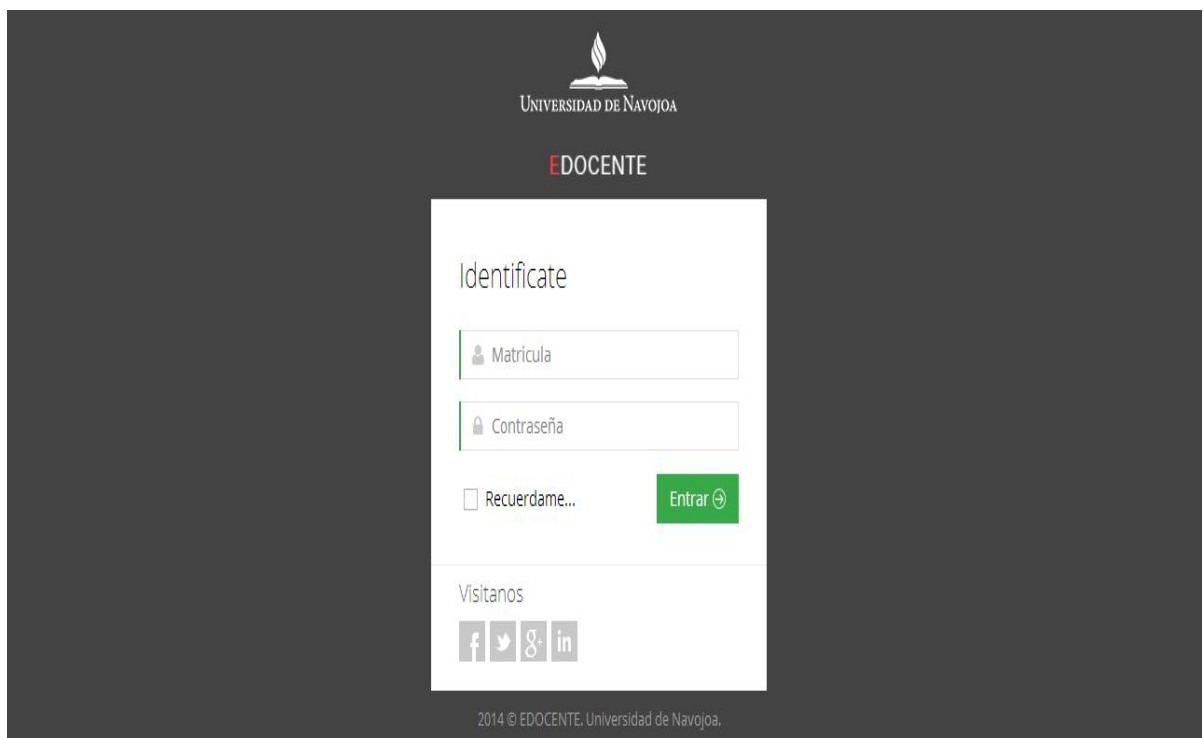


Figura 11. Validación del usuario.

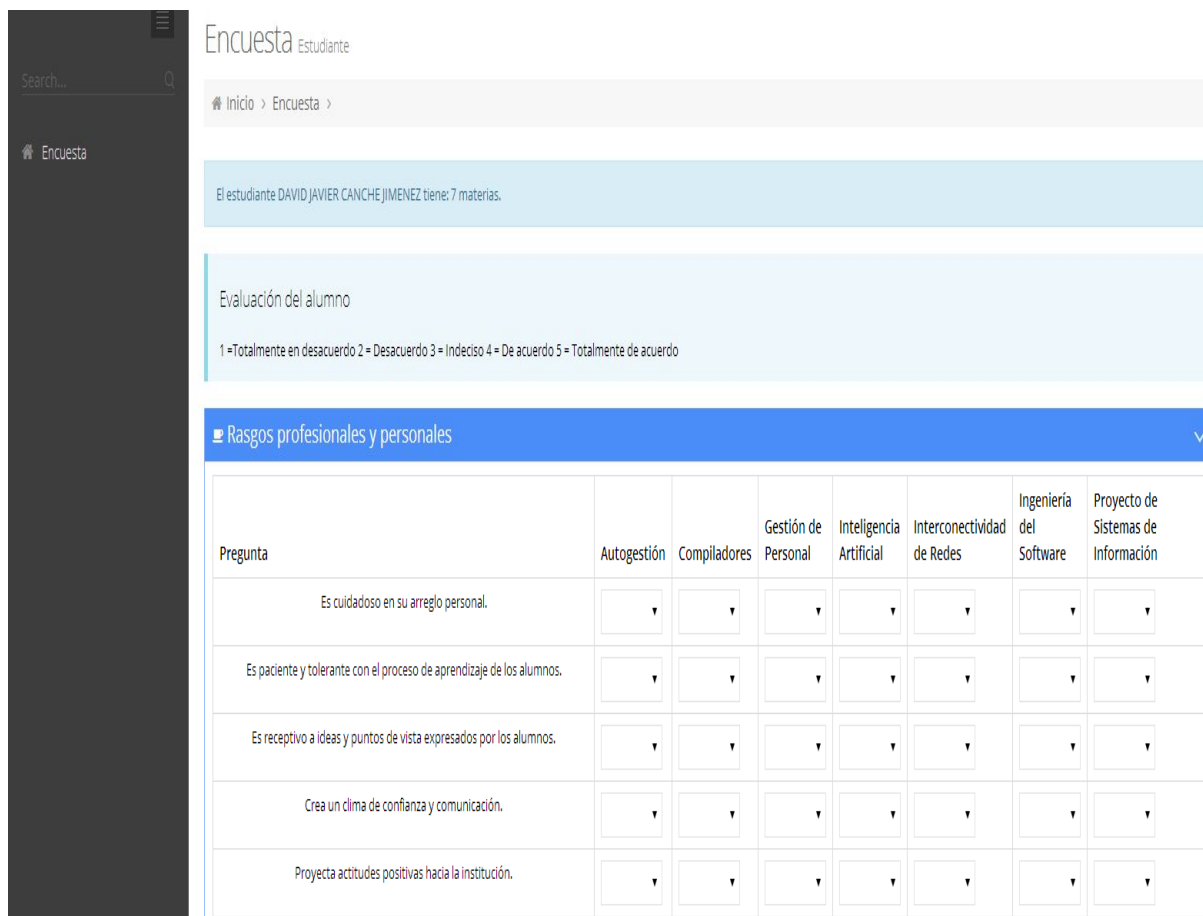


Figura 12. Respuestas de la evaluación.

Prototipo de reporte de evaluación

La Figura 13 muestra el resultado que el docente puede visualizar en base al documento entregado por el departamento de calidad educativa (ver Apéndice C).

Prototipo de reporte del administrador y el director

La Figura 14 muestra la opción de buscar a un docente en específico para ver su evaluación por parte del administrador o director de escuela o facultad.

Pregunta	Promedio	Escuela	Universidad
Es cuidadoso en su arreglo personal.	4.0000	5	5
Es paciente y tolerante con el proceso de aprendizaje de los alumnos.	3.3333	5	5
Es receptivo a ideas y puntos de vista expresados por los alumnos.	3.5000	5	5
Crea un clima de confianza y comunicación.	3.3333	5	5
Proyecta actitudes positivas hacia la institución.	4.0000	5	5
Transmite principios cristianos.	4.0000	5	5
Existe integración de los principios bíblicos con los contenidos del curso que imparte.	4.0000	5	5
Establece claramente y en acuerdo con los alumnos, las normas de convivencia, sustentadas en la filosofía institucional.	4.0000	5	5

Figura 13. Resultado de la evaluación.

Matriculación	Nombre	Paterno	Materno	Editar
9800003	AMADO VÍCTOR	FRAGOSO	VÁZQUEZ	Seleccionar
9800021	VÍCTOR MARTÍN	CAMPOY	FERNÁNDEZ	Seleccionar
9800130	VICTOR HUGO	SOLIS	VALLADARES	Seleccionar
9800132	VICTOR	MONARREZ	PEREZ	Seleccionar
9800223	VICTOR	MATUS	SANCHEZ	Seleccionar

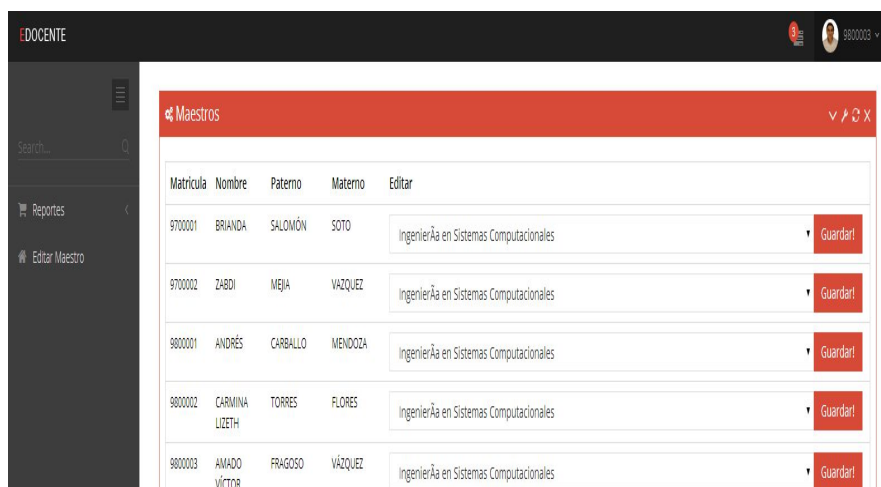
Figura 14. Búsqueda del docente.

Prototipo de reporte de asignación docente-escuela

Cada maestro que labora en la universidad pertenece al consejo técnico de una escuela. En la Figura 15 se muestra la pantalla donde se asigna al docente la escuela a la que pertenece.

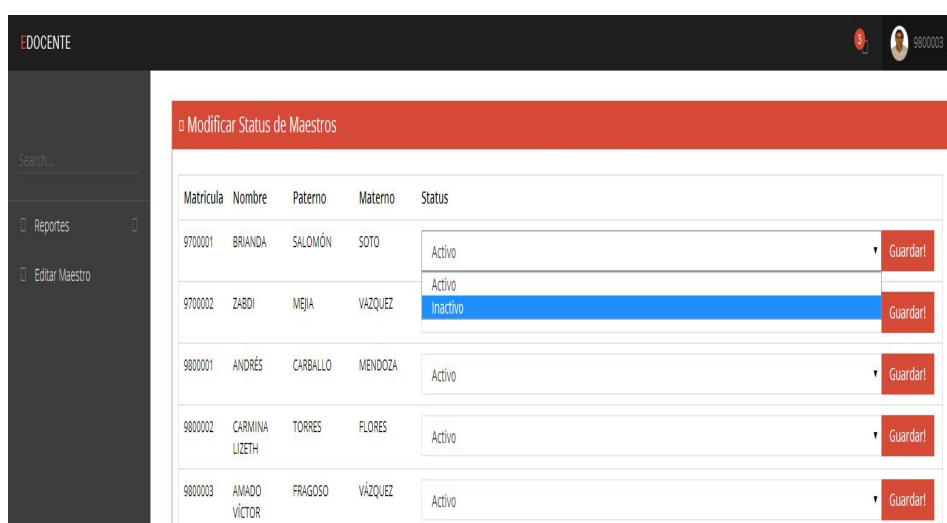
Prototipo de estatus docente

Los maestros pueden quedar activos o inactivos para la docencia y en la Figura 16 se muestra cómo modificar su estatus. En esta pantalla se deja inactivo a un docente, el cual ya no podrá pertenecer a ningún consejo técnico.



Matricula	Nombre	Paterno	Materno	Editar
9700001	BRIANDA	SALOMÓN	SOTO	Ingeniería en Sistemas Computacionales Guardar!
9700002	ZABDI	MEJIA	VAZQUEZ	Ingeniería en Sistemas Computacionales Guardar!
9800001	ANDRÉS	CARBALLO	MENDOZA	Ingeniería en Sistemas Computacionales Guardar!
9800002	CARMINA LIZETH	TORRES	FLORES	Ingeniería en Sistemas Computacionales Guardar!
9800003	AMADO VÍCTOR	FRAGOSO	VÁZQUEZ	Ingeniería en Sistemas Computacionales Guardar!

Figura 15. Asignación docente-escuela.



Matricula	Nombre	Paterno	Materno	Status
9700001	BRIANDA	SALOMÓN	SOTO	Activo Guardar!
9700002	ZABDI	MEJIA	VAZQUEZ	Inactivo Guardar!
9800001	ANDRÉS	CARBALLO	MENDOZA	Activo Guardar!
9800002	CARMINA LIZETH	TORRES	FLORES	Activo Guardar!
9800003	AMADO VÍCTOR	FRAGOSO	VÁZQUEZ	Activo Guardar!

Figura 16. Estatus del docente.

Prototipo de asignación de derechos

Cada docente puede tener acceso al sistema con base en los actores ya definidos. En la Figura 17 se muestra cómo asignarle privilegios.

Prototipo de editar facultad

Cada escuela pertenece a una facultad. La Figura 18 muestra cómo se dan de alta.

The screenshot shows the 'Asignar derechos a usuario' (Assign rights to user) form in the EDOCENTE system. The form is displayed within a dark sidebar navigation menu. The sidebar contains a search bar and two menu items: 'Reportes' and 'Editar Maestro'. The main content area has a red header with the title 'Asignar derechos a usuario'. Below the header, there is a table with two columns: 'Concepto' and 'Seleccionar'. The first row has 'MAESTRO' in the 'Concepto' column and a dropdown menu with '9801' selected in the 'Seleccionar' column. The second row has 'PERMISO' in the 'Concepto' column and a dropdown menu with 'DOC' selected in the 'Seleccionar' column. A red 'Guardar!' button is located at the bottom left of the form. A dropdown menu is open for the 'PERMISO' row, showing the following options: 'DOCENTE' (highlighted in blue), 'ADMINISTRADOR', 'DIRECTOR DE FACULTAD', and 'DIRECTOR DE ESCUELA'. The top right of the interface shows the user's profile with a notification icon (3), a profile picture, and the ID '9800003'.

Figura 17. Asignación de derechos del usuario.

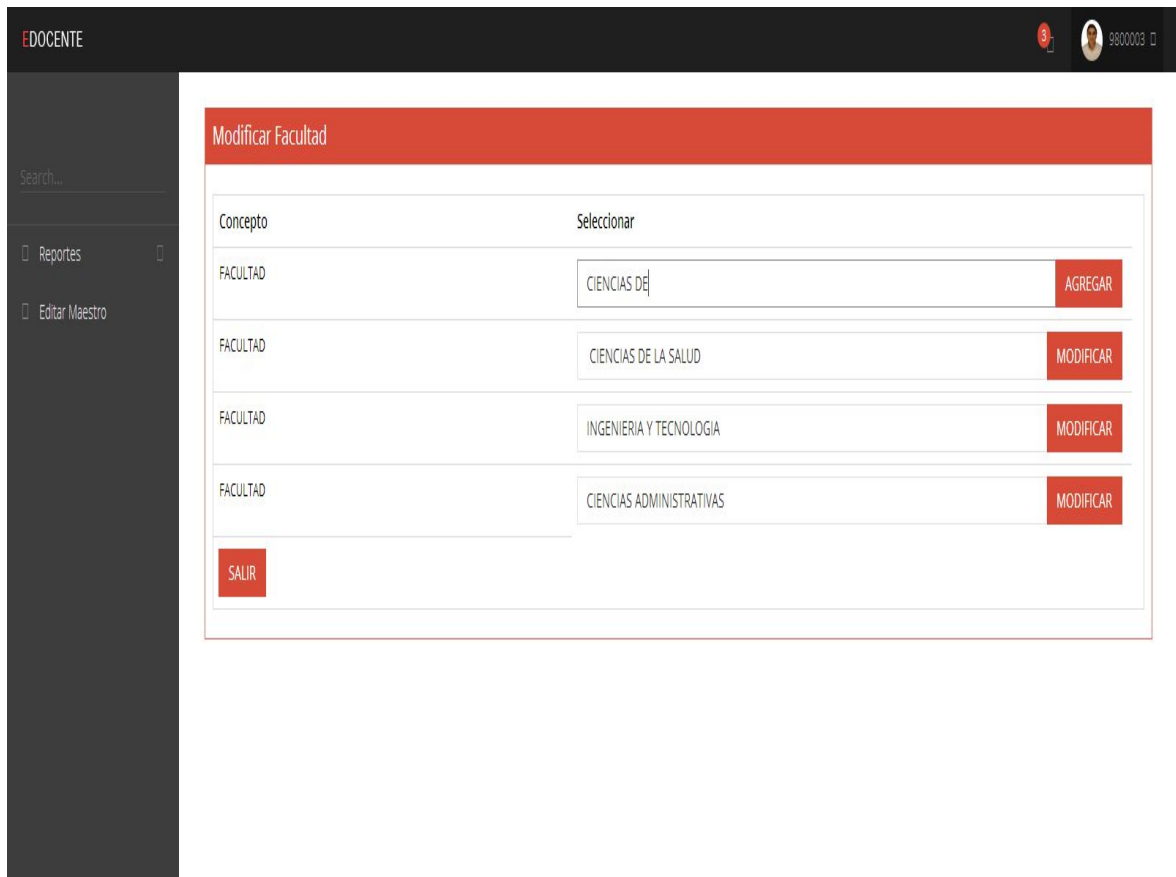


Figura 18. Edición de la facultad.

Prototipo de relación escuela-facultad

Las escuelas deben pertenecer a una facultad y en la Figura 19 se ve cómo se realiza este proceso.

Prototipo de asignación de evaluación

El administrador puede definir las fechas en que se va a aplicar la evaluación en un periodo determinado, como se observa en la Figura 20.

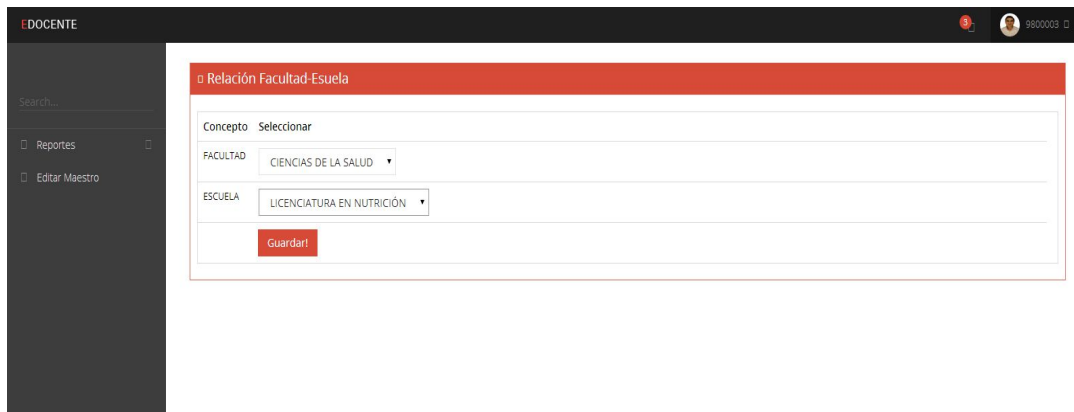


Figura 19. Asignación de facultad-escuela.

Modelo de casos de uso

El modelo de casos de uso comprende la identificación de los actores y el diagrama de los casos de uso; esto es para que el usuario vea la interacción con el sistema.

Actores

Un actor es el rol que un usuario juega con respecto al sistema. El uso de la palabra rol especifica que un actor no necesariamente representa a una persona física, sino más bien a la labor que realiza frente al sistema (Jalloul, 2004).

Se pueden ver en la Tabla 2 a los actores del sistema de evaluación docente, con base en la entrevista y observación del proceso. En general, se detectaron cinco diferentes clases de actores, que tendrán diferente nivel de acceso al sistema y la descripción del rol que cumplen.

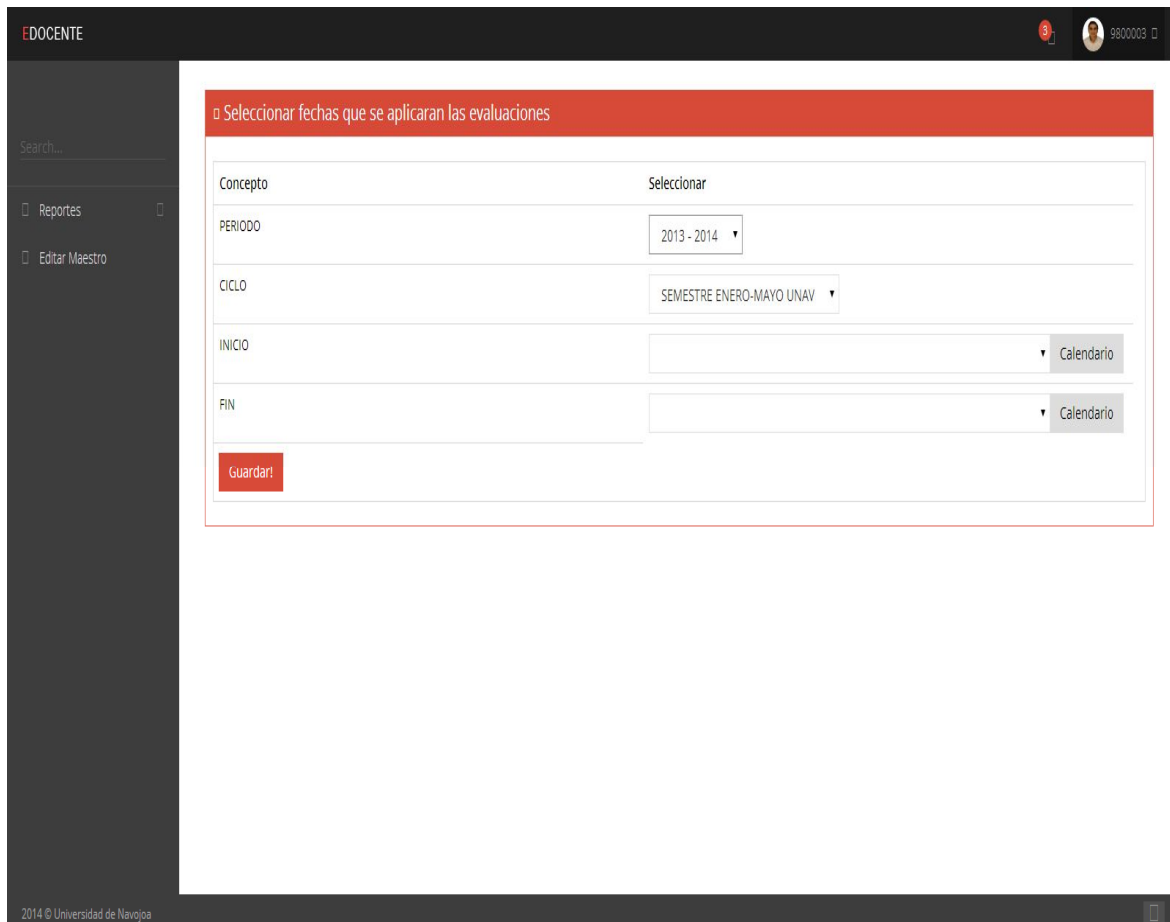


Figura 20. Asignación de la evaluación.

Tabla 2

Actores del sistema

Actor	Descripción
Usuario	Es quien se valida antes de entrar al sistema necesita tener una matrícula y una contraseña
Administrador	Es el usuario que tiene derecho de administrador del sistema
Director de facultad	Es el usuario que tiene derechos de acceder a más de una escuela
Director de escuela	Es el usuario que tiene derecho de acceder solo una escuela
Docente	Es el usuario que solamente podrá revisar sus resultados

Diagrama de caso de uso general

El diagrama de caso de uso que permite ver gráficamente los límites, acciones y la interacción con los actores de la aplicación es aquel donde el cliente puede observar el comportamiento del sistema. Se compone de casos de uso, actores y relaciones (Campderrich Falgueras, Ingeniería del software, 2002).

En la Figura 21 se describe el diagrama de casos de uso del sistema y su interacción con los diferentes actores del sistema, mostrando los derechos que tendrán.

La descripción de cada caso de uso se describe en el análisis; con esto, el cliente tiene ya una visión de lo que el sistema hará en esta iteración y lo que no se realizará. Es importante que se clarifique el alcance y se pueda empezar a visualizar lo que se puede implementar en iteraciones posteriores.

En el Apéndice D se observan los casos de uso por actor y los casos de uso por modulo que complementan la comprensión de los procesos que el sistema atiende de forma visual y aclara la visión del cliente.

Análisis y diseño preliminar

Descripción de casos de uso

En la Tabla 3 se puede ver el formato de descripción de casos de uso, donde se pone el nombre, resumen del caso de uso, el actor principal, precondiciones que están formadas por el conjunto de condiciones que tienen que cumplirse para que se pueda iniciar un caso de uso; flujo principal, que describe los casos normales del proceso; flujo alternativo, donde se explica qué sucede cuando algo causa una desviación desde el flujo principal y por condiciones que reflejan el estado en que se queda el sistema una vez ejecutado el caso de uso.

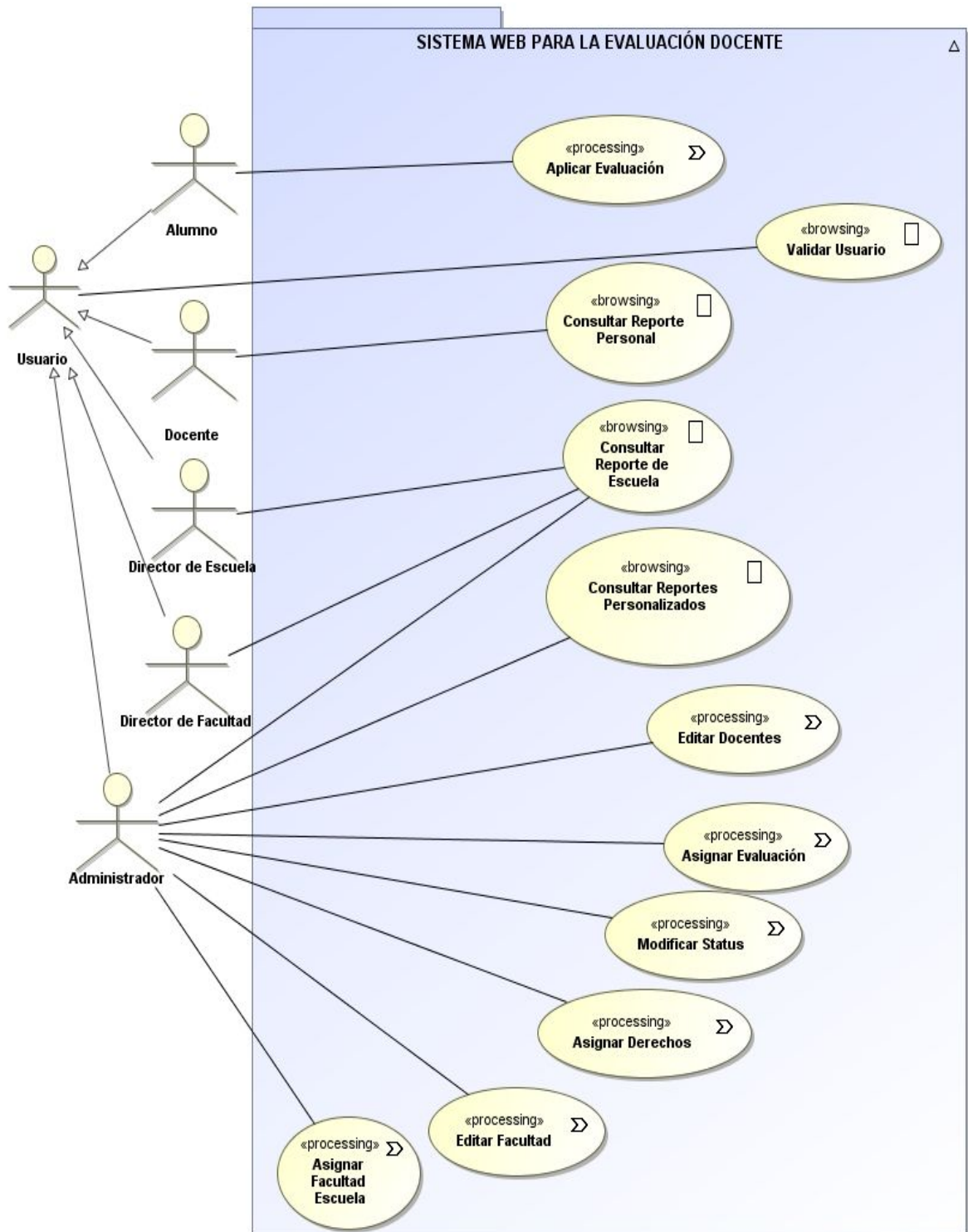


Figura 21. Diagrama de casos de uso.

Tabla 3

Descripción de casos de uso

Caso de uso	
Resumen	
Actor principal	
Precondiciones	
Flujo principal	
Flujo Alternativo	
Por condiciones	

A continuación se analizan los casos de uso.

Validar usuario

Este caso de uso valida al usuario para poder tener acceso al sistema y determinar los permisos que se le asignan a cada usuario (ver Apéndice E), siguiendo los siguientes pasos: (a) el usuario entra a la página web edocente.unav.edu.mx, (b) el sistema le muestra la página de validación, (c) el usuario ingresa su matrícula y la contraseña y (d) el sistema valida si el usuario y contraseña son correctos; entra al sistema con los permisos a los que tiene derecho y acaba el caso de uso.

Aplicar evaluación

El alumno llena la evaluación de las materias que llevó en el semestre (ver Apéndice F), descrito a continuación: (a) el usuario escoge en el menú aplicar evaluación, (b) el sistema verifica si tiene evaluaciones disponibles y le muestra las evaluaciones a

que tiene acceso, (c) el usuario elige la evaluación a contestar, (d) el Sistema le muestra la evaluación, (e) el usuario llena la evaluación y, al terminar oprime enviar y (f) el sistema verifica que todas las respuestas se hayan llenado, guarda los resultados de la evaluación y muestra un mensaje de gracias por llenar la evaluación.

Consultar reporte personal

El docente consulta el reporte de su evaluación en el sistema (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú reportes por maestro, (b) el sistema verifica en qué periodos tiene evaluaciones y le muestra las evaluaciones disponibles, (c) el usuario elige la evaluación a consultar y (d) el sistema le muestra la evaluación.

Consultar reporte de escuela

El director de escuela consulta el reporte de la evaluación de los docentes de su carrera (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú reportes por escuela, (b) el sistema verifica los docentes que pertenecen a su escuela y los muestra, (c) el usuario elige al docente a consultar, (d) el sistema verifica en qué periodos tiene evaluaciones y le muestra las evaluaciones disponibles, (e) el usuario elige la evaluación a consultar y (f) el sistema le muestra la evaluación.

Consultar reportes personalizados

El administrador consulta el reporte de la evaluación de los docentes de su carrera (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú reportes personalizados, (b), el sistema muestra una opción de búsqueda, (c) el usuario

escribe al docente a consultar, (d) el sistema verifica en qué periodos tiene evaluaciones y le muestra las evaluaciones disponibles, (e) el usuario elige la evaluación a consultar y (f) el sistema le muestra la evaluación.

Editar docentes

El administrador modifica al docente a la escuela a la que pertenece (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú administración editar docente, (b) el sistema muestra la lista de docentes activos y la escuela a la que pertenecen, (c) el usuario modifica la escuela a la que pertenece el docente y oprime guardar y (d) el sistema guarda el cambio de la escuela asignada al docente.

Modificar estatus

El administrador modifica el estatus de un docente (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú administración estatus, (b) el sistema muestra la lista de docentes, (c) el usuario modifica el estatus del docente y oprime guardar y (d) el sistema guarda el cambio de estatus del docente.

Editar facultad

El administrador modifica o agrega una facultad (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú administración facultad, (b) el sistema muestra la lista de facultades y, al final, agregar una nueva, (c) el usuario modifica una escuela y oprime modificar y (d) el sistema guarda las modificaciones.

Asignar facultad-escuela

El administrador asigna relación facultad escuela (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú administración facultad-escuela, (b) el sistema muestra la lista de escuelas y al frente las facultades, (c) el usuario modifica la facultad a la que pertenece una escuela y oprime guardar y (d) el sistema guarda las modificaciones.

Asignar derechos

El administrador asigna los derechos a los usuarios (ver Apéndice A), como se describe a continuación: (a) el usuario escoge en el menú administración derechos, (b) el sistema muestra una página donde se introduce la matrícula a que se le modificarán los derechos, (c) el usuario introduce la matrícula a modificar derechos, (d) el sistema muestra los derechos del sistema, (e) el usuario selecciona los derechos del usuario y (f) el sistema guarda las modificaciones.

Asignar evaluación

El administrador asigna las fechas que se aplicará una evaluación (ver Apéndice A), descrito a continuación: (a) el usuario escoge en el menú administración asignar evaluación, (b) el sistema muestra los periodos disponibles, (c) el usuario determina las fechas de aplicación de la evaluación y (d) el sistema guarda las modificaciones.

Diagrama de robustez

Los diagramas de robustez muestran las interacciones entre los objetos que participan en los casos de uso; sirven para saber si las especificaciones del sistema son razonables. Al ser una metodología de desarrollo semi-ágil, no hay que olvidar que

se debe actualizar el diagrama de clases definido en el modelo de dominio con las clases y atributos encontrados en los diagramas de robustez.

Los diagramas de robustez se usan en las siguientes funciones: (a) validar usuario, (b) aplicar evaluación, consultar reporte personal, (c) consultar reporte de escuela, (d) consultar reportes personalizados, (e) editar docentes, (f) modificar estatus, (g) editar facultad, (h) asignar facultad-escuela, (i) asignar derechos y (j) asignar evaluación (ver Apéndice E), donde viene definida cada interacción con los usuarios, los objetos fronterizos, la entidad y el control (ver Figura 22).



Figura 22. Elementos del diagrama de robustez.

Implementación

La implementación fue definida en los diagramas como el orden que llevará la estructura de la aplicación. Se espera que desarrolle la aplicación y sus especificaciones; esto implica un conjunto de pasos a seguir para llegar a la solución de un problema, con una serie de herramientas y la codificación del código. En la Figura 23 se muestra el diagrama de la base de datos como resultado del análisis y diseño.

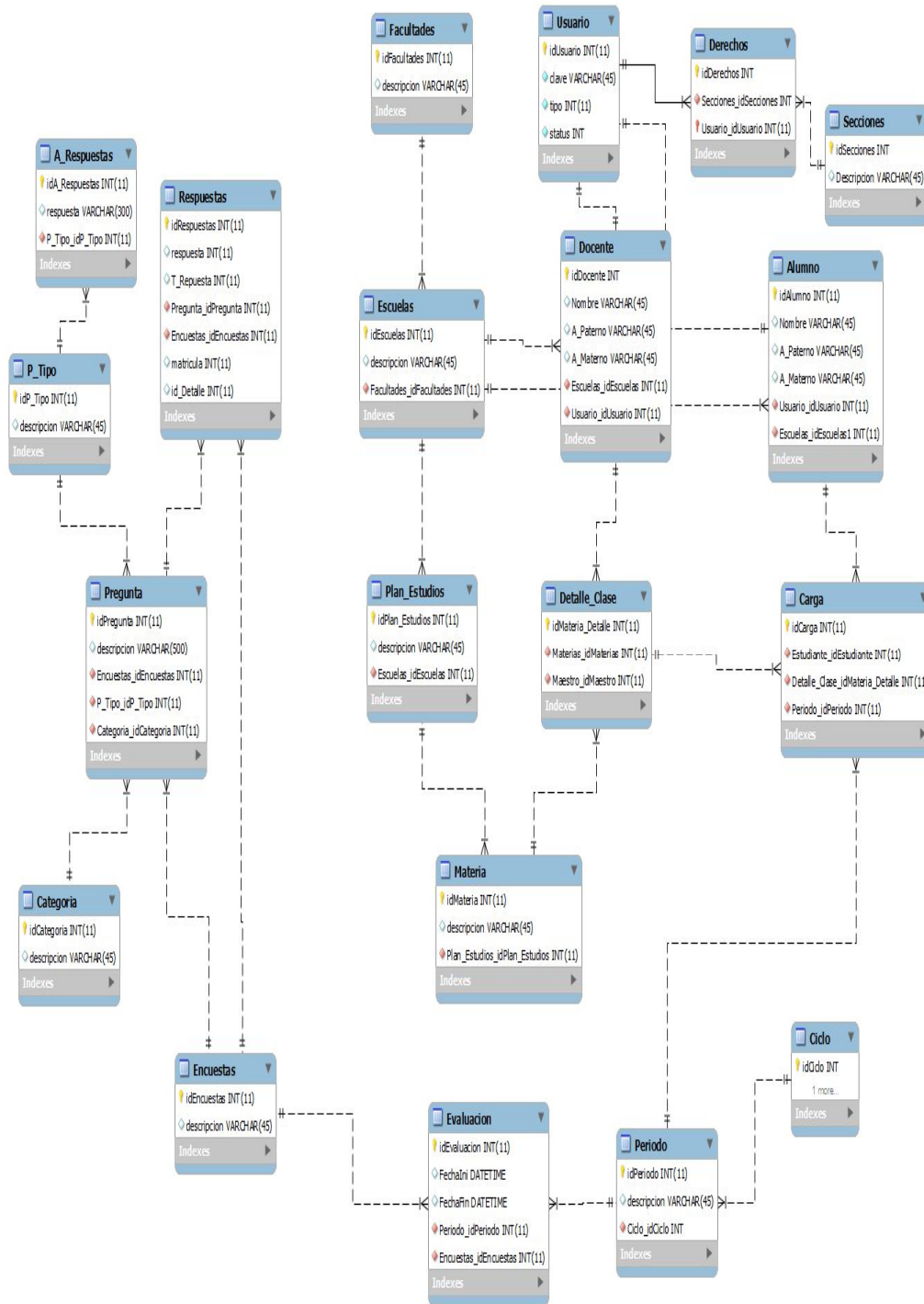


Figura 23. Diagrama ER de la base de datos.

Herramientas

Base de datos

MySQL es de las base de datos más utilizada en el desarrollo de aplicaciones web. Entre sus principales características está que es relacional, multihilo, multiusuario y embebida. La base de datos es de código abierto, como se puede ver en la implementación de la base de datos de la aplicación (ver Figura 24).



Idioma: Español MySQL » daldb.vpsdime.com » Base de datos: serv1_edocente

Adminer 4.1.0 Base de datos: serv1_edocente

DB: serv1_edocente [Modificar Base de datos](#) [Esquema de base de datos](#) [Privilegios](#)

[Comando SQL](#) [Importar](#) [Export](#) [Crear tabla](#)

registros A_Respuestas
registros Carga
registros Categorías
registros Contestados
registros Detalle_Clase
registros Encuestas
registros Escuelas
registros Estudiante
registros Facultades
registros grupos
registros Licenciaturas
registros Maestro
registros Materias
registros Periodo
registros Plan_Estudios
registros Pregunta
registros P_Tipo
registros relacion_docente
registros Respuestas
registros Usuario

Tablas y vistas

Buscar datos en tablas (20)

Tabla	Motor	Colación	Longitud de datos	Longitud de índice	Espacio libre	Incremento automático	Registros	Comentario
<input type="checkbox"/> A_Respuestas	InnoDB	latin1_swedish_ci	16 384	0	0	1	0	
<input type="checkbox"/> Carga	InnoDB	utf8_general_ci	131 072	163 840	0	2 324	~ 2 323	
<input type="checkbox"/> Categorías	InnoDB	latin1_swedish_ci	16 384	0	0	4	~ 2	
<input type="checkbox"/> Contestados	InnoDB	latin1_swedish_ci	16 384	0	0	60	~ 28	
<input type="checkbox"/> Detalle_Clase	InnoDB	latin1_swedish_ci	65 536	0	0		~ 359	
<input type="checkbox"/> Encuestas	InnoDB	latin1_swedish_ci	16 384	0	0	2	0	
<input type="checkbox"/> Escuelas	InnoDB	utf8_general_ci	16 384	0	0	21 403	~ 42	
<input type="checkbox"/> Estudiante	InnoDB	utf8_general_ci	49 152	0	0		~ 300	
<input type="checkbox"/> Facultades	InnoDB	utf8_general_ci	16 384	0	0	1	0	
<input type="checkbox"/> grupos	InnoDB	latin1_swedish_ci	16 384	32 768	0		~ 2	
<input type="checkbox"/> Licenciaturas	InnoDB	latin1_swedish_ci	16 384	0	0		0	
<input type="checkbox"/> Maestro	InnoDB	utf8_general_ci	16 384	0	0		~ 242	
<input type="checkbox"/> Materias	InnoDB	utf8_general_ci	16 384	16 384	0	1	0	
<input type="checkbox"/> Periodo	InnoDB	utf8_general_ci	16 384	0	0	1	0	
<input type="checkbox"/> Plan_Estudios	InnoDB	utf8_general_ci	16 384	0	0	1	0	
<input type="checkbox"/> Pregunta	InnoDB	latin1_swedish_ci	16 384	32 768	0	55	~ 45	
<input type="checkbox"/> P_Tipo	InnoDB	latin1_swedish_ci	16 384	0	0	2	0	
<input type="checkbox"/> relacion_docente	InnoDB	latin1_swedish_ci	16 384	0	0	49	~ 47	
<input type="checkbox"/> Respuestas	InnoDB	latin1_swedish_ci	1 589 248	622 592	4 194 304	20 161	~ 15 499	
<input type="checkbox"/> Usuario	InnoDB	latin1_swedish_ci	294 912	0	0		~ 3 896	
20 en total	InnoDB	latin1_spanish_ci	2 375 680	868 352	0			

Figura 24. Base de datos en Mysql.

Editor de código

Se ha utilizado Sublime text, editor de código multiplataforma, que permite de manera dinámica la codificación, con un entorno que deja programar sin distracciones y colores que ayudan a concentrarse en el desarrollo, además de poder tener varios documentos abiertos mediante pestañas, e incluso emplear varios paneles para cuando se utilice más de un monitor, permitiendo trabajar de forma simultánea con otros programadores conectados directamente a la aplicación en un servidor remoto a través de sftp. Es una herramienta de código abierto y en la Figura 25 se puede ver el entorno de la aplicación.

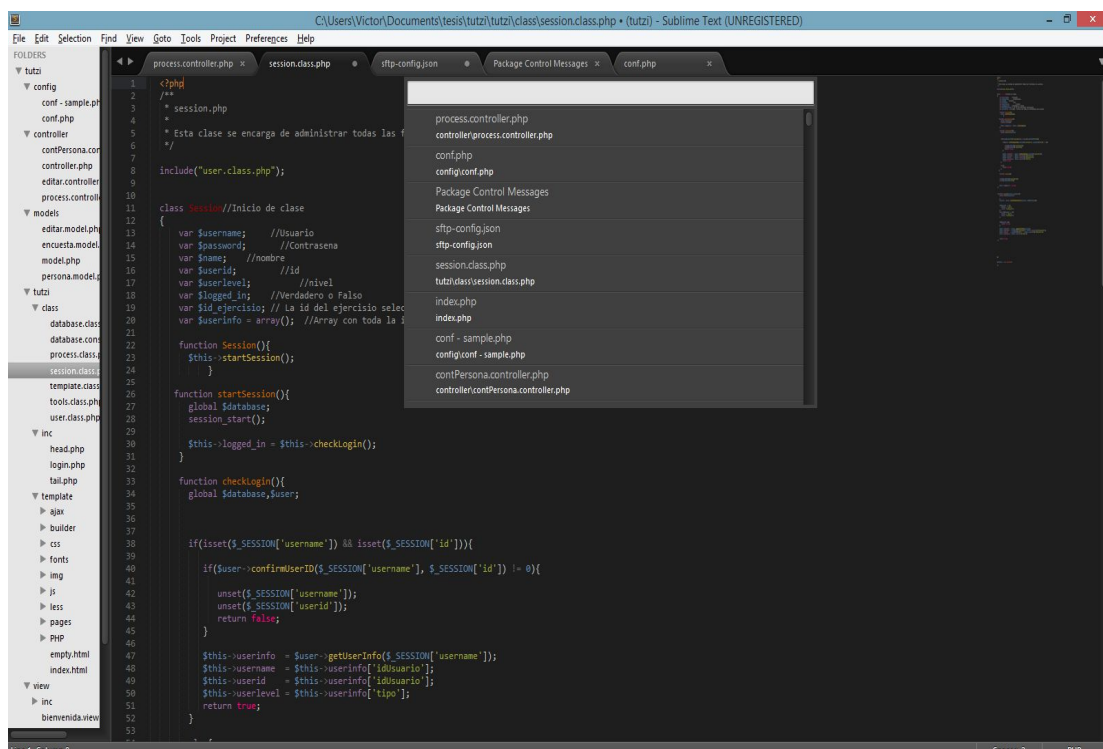


Figura 25. Entorno de desarrollo en Sublime Text.

Servidor http apache

El servidor HTTP Apache es el servidor que se utiliza en la aplicación, ya que es un servidor web HTTP de código abierto que ayuda a mantenerse en constante mantenimiento, atendiendo a las nuevas especificaciones y vulnerabilidades que se descubren. Es utilizado en el 52% de los sitios web activos en internet, de acuerdo con la netcraft, como se muestra en la Figura 26. El servidor HTTP apache es usado principalmente para enviar páginas web estáticas y dinámicas en la world wide web. Permite establecer distintos niveles de control de acceso a la información incluyendo el soporte acifrado SSL y utilizando protocolo seguro HTTPS.



Developer	May 2014	Percent	June 2014	Percent	Change
Apache	533,232	53.32%	519,227	51.92%	-1.40
nginx	181,612	18.16%	192,142	19.21%	1.05
Microsoft	123,823	12.38%	124,487	12.45%	0.07
Google	29,238	2.92%	26,833	2.68%	-0.24

Figura 26. Cuota de mercado sitios web de mayor actividad.

Escribir y generar código

Para la codificación de la aplicación web en la evaluación docente se muestra su estructura en la Figura 27. Se utilizó la plantilla de la parte del cliente AdminLTE.

Para mejor integración con el cliente (ver Figura 28), se emplea la licencia open source; entre los derechos que tiene, está el poder de utilizar, copiar, modificar y fusionar la plantilla. El código fuente se encuentra en el Apéndice F, donde se muestra la implementación de este.

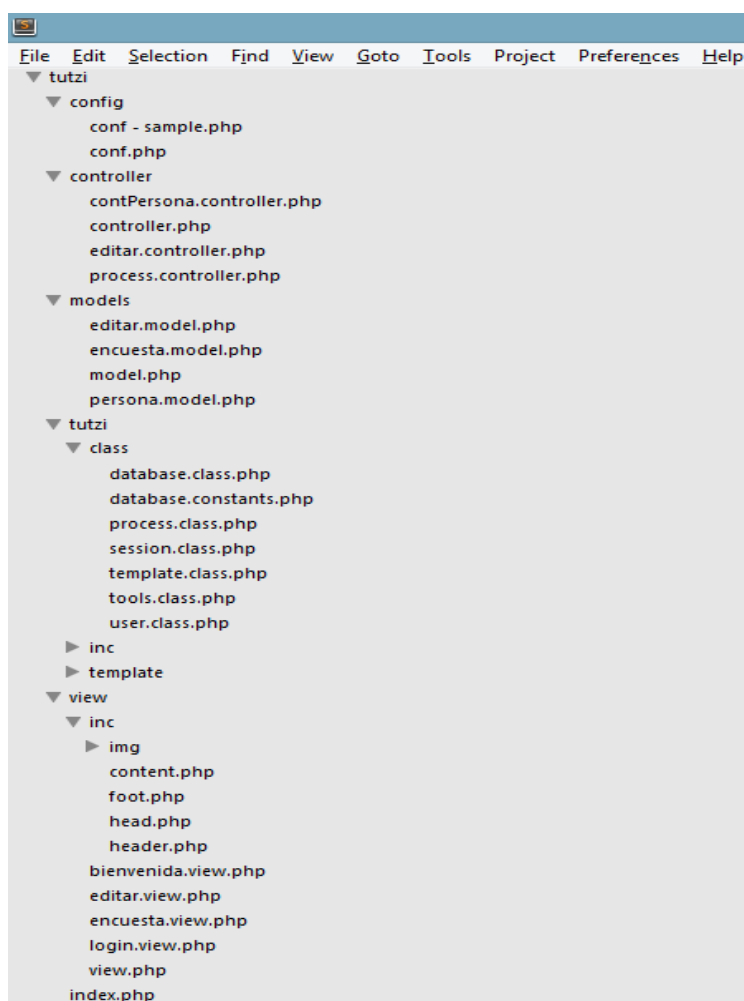


Figura 27. Estructura de la aplicación web.

GitHub This repository Search or type a command Explore Features Enterprise Blog Sign up Sign in

PUBLIC almasaeed2010 / AdminLTE 1,900 Stars 502 Forks

AdminLTE - Free Premium Admin control Panel Theme That Is Based On Bootstrap 3.x
<http://www.almasaeedstudio.com>

48 commits 2 branches 6 releases 7 contributors

branch: master AdminLTE / +

Merge pull request #73 from joey-di-nardo/patch-1 latest commit 5a659ff5d6

ajax	Redesign	6 months ago
css	Remove .DS_Store files and add to gitignore	6 hours ago
fonts	Redesign	6 months ago
img	Remove .DS_Store files and add to gitignore	6 hours ago
js	Merge pull request #73 from joey-di-nardo/patch-1	3 hours ago
less	Merge pull request #46 from seven1m/master	25 days ago
pages	Fixed demo.js link errors	18 days ago
.gitignore	Remove .DS_Store files and add to gitignore	6 hours ago
LICENSE	Initial commit	7 months ago
README.md	Update README.md	3 days ago
empty.html	Updating to version 1.2	4 months ago
index.html	split demo functions to different javascript file	3 months ago

Code Issues (3) Pull Requests (0) Wiki Pulse Graphs Network

HTTPS clone URL
<https://github.com/almasaeed2010/AdminLTE>

You can clone with HTTPS or Subversion.

Clone in Desktop Download ZIP

README.md

Figura 28. GitHub de AdminLTE.

CAPÍTULO IV

ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

Introducción

La tarea de realimentación y crecimiento de las instituciones universitarias exige que los procesos sean lo más rápidos y eficientes posible para la toma de decisiones y la aplicación de estrategias que las llevarán al éxito. La Universidad de Navojoa se encuentra ante la obligación de mejorar hacia la dirección correcta. Ese motivo llevó al investigador a ayudar a mejorar el proceso de evaluación docente asignado al Departamento de Calidad Educativa, por medio de una aplicación web.

Acceso

El acceso desde cualquier equipo conectado a internet facilita el proceso de evaluación, alcanzando a estudiantes que, por motivos de su plan de estudios o de salud, tienen que ausentarse de la institución en las fechas indicadas para esta tarea.

Automatización

El desarrollo de la aplicación Web para la evaluación docente logró mejorar la optimización del proceso del departamento de calidad educativa, ya que antes este proceso era muy tardado. Ahora no se tuvieron que repartir hojas a los estudiantes y pedirles, que evaluaran a los docentes, sin saber si evaluaban todas las materias, con respuestas completas de los docentes que les dan clases. En la aplicación web solo

aparecen las materias que tienen registradas y no pueden salir de la evaluación si no han terminado de contestar las preguntas.

Satisfacción

El anonimato del alumno está garantizado en la aplicación de la evaluación, ya que en el proceso el departamento de calidad educativa y los docentes no pueden saber lo que contestó cada alumno en su evaluación. Aunque no se sabe la respuesta de un alumno en específico, sí se puede determinar cuántos alumnos no han respondido a la encuesta, dando la posibilidad de solicitar una mejor participación.

Resultados

Los resultados de la evaluación son automáticos, por lo que el docente puede conocer los puntos que necesita mejorar para una mayor satisfacción del alumno, planificar bien sus clases y saber a qué necesita dar mayor hincapié en el desarrollo de su clase. El tener los resultados en una base de datos ayuda a poder realizar análisis más profundos y detallados de los mismos, para la toma de decisiones.

CAPÍTULO V

DISCUSIÓN DE RESULTADOS

Introducción

De acuerdo con los objetivos planteados inicialmente, la conclusión de este trabajo de la automatización en una aplicación web para el proceso de evaluación docente garantiza que la información que se está mostrando es de total confiabilidad.

La metodología y los diagramas que se utilizaron en el desarrollo de la aplicación dieron la flexibilidad y las características principales para satisfacer las necesidades presentes y futuras, demostrando verdaderamente que el sistema cumple con lo que en realidad estaba planeado, con una mayor aceptación.

Conclusiones

Las conclusiones obtenidas en el desarrollo de este proyecto de investigación fueron las siguientes:

1. La tecnología ayuda a que los procesos de las organizaciones puedan agilizarse y reducir los tiempos de aplicación y ejecución. En el presente proyecto se mejoró el proceso de evaluación.

2. El uso de una metodología ayuda a una mejor documentación y comprensión del funcionamiento de la aplicación web, facilitando el crecimiento y la adaptación y reduciendo los errores y costos de modificación de los requerimientos.

3. Los diagramas de la base de datos dan la posibilidad de migrar o crecer con Mysql, ya que gracias a la reducción en el tiempo de procesamiento de la ejecución de diferentes procesos, permite la obtención de mejores resultados.

4. El uso de tecnologías abiertas ayuda a reducir los costos sin sacrificar el rendimiento, ya que se usaron en la implementación herramientas comprobadas y utilizadas en el mercado, sin limitar el desarrollo de la aplicación.

5. El desarrollo de una aplicación web abre la posibilidad de poder realizar la evaluación en cualquier equipo con acceso a internet, garantizando la libertad y el anonimato del alumno.

6. Los resultados se pueden consultar de manera instantánea, ya que el proceso de consulta es automático, reduciendo y optimizando el proceso de la evaluación dirigida por el departamento de calidad educativa.

7. El PHP es un lenguaje que está en constante evolución y documentación, garantizando que en futuras ampliaciones se facilite el crecimiento de la aplicación de evaluación e integración de nuevos módulos.

8. El modelo MVC ayuda a la seguridad de los datos y a la flexibilidad de la aplicación.

Trabajos futuros

El término de esta iteración ayuda a visionar un crecimiento en la aplicación de la evaluación docente. La tecnología que se utilizó y la diagramación le dan la posibilidad de integrarla con nuevos módulos y/o transformar en alguno de los siguientes proyectos:

1. El módulo de autoevaluación del docente que se integrará con el módulo de autoevaluación del alumno.
2. El módulo de evaluación de pares que se puede integrar.
3. El módulo de evaluación del jefe inmediato superior para mejorar la herramienta de consulta.
4. La integración con un módulo del portafolio docente que sea complemento del proceso que realiza el Departamento de Calidad Educativa.
5. La integración a un módulo de certificación de materias, que es otro complemento del proceso que realiza el Departamento de Calidad Educativa.
6. La reutilización de parte del código para que los alumnos de investigación puedan realizar encuestas online para sus investigaciones, agilizando el proceso de integración y análisis de resultados.
7. La reutilización del código para hacer una aplicación donde los docentes puedan realizar sus evaluaciones o encuestas.

APÉNDICE A

EVALUACIÓN DEL DESEMPEÑO DOCENTE



UNIVERSIDAD DE NAVOJOA

EVALUACIÓN DEL DESEMPEÑO DOCENTE

Desarrollo Curricular

Nombre del docente: _____ Semestre: _____

Curso que imparte: _____ Programa: _____

La Universidad, con el propósito de ofrecer una educación de calidad, requiere de su colaboración responsable y honesta para evaluar el desempeño de los profesores en el proceso educativo.

El cuestionario está dividido en dos partes. La primera parte evalúa al profesor y la segunda parte evalúa el curso en general.

Lee cuidadosamente cada uno de los enunciados y marca con una x el número que más se ajuste a lo que piensa sobre el desempeño del docente. La información se manejará en forma estrictamente confidencial.

1=Totalmente en desacuerdo 2 = Desacuerdo 3 = Indeciso 4 = De acuerdo 5 = Totalmente de acuerdo

PRIMERA PARTE. Evaluación del profesor

Rasgos profesionales y personales	1	2	3	4	5
1. Es cuidadoso en su arreglo personal .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. Es paciente y tolerante con el proceso de aprendizaje de los alumnos.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. Es receptivo a ideas y puntos de vista expresados por los alumnos.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4. Crea un clima de confianza y comunicación.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. Proyecta actitudes positivas hacia la institución.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. Transmite principios cristianos .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7. Existe integración de los principios bíblicos con los contenidos del curso que imparte.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8. Establece claramente y en acuerdo con los alumnos, las normas de convivencia , sustentadas en la filosofía institucional.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9. Estructura lógicamente la presentación de sus ideas, tanto en presentaciones orales como escritas .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10. Se expresa con claridad y complementa su exposición mediante el lenguaje corporal y el uso adecuado del tono de la voz .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

11. Varía la clase con gestos, pausas y silencios para resaltar alguna idea o en algún momento específico.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12. Domina los saberes del curso que imparte.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13. Muestra seguridad al impartir la clase.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
14. Asiste a las clases programadas.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
15. Es puntual al iniciar y al terminar la clase.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Investigación					
16. Usa fuentes bibliográficas y electrónicas apropiadas para el curso.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
17. Recomienda bibliografía apropiada para el curso.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18. A través de sus clases nos motiva a leer e investigar .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19. Relaciona los contenidos de la materia con otras asignaturas.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20. Asigna trabajos prácticos que desarrollen habilidades profesionales .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gestión del aprendizaje					
21. Entrega oportunamente el contrato de aprendizaje y los criterios de evaluación.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
22. Selecciona o desarrolla materiales didácticos .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
23. Estructura los saberes para facilitar experiencias de aprendizaje.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
24. Diseña situaciones para facilitar experiencias de aprendizaje: significativo, colaborativo y autónomo.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
25. Organiza y distribuye correctamente el tiempo de la clase .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
26. Distribuye adecuadamente el tiempo y las actividades para cumplir con los contenidos y objetivos del curso, de acuerdo al contrato de aprendizaje.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
27. Incorpora el uso de tecnologías de la información y la comunicación para apoyar los procesos de enseñanza, aprendizaje y evaluación.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
28. Establece con claridad qué se aprenderá durante la sesión.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
29. Demuestra claridad y organización al presentar la información .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

30. Da ejemplos útiles que facilitan el aprendizaje.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
31. Promueve el desarrollo de un pensamiento crítico sobre los temas del curso.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
32. Utiliza métodos y técnicas de enseñanza-aprendizaje (casos, aprendizaje basado en problemas, proyectos, mesa redonda, debate, etc.)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
33. Tiene un adecuado control del grupo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
34. Guarda un adecuado balance entre exigencia y flexibilidad en el trato al grupo.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
35. Intercala las exposiciones y explicaciones con preguntas y participaciones de los alumnos.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
36. El profesor reafirma o retoma las participaciones de los alumnos.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
37. Responde con claridad a las preguntas .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
38. Formula preguntas acorde a la temática de clase.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
39. Relaciona los contenidos del curso con aplicaciones prácticas .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
40. Realiza actividades o comentarios de retroalimentación al final de la sesión.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
41. Evalúa el logro de las metas del curso utilizando estrategias e instrumentos diversificados y acordes con los contenidos y propósitos.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
42. Entrega oportunamente los resultados de las evaluaciones realizadas.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
43. La evaluación está ligada con lo establecido en el contrato de aprendizaje.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
44. Las estrategias e instrumentos de evaluación muestran ser didácticos .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
45. Su forma de evaluar es entendible .	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SEGUNDA PARTE. Evaluación general del curso					
46. Los contenidos del curso se relacionan con temas de otras materias que cursaste durante el semestre.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
47. Los contenidos del curso siguen una secuencia lógica.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
48. En general, este curso se compara como uno de los mejores de los que has tomado en la universidad.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

APÉNDICE B

RESULTADOS DE EVALUACIÓN

Resultado de la evaluación del alumno

1 = Totalmente en desacuerdo 2 = Desacuerdo 3 = Indeciso 4 = De acuerdo 5 = Totalmente de acuerdo

Rasgos profesionales y personales	PROMEDIOS		
	Personal	Escuela	Institución
1. Es cuidadoso en su arreglo personal .			
2. Es paciente y tolerante con el proceso de aprendizaje de los alumnos.			
3. Es receptivo a ideas y puntos de vista expresados por los alumnos.			
4. Crea un clima de confianza y comunicación.			
5. Proyecta actitudes positivas hacia la institución.			
6. Transmite principios cristianos .			
7. Existe integración de los principios bíblicos con los contenidos del curso que imparte.			
8. Establece claramente y en acuerdo con los alumnos, las normas de convivencia , sustentadas en la filosofía institucional.			
9. Estructura lógicamente la presentación de sus ideas, tanto en presentaciones orales como escritas .			
10. Se expresa con claridad y complementa su exposición mediante el lenguaje corporal y el uso adecuado del tono de la voz .			
11. Varía la clase con gestos, pausas y silencios para resaltar alguna idea o en algún momento específico.			
12. Domina los saberes del curso que imparte.			
13. Muestra seguridad al impartir la clase.			
14. Asiste a las clases programadas.			
15. Es puntual al iniciar y al terminar la clase.			
Investigación			
16. Usa fuentes bibliográficas y electrónicas apropiadas para el curso			
17. Recomienda bibliografía apropiada para el curso			
18. A través de sus clases nos motiva a leer e investigar .			
19. Relaciona los contenidos de la materia con otras asignaturas.			

20. Asigna trabajos prácticos que desarrollen habilidades profesionales			
Gestión del aprendizaje			
21. Entrega oportunamente el contrato de aprendizaje y los criterios de evaluación.			
22. Selecciona o desarrolla materiales didácticos .			
23. Estructura los saberes para facilitar experiencias de aprendizaje.			
24. Diseña situaciones para facilitar experiencias de aprendizaje: significativo, colaborativo y autónomo.			
25. Organiza y distribuye correctamente el tiempo de la clase .			
26. Distribuye adecuadamente el tiempo y las actividades para cumplir con los contenidos y objetivos del curso, de acuerdo al contrato de aprendizaje.			
27. Incorpora el uso de tecnologías de la información y la comunicación para apoyar los procesos de enseñanza, aprendizaje y evaluación.			
28. Establece con claridad qué se aprenderá durante la sesión.			
29. Demuestra claridad y organización al presentar la información .			
30. Da ejemplos útiles que facilitan el aprendizaje.			
31. Promueve el desarrollo de un pensamiento crítico sobre los temas del curso.			
32. Utiliza métodos y técnicas de enseñanza-aprendizaje (casos, aprendizaje basado en problemas, proyectos, mesa redonda, debate, etc.)			
33. Tiene un adecuado control del grupo			
34. Guarda un adecuado balance entre exigencia y flexibilidad en el trato al grupo.			
35. Intercala las exposiciones y explicaciones con preguntas y participaciones de los alumnos.			
36. El profesor reafirma o retoma las participaciones de los alumnos.			
37. Responde con claridad a las preguntas .			
38. Formula preguntas acorde a la temática de clase.			
39. Relaciona los contenidos del curso con aplicaciones prácticas .			
40. Realiza actividades o comentarios de retroalimentación al final de la sesión.			
41. Evalúa el logro de las metas del curso utilizando estrategias e instrumentos diversificados y acordes con los contenidos y propósitos.			

42. Entrega oportunamente los resultados de las evaluaciones realizadas.			
43. La evaluación está ligada con lo establecido en el contrato de aprendizaje.			
44. Las estrategias e instrumentos de evaluación muestran ser didácticos .			
45. Su forma de evaluar es entendible .			
PROMEDIO GENERAL			

APÉNDICE C

DESCRIPCIÓN DE CASOS DE USO

Validar usuario

Caso de uso	validar usuario
Resumen	Este caso de uso valida al usuario para poder tener acceso al sistema y determinar sus permisos que se le asignan a cada usuario
Actor principal	Usuario
Precondiciones	Tener perfil de usuario
Flujo principal	<ol style="list-style-type: none"> 1. El usuario entrar El usuario entra a la página Web docente.unav.edu.mx. 2. El sistema le muestra la pagina de validación 3. El usuario ingresa su matrícula y contraseña 4. El sistema valida si el usuario y contraseña son correctos, entra al sistema con los permisos que tiene derecho el usuario y acaba el caso de uso
Flujo Alternativo	<ol style="list-style-type: none"> 4. El sistema muestra error de Usuario o Contraseña equivocada y regresa el paso 2
Pos condiciones	El usuario queda validado, tomando los permisos de Docente o Administrador o Director de Escuela o Director de Facultad o Alumno

Aplicar evaluación

Caso de uso	Aplicar evaluación
Resumen	El alumno llena la evaluación de las materias que llevo en el semestre
Actor principal	Alumno
Precondiciones	Debes de estar validados y tener permisos de alumno

Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú aplicar evaluación 2. El sistema verifica si tiene evaluaciones disponibles y le muestra las evaluaciones que tiene acceso 3. El usuario elije la evaluación a contestar 4. El Sistema le muestra la evaluación 5. El usuario llena la evaluación y al terminar oprime enviar 6. El sistema verifica que todas las respuestas se hayan llenado y guarda los resultados de la evaluación y muestra un mensaje de gracias por llenar la evaluación
Flujo Alternativo	<ol style="list-style-type: none"> 2. El sistema verifica que no tiene evaluaciones disponibles y le muestra un mensaje evaluaciones disponibles
Pos condiciones	Los resultados de la encuesta es guardada y el alumno ya no puede volver a llenarla.

Consultar reporte personal

Caso de uso	Consultar reporte personal
Resumen	El Docente consulta el reporte de su evaluación en el sistema
Actor principal	Docente
Precondiciones	Debes de estar validados y tener permisos de Docente
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú reportes por maestro 2. El sistema verifica en que periodos tiene evaluaciones y los muestra las evaluaciones disponibles 3. El usuario elije la evaluación a consultar 4. El Sistema le muestra la evaluación
Flujo Alternativo	<ol style="list-style-type: none"> 2. El sistema verifica que no tiene evaluaciones disponibles y le muestra un mensaje evaluaciones disponibles
Pos condiciones	

Consultar reporte de escuela

Caso de uso	Consultar reporte de escuela
Resumen	El Director de Escuela consulta el reporte de la evaluación de los Docentes de su carrera
Actor principal	Director de Escuela

Precondiciones	Debes de estar validados y tener permisos de Director de Escuela
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú reportes por escuela 2. El sistema verifica los docentes que pertenecen a su escuela y los muestra 3. El usuario elije al docente a consultar 4. El sistema verifica en que periodos tiene evaluaciones y los muestra las evaluaciones disponibles 5. El usuario elije la evaluación a consultar 6. El Sistema le muestra la evaluación
Flujo Alternativo	4. El sistema verifica que no tiene evaluaciones disponibles y le muestra un mensaje evaluaciones disponibles
Pos condiciones	

Consultar reportes personalizados

Caso de uso	Consultar reporte personalizados
Resumen	El Administrador consulta el reporte de la evaluación de los Docentes de su carrera
Actor principal	Administrador
Precondiciones	Debes de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú reportes personalizados 2. El sistema muestra una opción de búsqueda 3. El usuario escribe al docente a consultar 4. El sistema verifica en que periodos tiene evaluaciones y los muestra las evaluaciones disponibles 5. El usuario elije la evaluación a consultar 6. El Sistema le muestra la evaluación
Flujo Alternativo	4. El sistema verifica que no tiene evaluaciones disponibles y le muestra un mensaje evaluaciones disponibles
Pos condiciones	

Editar docentes

Caso de uso	Editar docentes
Resumen	El Administrador modifica al docente la escuela a la que pertenece
Actor principal	Administrador
Precondiciones	Debes de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none">1. El usuario escoge en el menú administración editar docente2. El sistema muestra la lista de docentes activos y la escuela a la que pertenecen.3. El usuario modifica la escuela a la que pertenecen el docente y oprime guardar4. El sistema guarda el cambio de la escuela asignada al docente
Flujo Alternativo	
Pos condiciones	Se modifica la tabla del campo escuela-docente

Modificar status

Caso de uso	Modificar status
Resumen	El Administrador modifica el status de un docente
Actor principal	Administrador
Precondiciones	Debes de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none">1. El usuario escoge en el menú administración status2. El sistema muestra la lista de docentes3. El usuario modifica el estatus del docente y oprime guardar4. El sistema guarda el cambio de status del docente
Flujo Alternativo	
Pos condiciones	Se modifica la tabla del campo status del docente

Editar facultad

Caso de uso	Editar facultad
Resumen	El Administrador modifica o agrega una facultad
Actor principal	Administrador
Precondiciones	Debes de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú administración Facultad 2. El sistema muestra la lista de facultades y al final agregar una nueva 3. El usuario modifica una escuela y oprime modificar 4. El sistema guarda las modificaciones
Flujo Alternativo	<ol style="list-style-type: none"> 3. El usuario agregar una nueva escribiendo la que quiere añadir 4. El Sistema agrega una nueva facultad si le oprimió a agregar una nueva
Pos condiciones	Se modifica la tabla del campo nombre o se añade un campo

Asignar facultad-escuela

Caso de uso	Asignar facultad-escuela
Resumen	El Administrador asigna relación Facultad Escuela
Actor principal	Administrador
Precondiciones	Debes de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú administración Facultad-Escuela 2. El sistema muestra la lista de escuelas y al frente las facultades 3. El usuario modifica la facultad a la que pertenece una escuela y oprime guardar 4. El sistema guarda las modificaciones
Flujo Alternativo	
Pos condiciones	Se modifica la tabla de escuela el campo facultad que pertenece

Asignar derechos

Caso de uso	Asignar derechos
Resumen	El Administrador asigna los derechos a los usuarios
Actor principal	Administrador
Precondiciones	Deben de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú Administración Derechos 2. El sistema muestra una página donde se introduzcan la matrícula que se le modificaran los derechos 3. El usuario introduce la matrícula a modificar derechos 4. El sistema muestra los derechos del sistema 5. El usuario selecciona los derechos del usuario 6. El sistema guarda las modificaciones
Flujo Alternativo	4. El sistema no encuentra la matrícula del usuario y manda un mensaje de matrícula no encontrada y vuelve al paso 2
Pos condiciones	Se modifica la tabla de fechas de aplicación de evaluación

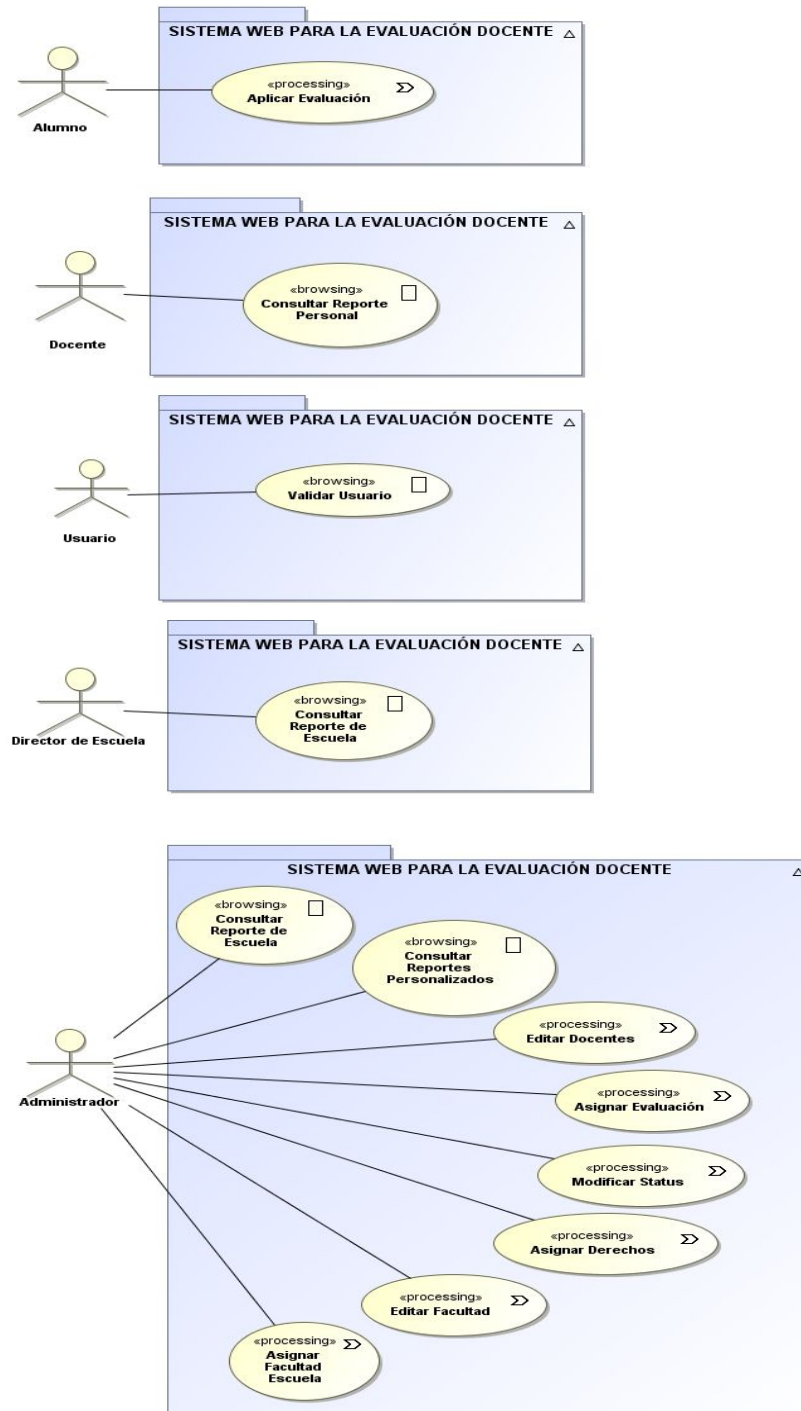
Asignar evaluación

Caso de uso	Asignar evaluación
Resumen	El Administrador asigna las fechas que se aplicara una evaluación
Actor principal	Administrador
Precondiciones	Debes de estar validados y tener permisos de Administrador
Flujo principal	<ol style="list-style-type: none"> 1. El usuario escoge en el menú Administración Asignar Evaluación 2. El sistema muestra los periodos disponibles 3. El usuario determina las fechas de aplicación de la evaluación 4. El sistema guarda las modificaciones
Flujo Alternativo	
Pos condiciones	Se modifica la tabla de fechas de aplicación de evaluación

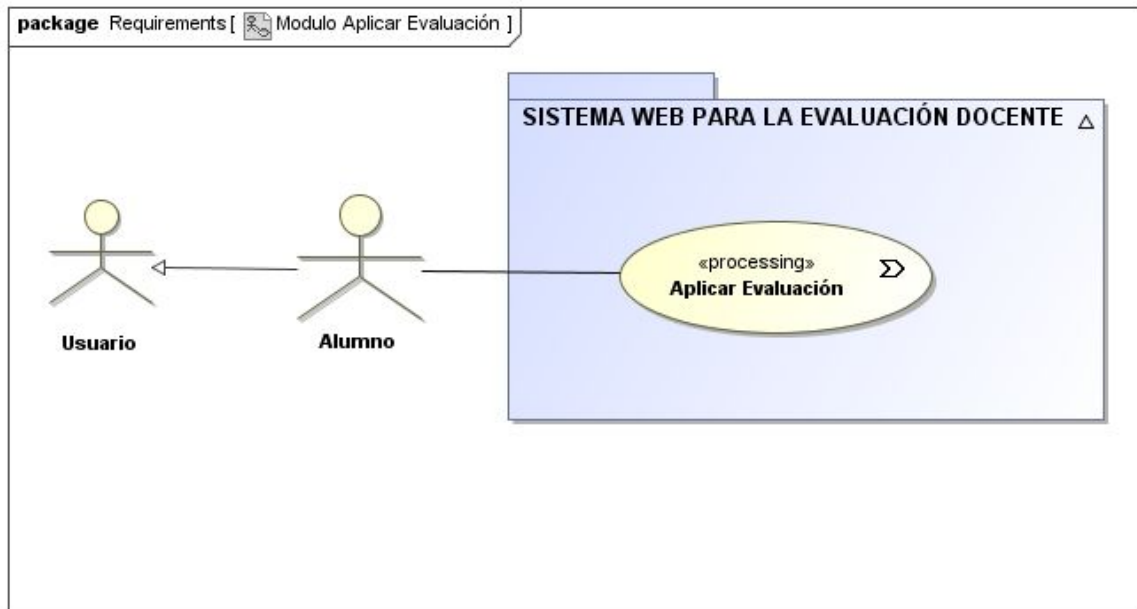
APÉNDICE D

CASO DE USO POR ACTOR Y MODULO

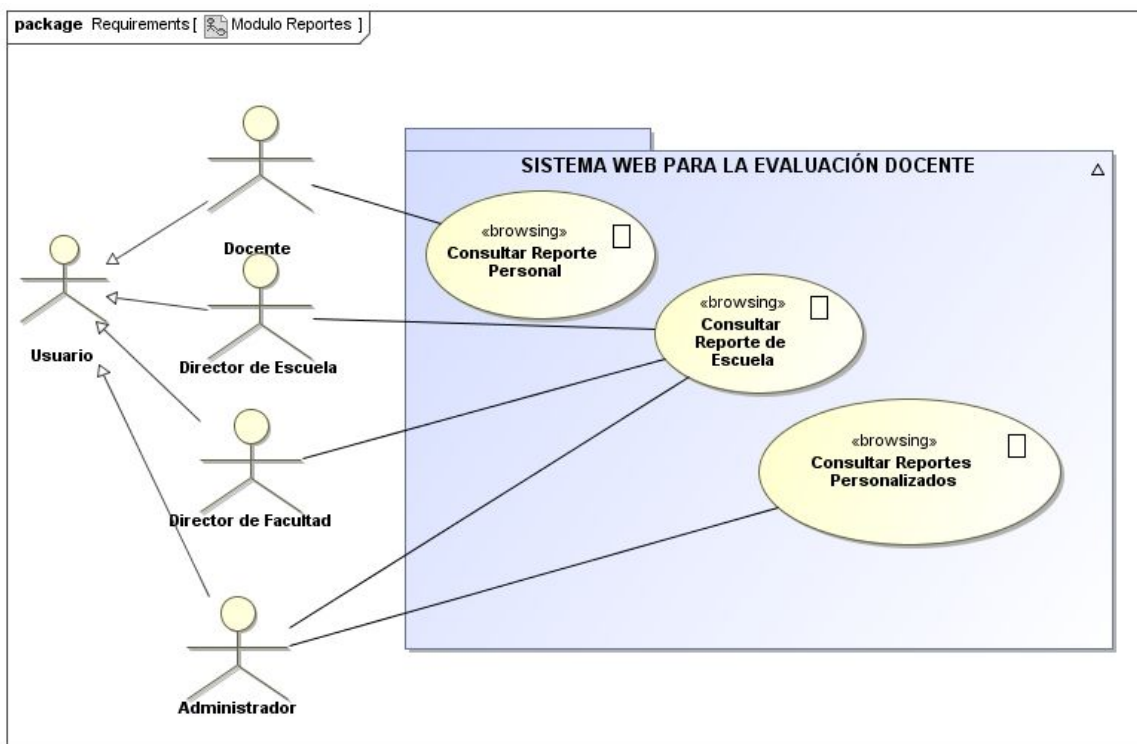
1. Caso de uso por actor



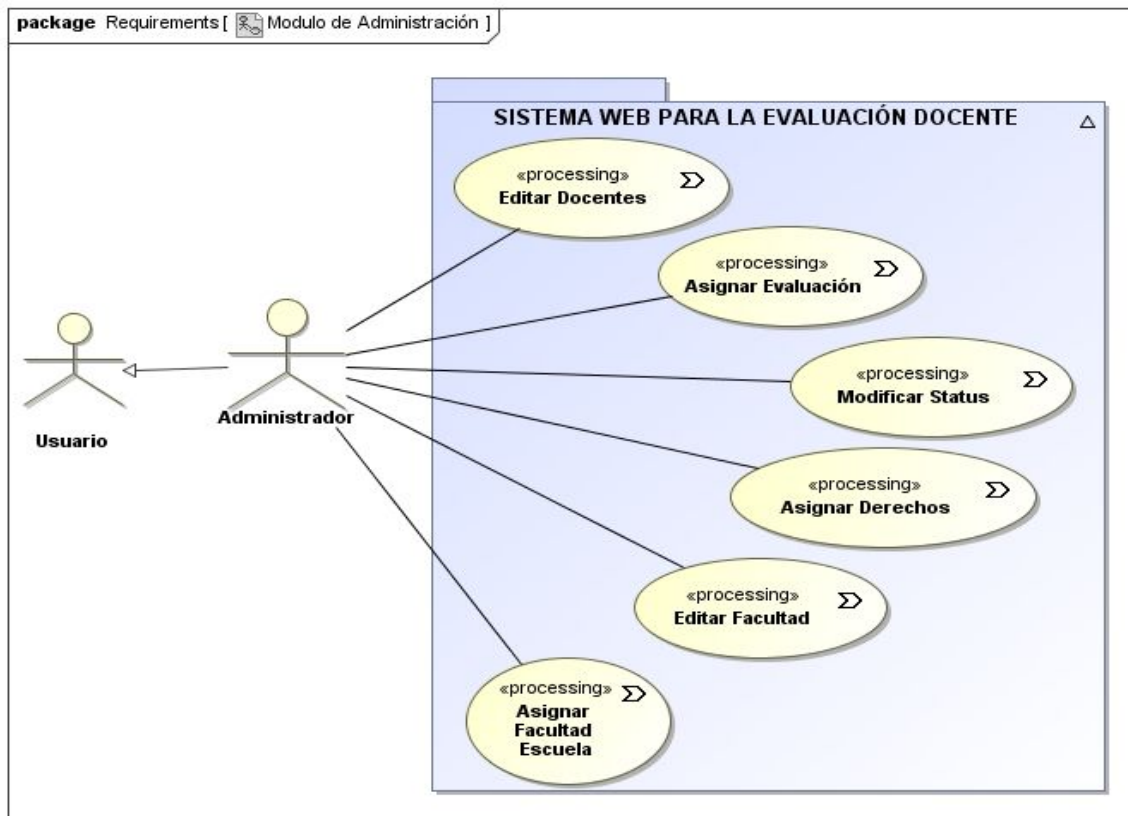
2. Módulo de aplicar evaluación



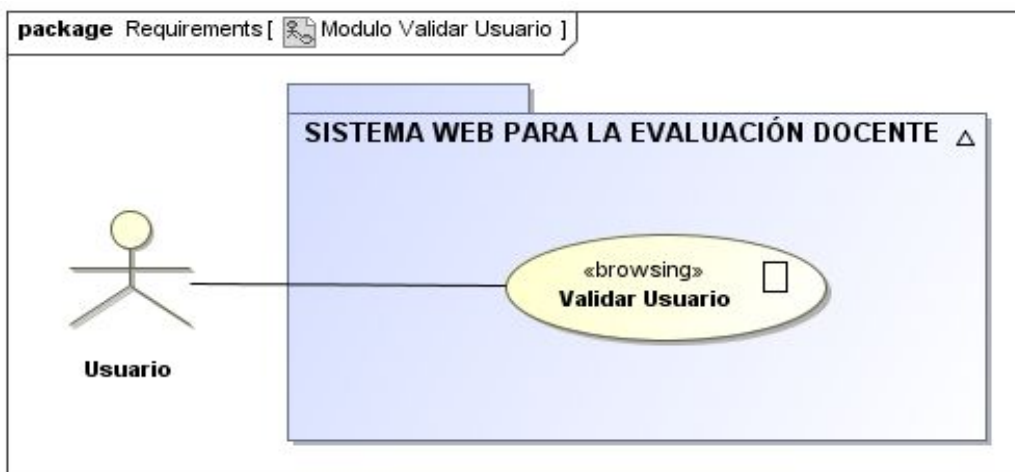
3. Módulo de reportes



4. Módulo de administración



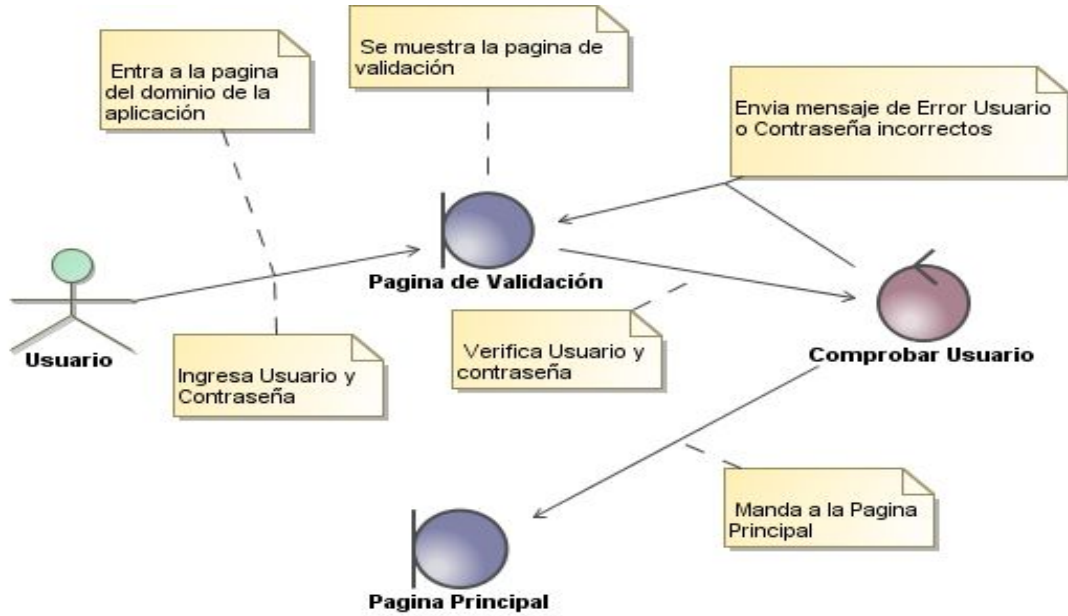
5. Módulo de validar usuario



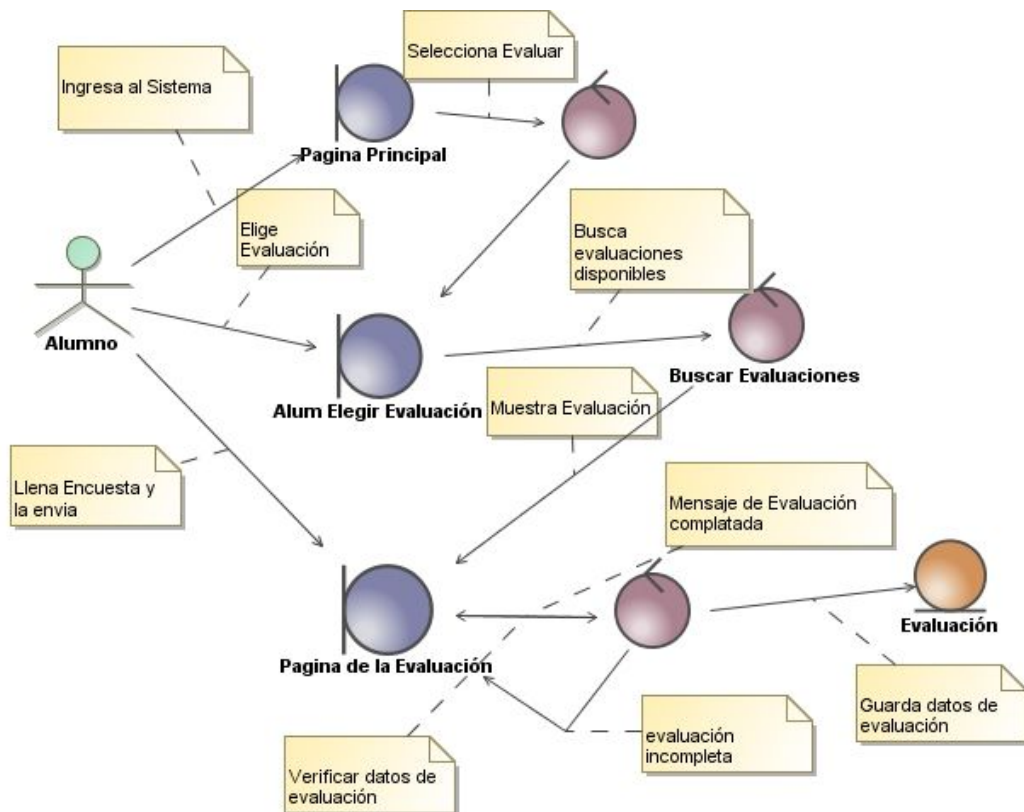
APÉNDICE E

DIAGRAMA DE ROBUSTEZ

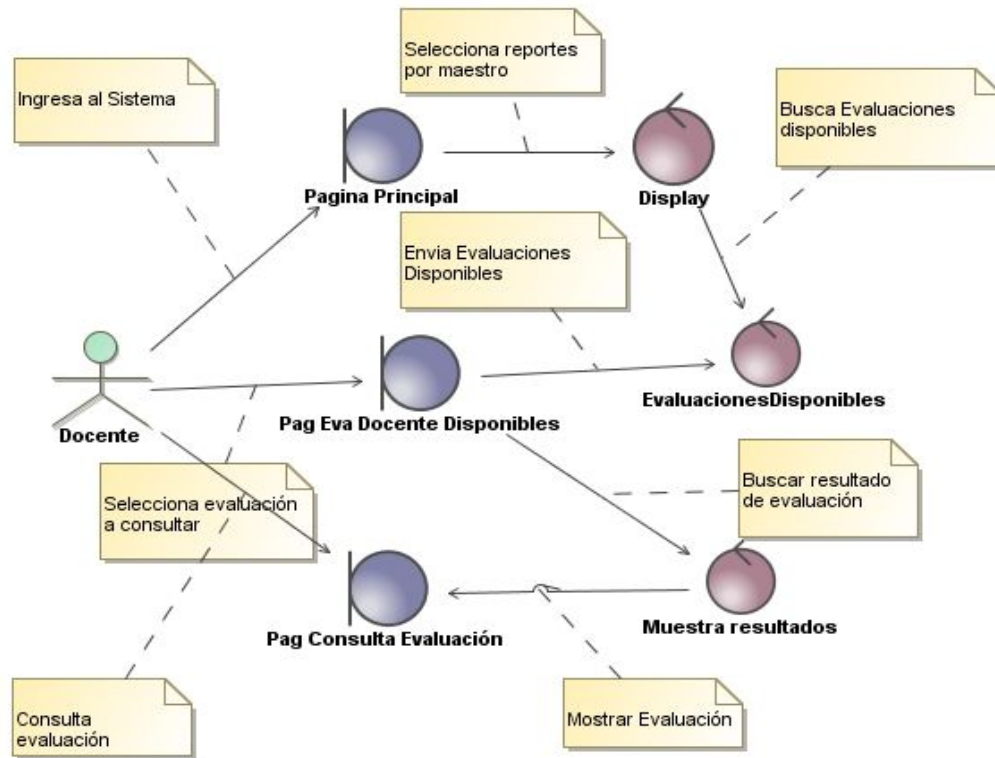
1. Validar usuario



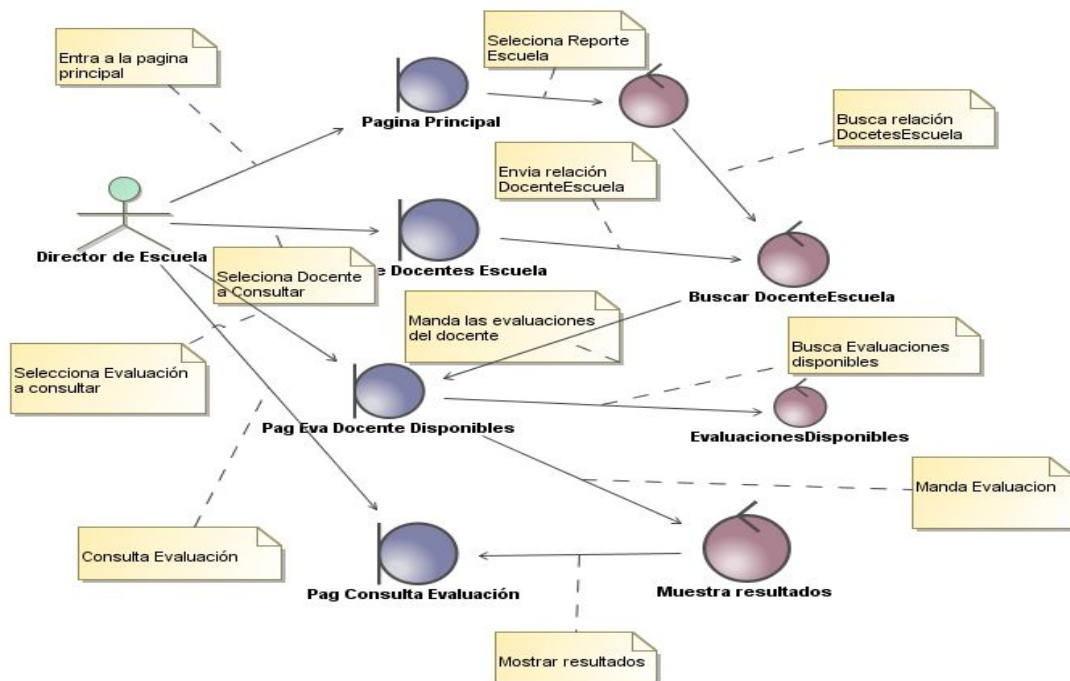
2. Aplicar evaluación



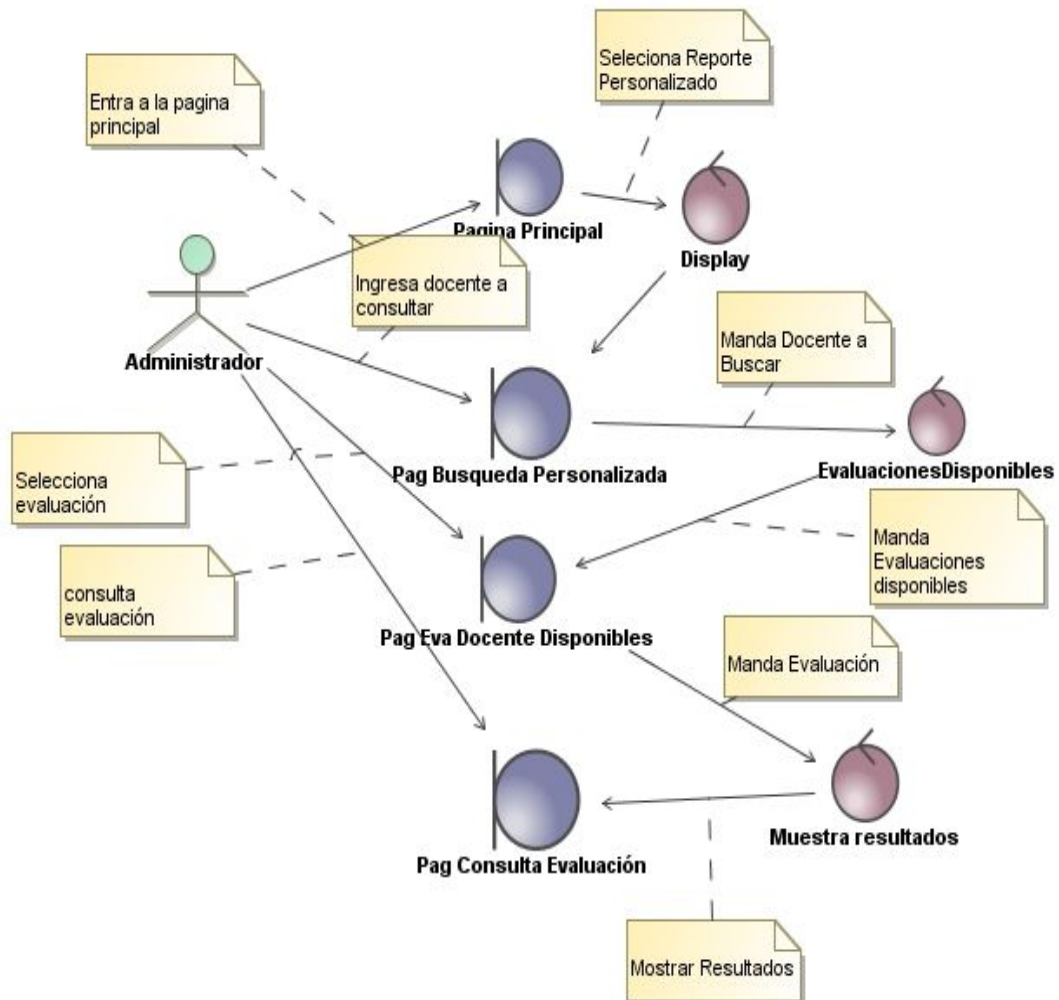
3. Consultar reporte personal



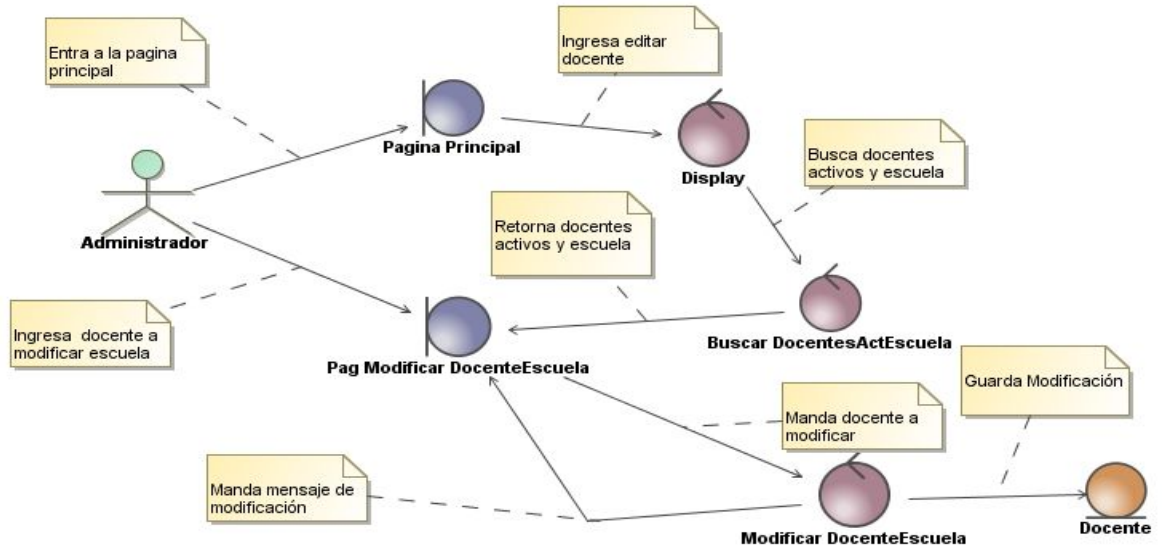
4. Consultar reporte de escuela



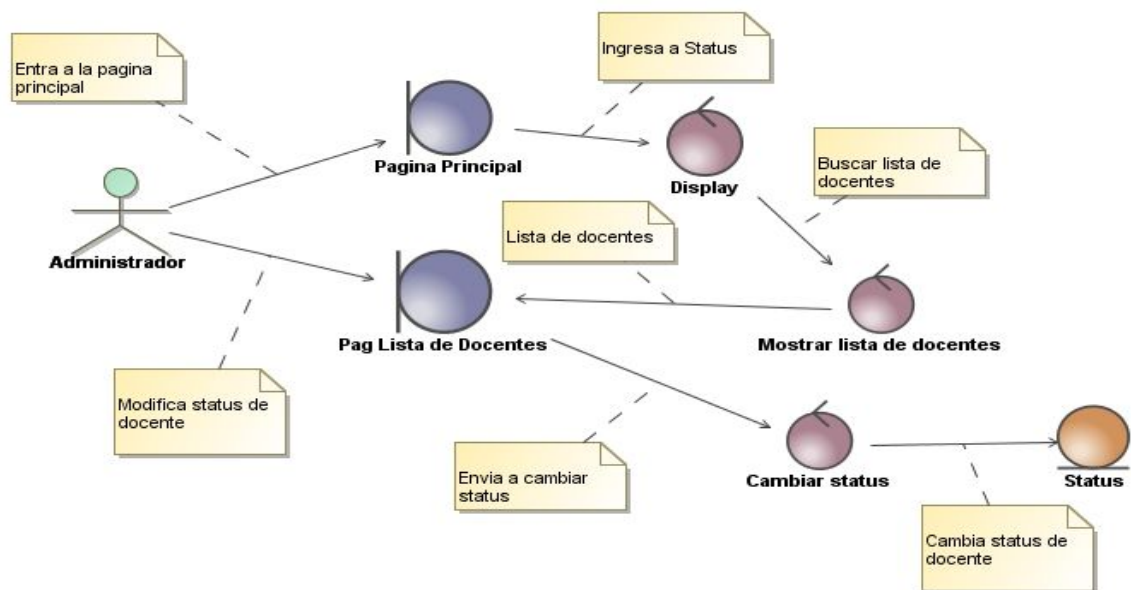
5. Consultar reportes personalizados



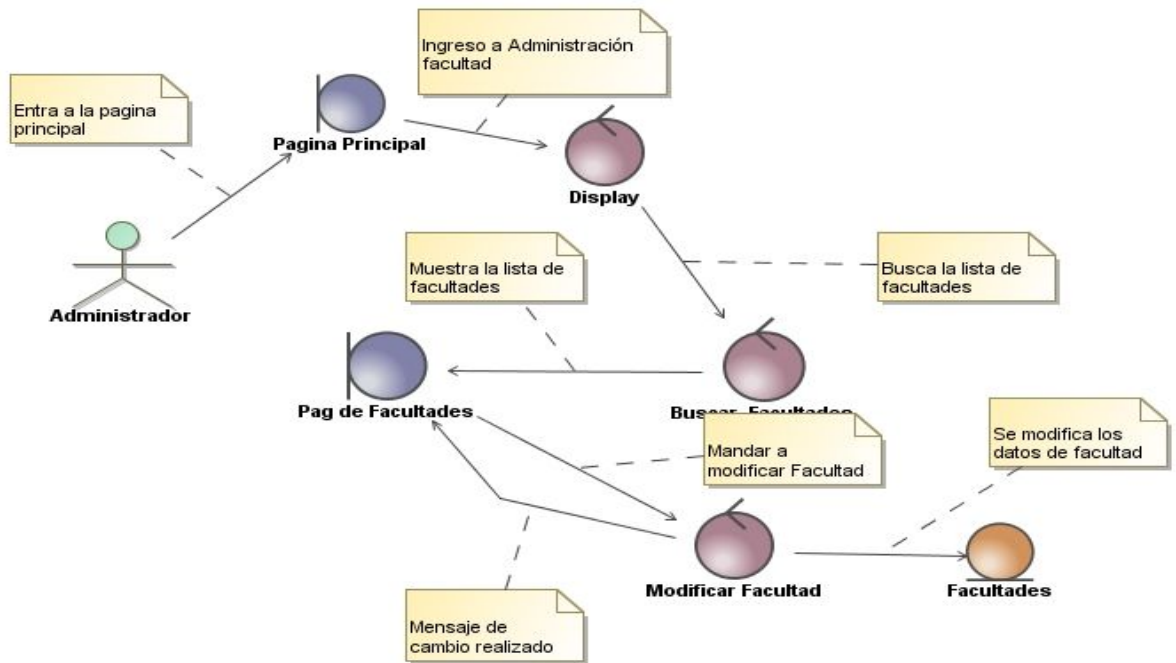
6. Editar docentes



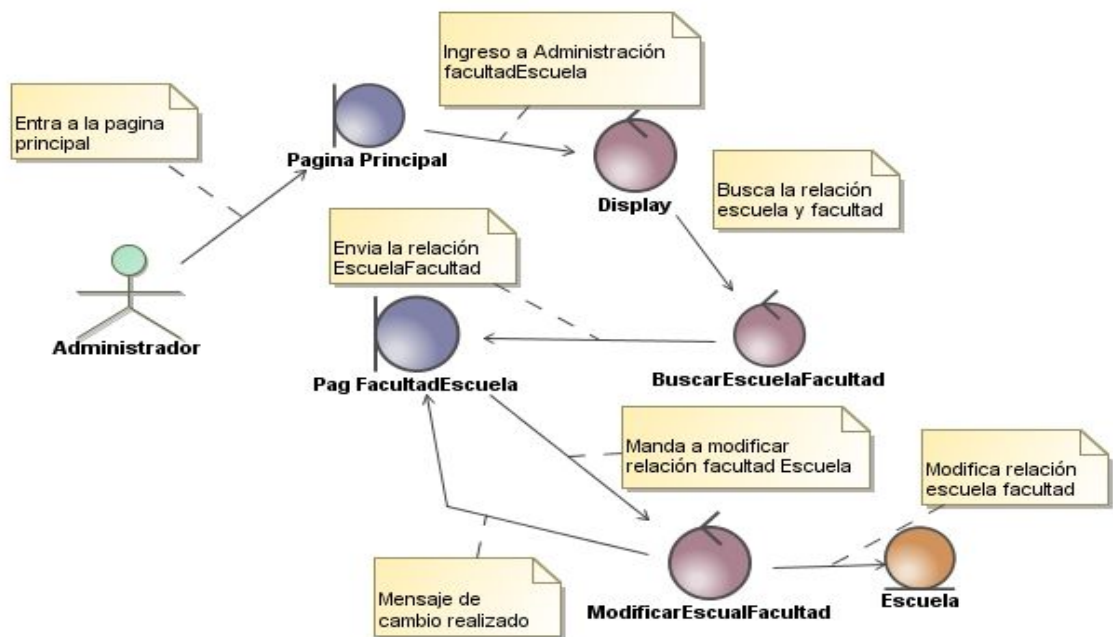
7. Modificar status



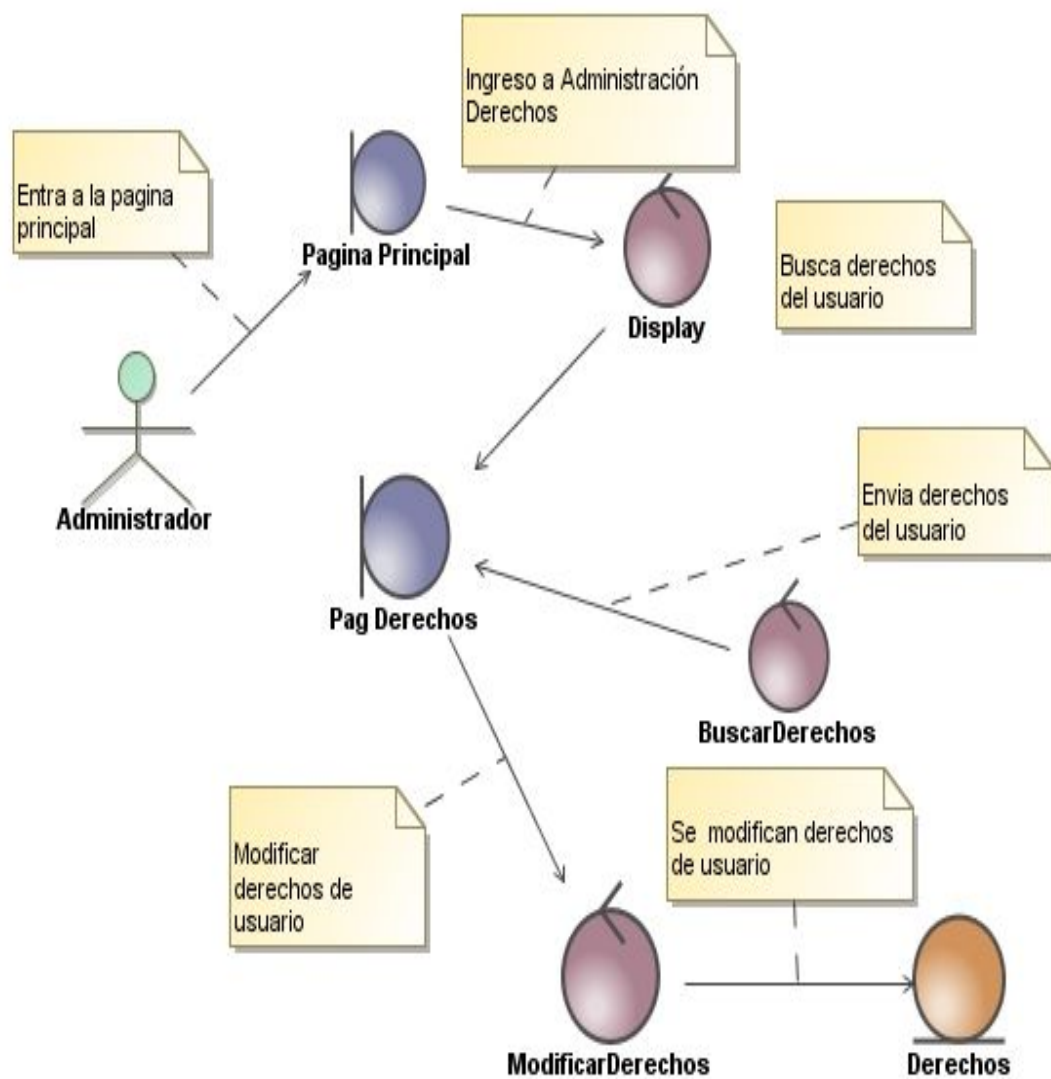
8. Editar facultad



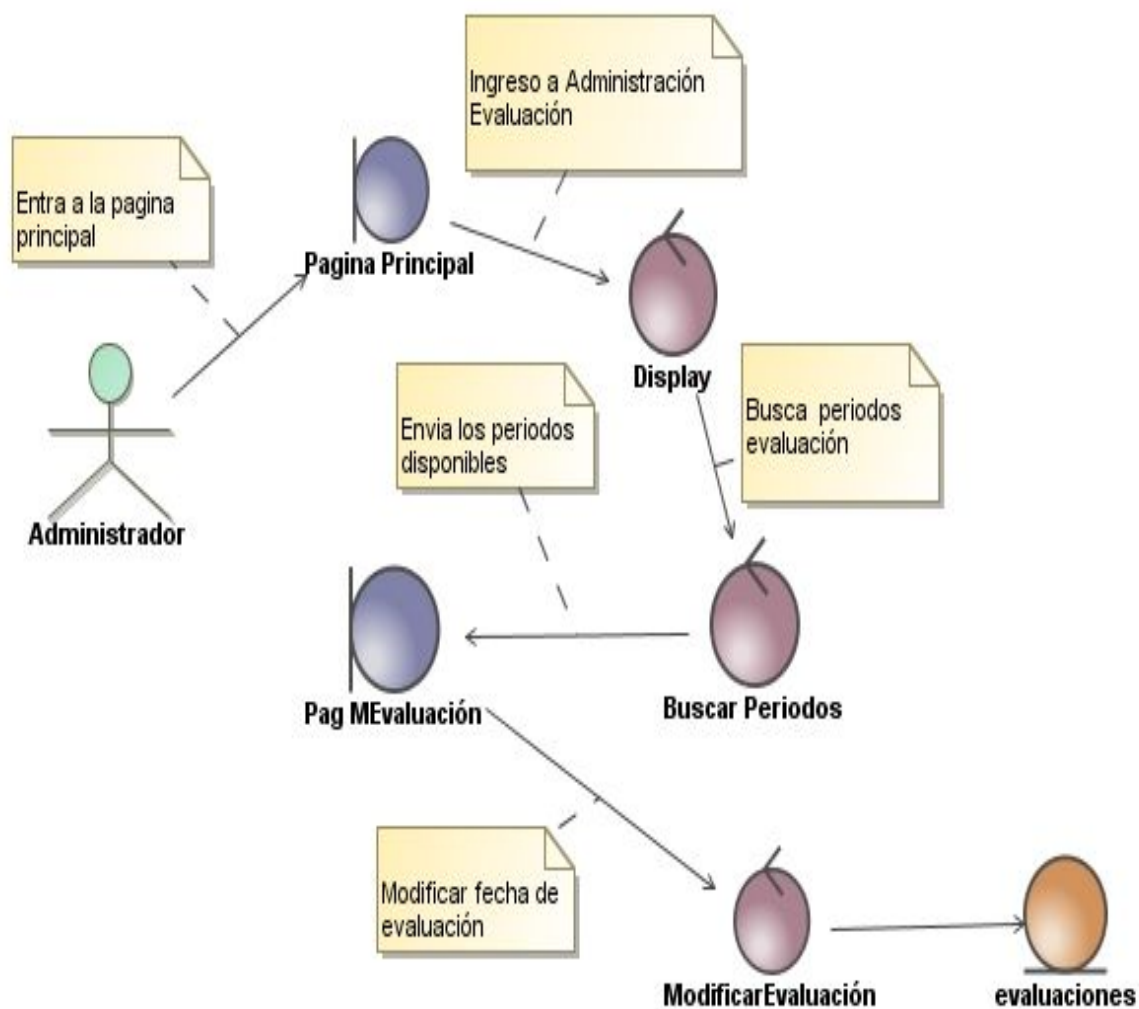
9. Asignar facultad-escuela



10. Asignar derechos



11. Asignar evaluación



APÉNDICE F

CÓDIGO FUENTE

```

<?php
/**
 * conf.php
 * Esta clase es necesaria para especificar todas las
 * diferentes variables para que funcione el programa.
 */
//Constantes de Base de Datos//
define("DB_SERVER", "");
define("DB_USER", ""); //usuario
define("DB_PASS", ""); //contrasena
define("DB_NAME", ""); //nombre
//Constantes de SESSIONES
define("USER_TABLE", ""); //tabla de usuarios
define("USER_FIELD", ""); //compo de usuario
define("PASSWORD_FIELD", ""); //campo de clave
define("PERMISSION_FIELD", ""); //campo de permisos
?>
<?php
/**
 * Process.php *
 * Esta clase cumple con el trabajo de procesar todas las formas.
 * */
if(!isset($nivel_dir)){
$nivel_dir="../";}

include($nivel_dir."tutzi/class/session.class.php");
//include("catalogos.class.php");
class ContPersona{};
$contPersona = new ContPersona;
?>
<?php
//Models go here
if(!isset($nivel_dir)){
$nivel_dir="../"; }
include ($nivel_dir."models/editar.model.php");
class Editar_Controller{//Clase
    function Editar_Controller(){//Constructor
if(isset($_POST['tipo'])){
    switch ($_POST['tipo']) {
        case "maestro":
            # code...
            $this->updatemaestro();
            break;
        default:
            # code...
            break;    }    } }
function updatemaestro(){
global $editar;
$editar->updatemaestro($_POST['campo'],$_POST['value'],$_POST['id']);

```

```

}
function cargarPagina($pagina){
global $editar;
    switch ($pagina) {
        case "maestro":
            # code...
            return $editar->maestro();
            break;
        case "escuela":
            # code...
            return $editar->escuela();
            break;
        case "facultad":
            # code...
            return $editar->facultad();
            break;
        default:
            return "NOT FOUND";
            break; } //echo "TIMEOUT!"; }
function cargarAccion($accion,$id){
    global $editar;
    if($accion=="MaestroEscuela"){
        return $editar->maestroEscuela($id);
    }else if($accion=="MaestroEstatus"){
        return $editar->maestroEstatus($id); } }}

$editar_controller = new Editar_Controller();
?>
<?php
/**
 * Process.php *
 * Esta clase cumple con el trabajo de procesar todas las formas.
 * */
if(!isset($nivel_dir)){
$nivel_dir="../"; }
include($nivel_dir."models/editar.model.php");
include_once($nivel_dir."tutzi/class/tools.class.php");

class MainController{
    /* Constructor de clase */
    function MainController(){
        if($_GET['tipo']=="update"){
            $this->update(); } }
function update(){
global $editar;
global $tools;

```

```

$editar-
>update($_GET['tabla'],$_GET['campo'],$_GET['value'],$_GET['where'],$_GET['
id']);
$tools->safe_redirect($_GET['url']);}};
$mainController = new MainController;?>
<?php
/**
 * Process.php *
 * Esta clase cumple con el trabajo de procesar todas las formas.
 * */
include("../tutzi/class/session.class.php");
//include("catalogos.class.php");
class ContPersona{ };
$contPersona = new ContPersona;?>
<?php/**
 * Process.php *
 * Esta clase cumple con el trabajo de procesar todas las formas. * */
include("session.class.php");
include("ejercicio.class.php");
include("catalogos.class.php");
//include("catalogos.class.php");
class Process{
    /* Constructor de clase */
    function Process(){
        global $session;
        /* Prosesar formas */
        if(isset($_POST['login'])){
            $this->login();        }
        if(isset($_GET['timeout'])){
            $this->timeout();
            //echo "TIMEOUT!";        }
        if(isset($_GET["logout"])){
            $this->Logout();        }
        if(isset($_GET["ejer_ins"])){
            $this->ejer_ins();        }
        if(isset($_GET["cuenta_ins"])){
            $this->cuenta_ins();        }
        if(isset($_GET["set_ejer"])){
            $this->set_ejer();}}
    function login(){
        global $session;
        $retval = $session->login($_POST['user'], $_POST['pass']);
        if($retval){header("Location:../main.php?encuesta");
        }
        else{header("location:../index.php?error=loginfail");
        }    }

    function Logout(){
        global $session;

```

```

    $retval = $session->logout();
    header("Location: ../index.php?logout");    }
function timeout(){
    global $session;
    $nom_usuario=$_SESSION['username'];
    $session->logout();
    header("Location: ../timeout.php?user=$nom_usuario");    }
function set_ejer(){
    global $session;
    $session->setEjersio($_GET['set_ejer']);
    header("Location: ../main.php");    }
function ejer_ins(){
    global $ejercicio;
    $ejercicio-
>insEjercicio($_POST['nom_Empresa'],$_POST['descripcion'],$_POST['rfc'],$_P
OST['direccion'],$_POST['telefono']);
    header("Location: ../main.php?ejercicios");    }
function cuenta_ins(){
    global $catalogo;
    $catalogo-
>insCuenta($_POST['nombre'],$_POST['descripcion'],$_POST['tipo'],$_POST['ej
ercicio']);}}};
$process = new Process;?> <?php
//Models go here
if(!isset($nivel_dir)){
    $nivel_dir="../"; }
include ($nivel_dir."models/reporte.model.php");

class Reporte_Controller{//Clase
    function Reporte_Controller(){//Constructor
    }
    function getDatosMaestro($id){
    global $reporte;
    return $reporte->getDatosMaestro($id);
    }
    function getPromedioMaestro($id){
    global $reporte;
    return $reporte->getPromedioMaestro($id);
    }
    function getPromedioEscuela($pregunta,$escuela){
    global $reporte;
    return $reporte->getPromedioEscuela($pregunta,$escuela);
    }
    function getPromedioUniversidad($pregunta){
    global $reporte;
    return $reporte->getPromedioUniversidad($pregunta);
    }
    function datosDisponibles($id){
    global $reporte;

```

```

        return $reporte->datosDisponibles($id);
    }
}
$reporte_controller = new Reporte_Controller();
?>
<?php
//Models go here
if(!isset($nivel_dir)){
    $nivel_dir="../";
}
include_once($nivel_dir."models/persona.model.php");
class Editar{//Clase

    var $html;

    function Model(){//Constructor
}
    function maestro(){
        $html= '
<a class="btn btn-app">
                <i class="fa fa-edit"></i> Docente
sin Escuela
                </a>
                <a class="btn btn-app">
                    <i class="fa fa-edit"></i>
Administradores
                </a>
                <a class="btn btn-app">
                    <i class="fa fa-save"></i> otro
                </a>
                <a class="btn btn-app">
                    <span class="badge bg-
yellow">3</span>
                    <i class="fa fa-edit"></i> otro
                </a>
<div class="box">
                <div class="box-header">
                    <h3 class="box-title">Docentes</h3>
                    <div class="box-tools">
                        <div class="input-group">
                            <input type="text"
name="table_search" class="form-control input-sm pull-right" style="width:
150px;" placeholder="Search">
                            <div class="input-group-btn">
                                <button class="btn btn-sm
btn-default"><i class="fa fa-search"></i></button>
                                </div>
                            </div>
                </div>

```

```

        </div>
    </div><!-- /.box-header -->
    <div class="box-body table-responsive no-
padding">
        <table class="table table-hover">
            <tbody><tr>
                <th>Matricula</th>
                <th>Nombre</th>
                <th>Status</th>
                <th>Grupo</th>
                <th>Escuela</th>
            </tr>
            ';

    $data=$this->getMaestros();
    foreach ($data as $maestro) {
        $html=$html. "<tr>";
        $html=$html."<td>".$maestro['idMaestro']."</td>";
        $html=$html."<td>".$maestro['Nombre']." ". $maestro['A_Paterno']."
".$maestro['A_Materno']."</td>";
        $html=$html. '<td><a
href="index.php?page=editar&tipo=maestro&accion=MaestroEstatus&id='.$maestr
o['idMaestro'].' " >';
        if($maestro['estatus']==0){
            $html=$html.<span class="label label-danger">Inactivo';
        }else if($maestro['estatus']==1){
            $html=$html.<span class="label label-success">Activo';
        }
        $html=$html.</span></a></td>';
        $html=$html. '<td><span class="label label-
success">'.$maestro['nombre'].'</span></td>';
        $html=$html.<td><a
href="index.php?page=editar&tipo=maestro&accion=MaestroEscuela&id='.$maestr
o['idMaestro'].' " ><span class="label label-
success">'.$maestro['descripcion'].'</span></td></a>';
        $html=$html. "</tr>";
    }

    $html=$html.'
    </tbody></table>
    </div><!-- /.box-body -->
    </div>';

    return $html;

}
function escuela(){
global $persona;
echo $persona->printPermisos();
//echo $persona->esEstudiante();
}

```



```

        function facultad(){
        function getMaestros(){
        global $database;
        $data=$database->query("SELECT * FROM
`Maestro`,`Usuario`,`grupos`,`Escuelas` WHERE `idMaestro`=`idUsuario` AND
`Usuario`.`grupo` = `grupos`.`id_grupo` AND
`Escuelas`.`idEscuelas`=`Maestro`.`id_escuela`");
        return $database->toArray($data); }
        function maestroEscuela($id){
        global $database;
        $data=$this->getEscuelas();
        $maestro=$database->query("SELECT * FROM `Maestro` WHERE `idMaestro` =
$id");
        $maestro=$database->toArray($maestro);
        $nombre="";
        $idEscuela="";
        foreach ($maestro as $key) {
        $nombre=$key['Nombre'];
        $idEscuela=$key['id_escuela'];
        }
        $this->htmladd('<div class="box box-primary">
                <div class="box-header">
                <h3 class="box-title">Editar
Maestro</h3>
                </div><!-- /.box-header -->
                <!-- form start -->
                <form
action="controller/main.controller.php" method="get">
                <div class="box-body">
                        <input type="hidden" name="id"
value="'. $id.'">
                        <input type="hidden" name="tipo"
value="update">
                        <input type="hidden" name="campo"
value="id_escuela">
                        <input type="hidden" name="tabla"
value="Maestro">
                        <input type="hidden" name="where"
value="idMaestro">
                        <input type="hidden" name="url"
value="../index.php?page=editar&tipo=maestro">
                        <div class="form-group">
                                <label>Nombre</label>
                                <input type="text" class="form-
control" placeholder="'. $nombre.'" disabled="">
                        </div>

```

```

        <div class="form-group">
            <label>Escuela</label>
            <select class="form-control"
name="value">');
        foreach ($data as $escuela) {
            # code...

            $this->htmladd('<option value="'. $escuela['idEscuelas']. '");
            if($escuela['idEscuela']==$idEscuela){
                $this->htmladd("selected");
            }
            $this->htmladd('>' . $escuela['descripcion'] . '</option>'); }

        $this->htmladd(' </select>
                                </div>
                                </div><!-- /.box-body -->
                                <div class="box-footer">
                                    <button type="submit" class="btn
btn-primary">Submit</button>
                                </div>
                                </form>
                                </div>');

        return $this->html;
        $this->html=""; }
        function maestroEstatus($id){
            global $database;
            $data=$this->getEscuelas();
            $maestro=$database->query("SELECT * FROM `Maestro` WHERE `idMaestro` =
$id");
            $maestro=$database->toArray($maestro);
            $nombre="";
            $idEscuela="";
            foreach ($maestro as $key) {
                $nombre=$key['Nombre'];
                $idEscuela=$key['id_escuela'];
            }
            $this->htmladd('<div class="box box-primary">
                                <div class="box-header">
                                    <h3 class="box-title">Editar
Maestro</h3>
                                </div><!-- /.box-header -->
                                <!-- form start -->
                                <form
action="controller/main.controller.php" method="get">
                                    <div class="box-body">

```

```

value="'. $id.'">
value="update">
value="estatus">
value="Usuario">
value="idUsuario">
value="../index.php?page=editar&tipo=maestro">
<input type="hidden" name="id"
<input type="hidden" name="tipo"
<input type="hidden" name="campo"
<input type="hidden" name="tabla"
<input type="hidden" name="where"
<input type="hidden" name="url"
<div class="form-group">
    <label>Nombre</label>
    <input type="text" class="form-
control" placeholder="'. $nombre.'" disabled="">
</div>
<div class="form-group">
    <label>Estatus</label>
    <select class="form-control"
name="value">
ACTIVO</option>
value="1">ACTIVO</option>
    ');
    $this->htmladd(' </select>
    </div>
</div><!-- /.box-body -->
<div class="box-footer">
    <button type="submit" class="btn
btn-primary">Submit</button>
    </div>
</form>
</div>');
return $this->html;
$this->html=""; }
function getEscuelas(){
global $database;
$data=$database->query("SELECT * FROM `Escuelas`");
return $database->toArray($data); }
function htmladd($html){
    $this->html=$this->html.$html; }
function updatemaestro($campo,$valor,$id){

```

```

        global $database;
        $query="UPDATE `Maestro` SET `$campo` = '$valor' WHERE `idMaestro`
= $id";
        echo $query;
        $database->query("UPDATE `Maestro` SET `$campo` = '$valor' WHERE
`idMaestro` = $id");
    }
    function update($tabla,$campo,$valor,$where,$id){
        global $database;
        $query="UPDATE `$tabla` SET `$campo` = '$valor' WHERE `$where` =
$id";
        //echo $query;
        // echo $query;
        $database->query($query);    }}

$editar = new Editor();
?>
<?php
// encuesta.model.php
// Esta clase es la primera que se encarga de recoger las variables
// de session y tomar las diferentes permisos.
if(!isset($nivel_dir)){
$nivel_dir="../";
}
include_once($nivel_dir."tutzi/class/session.class.php");
include_once($nivel_dir."tutzi/class/tools.class.php");

class Encuesta
{
var $nombre;        //Usuario
var $materias;      //Contrasena
var $id_estudiante;
var $escuela;

function Encuesta(){

    if(isset($_POST['tipo'])){
        $this->save2db();}

global $session;
global $tools;
global $database;
$estu_matri=$_SESSION['username'];
$q="SELECT * FROM `Carga`,`Detalle_Clase`,`Estudiante` WHERE
`Estudiante_idEstudiante` = $estu_matri AND
`Detalle_Clase`.`idMateria_Detalle` =
`Carga`.`Detalle_Clase_idMateria_Detalle` AND
`Carga`.`Estudiante_idEstudiante`=`Estudiante`.`idEstudiante`";
$content=$database->query($q);
$estudiante = $content->fetch_assoc();

```

```

$this->nombre= $estudiante['Nombre']." ".$estudiante['A_Paterno']." ";
$estudiante['A_Materno'];
$this->materias=$content->num_rows;
$this->id_estudiante=$_SESSION['username'];
$this->escuela=$estudiante['id_Escuela'];}
function disponible(){
global $session;
global $tools;
global $database;
$estu_matri=$_SESSION['username'];
$sq="SELECT * FROM `Contestados` WHERE `matricula` = $estu_matri";
$result=$database->query($sq);
if($result->num_rows>=1){
return 1;}
else{
return 2;}}
function getMaterias(){
global $database;
$estu_matri=$_SESSION['username'];
$q="SELECT * FROM `Carga`,`Detalle_Clase`,`Estudiante` WHERE
`Estudiante_idEstudiante` = $estu_matri AND
`Detalle_Clase`.`idMateria_Detalle` =
`Carga`.`Detalle_Clase_idMateria_Detalle` AND
`Carga`.`Estudiante_idEstudiante`=`Estudiante`.`idEstudiante`";
$content=$database->query($q);
return $database->toArray($content);}
function getMatCat($categoria){
global $database;
$estu_matri=$_SESSION['username'];
$q="SELECT * FROM `Pregunta` WHERE `Encuestas_idEncuestas` = 1 AND
`id_Categoria` ='".$categoria."' ORDER BY `Pregunta`.`idPregunta` ASC";
$content=$database->query($q);
return $database->toArray($content);}
function getCategorias(){
global $database;
$q="SELECT * FROM `Categorias` WHERE `id_Encuesta` = 1";
$cat=$database->query($q);
return $database->toArray($cat);
}

function save2db(){
global $database;
$escuela=$_POST['escuela'];
$matricula=$_POST['matricula'];

$query="INSERT INTO `Respuestas` (`idRespuestas`,`respuesta`,
`T_Repuesta`,`Pregunta_idPregunta`,`Encuestas_idEncuestas`,`matricula`,
`id_Detalle`) VALUES ";
$q="SELECT * FROM `Pregunta` WHERE `Encuestas_idEncuestas` = 1";

```

```

$preguntas=$database->query($q);
while($preg_array = $preguntas->fetch_assoc())
{
$q="SELECT * FROM `Carga`,`Detalle_Clase`,`Estudiante` WHERE
`Estudiante_idEstudiante` = $matricula AND
`Detalle_Clase`.`idMateria_Detalle` =
`Carga`.`Detalle_Clase_idMateria_Detalle` AND
`Carga`.`Estudiante_idEstudiante`=`Estudiante`.`idEstudiante`;
$materias=$database->query($q);
while($mat_array = $materias->fetch_assoc())
{
$id=$preg_array['idPregunta'].",".$mat_array['idMateria_Detalle'];
//echo "Respuesta de pregunta: ".$preg_array['idPregunta']." Materia:
".$mat_array['idMateria_Detalle'];
//echo " ".$_POST[$id];
$idPregunta=$preg_array['idPregunta'];
$idMateria=$mat_array['idMateria_Detalle'];
$respuesta=$_POST[$id];
$query = $query . "(NULL, '".$respuesta."', '1', '".$idPregunta."', '1',
'".$matricula."', '".$idMateria."'),";
//echo $query;}}
if($this->disponible()==2){
$query=substr($query, 0, -1);
//echo $query;
$database->query($query);
$final_query="INSERT INTO `Contestados` (`matricula`, `id_Encuesta`,
`id_Periodo`, `id_Escuela`) VALUES ('$matricula', '1', '1', '$escuela');";
$database->query($final_query);
echo $final_query;}
else {
echo "ERROR";}
header("location:main.php?encuesta");}}
$encuesta = new Encuesta();
?>
<?php
// persona.model.php
// Esta clase es la primera que se encarga de recoger las variables
// de session y tomar las diferentes permisos.
if(!isset($nivel_dir)){
$nivel_dir="../";
}
include_once($nivel_dir."tutzi/class/session.class.php");
include_once($nivel_dir."tutzi/class/tools.class.php");
class Persona{
var $grupo; //Usuario
var $estudiante; //Contrasena
var $maestro;
var $dirFacultad;
var $dirEscuela;

```

```

var $nombre;      //Usuario
var $materias;    //Contraseña
var $id_estudiante;
var $escuela;
function Persona(){
global $session;
global $tools;
global $database;
$check=$session->checkLogin();
if(!$check&&empty($isindex))
{
$tools->safe_redirect("index.php?error=3");
//header("Location:index.php?error=3");
}
$this->getPermisos();
if($check&&isset($isindex)){

$tools->safe_redirect("index.php?page=bienvenida");
//header("Location:main.php");}}
function getName(){
global $session;
global $tools;
global $database;
$estu_matri=$_SESSION['username'];
$q="SELECT * FROM `Carga`,`Detalle_Clase`,`Estudiante` WHERE
`Estudiante_idEstudiante` = $estu_matri AND
`Detalle_Clase`.`idMateria_Detalle` =
`Carga`.`Detalle_Clase_idMateria_Detalle` AND
`Carga`.`Estudiante_idEstudiante`=`Estudiante`.`idEstudiante`;
$content=$database->query($q);
$estudiante = $content->fetch_assoc();
$this->nombre= $estudiante['Nombre']." ".$estudiante['A_Paterno']." ".
$estudiante['A_Materno'];
return $this->nombre;}
function getPermisos(){
$this->estudiante=$this->esEstudiante();
$this->maestro=$this->esMaestro();
$this->grupo=$this->grupo();
$this->dirFacultad=$this->dirFacultad();
$this->dirEscuela=$this->dirEscuela();}
function permiso($grupo){
if($this->grupo==$grupo||$this->grupo==7){
return TRUE;}
else {return FALSE;}}
function printPermisos(){
echo "Maestro?". $this->maestro."<br>";
echo "Estudiante?". $this->estudiante."<br>";
echo "id Grupo". $this->grupo."<br>";
if ($this->dirFacultad==NULL){

```

```

        echo "Director Facultad:NO <br>";
    }else{echo "Director Faculad:". $this->dirFacultad." <br>";}
    if ($this->dirEscuela==NULL){
        echo "Director Escuela:NO <br>";
    }else{echo "Director Escuela:". $this->dirEscuela." <br>";}}
function esEstudiante(){
global $database;
$usuario=$_SESSION['username'];
$query="SELECT * FROM `Estudiante` WHERE `idEstudiante` = $usuario";
if($database->exist($query)){
return 1;
}else {return 0;}}
function esMaestro(){
global $database;
$usuario=$_SESSION['username'];
$query="SELECT * FROM `Maestro` WHERE `idMaestro` = $usuario";
if($database->exist($query)){
return 1;
}else {return 0;}}
function grupo(){
global $database;
$usuario=$_SESSION['username'];
//$query="SELECT * FROM `Maestro` WHERE `idMaestro` = = $usuario";
return $database->
>selectSingleField("Usuario","grupo","idUsuario",$usuario);
//$result=$database->query($query);
//$result=$database->toArray($result);
//return $result['grupo'];
function dirEscuela(){
global $database;
$usuario=$_SESSION['username'];
$query="SELECT * FROM `Escuelas` WHERE `id_Director` = $usuario";
if($database->exist($query)){
return $database->
>selectSingleField("Escuelas","idEscuelas","id_Director",$usuario); }
else {return NULL;}}
function dirFacultad(){
global $database;
$usuario=$_SESSION['username'];
$query="SELECT * FROM `Facultades` WHERE `id_Director` = $usuario";
if($database->exist($query)){
return $database->
>selectSingleField("Facultades","idFacultades","id_Director",$usuario);}
else {return NULL;}}
$persona = new Persona();
?>
<?php
//Models go here
if(!isset($nivel_dir)){

```



```

$nivel_dir="../"; }
include_once($nivel_dir."models/persona.model.php");
class Reporte{//Clase
    var $html;

    function Reporte(){//Constructor    }
    function getDatosMaestro($id){
global $database;
return $database->toarray($database->query("SELECT * FROM `Maestro` WHERE
`idMaestro` = $id")); }
    function getPromedioMaestro($id){
        global $database;
        $query="SELECT Pregunta_idPregunta, Pregunta.descripcion,
AVG(respuesta) as promedio
FROM Detalle_Clase, Respuestas, Pregunta where id_Detalle =
idMateria_Detalle
and Pregunta_idPregunta=idPregunta and Maestro_idMaestro = $id group by
Pregunta_idPregunta ";
        return $database->toarray($database->query($query));    }
    function datosDisponibles($id){
global $database;
$query="SELECT Pregunta_idPregunta, Pregunta.descripcion,
AVG(respuesta) as promedio
FROM Detalle_Clase, Respuestas, Pregunta where id_Detalle =
idMateria_Detalle
and Pregunta_idPregunta=idPregunta and Maestro_idMaestro = $id group by
Pregunta_idPregunta ";
        if(!$database->query($query)){
            return FALSE;
        }else {
            return TRUE;    }    }
    function getPromedioEscuela($pregunta,$escuela){
        global $database;
        $query="SELECT AVG(respuesta) as promedio FROM Detalle_Clase,
Respuestas, Pregunta, Maestro where id_Detalle = idMateria_Detalle and
Pregunta_idPregunta=idPregunta and Pregunta_idPregunta=$pregunta and
`Detalle_Clase`.`Maestro_idMaestro` = `Maestro`.`idMaestro` and
`Maestro`.`id_escuela`=$escuela group by Pregunta_idPregunta";
        return $database->singleField($query);    }
    function getPromedioUniversidad($pregunta){
        global $database;
        $query="SELECT AVG(respuesta) as promedio FROM Detalle_Clase,
Respuestas, Pregunta where id_Detalle = idMateria_Detalle and
Pregunta_idPregunta=idPregunta and Pregunta_idPregunta= $pregunta group by
Pregunta_idPregunta";
        return $database->singleField($query); }
    function htmladd($html){
        $this->html=$this->html.$html;
    }
}

```

```

    }}
    $reporte = new Reporte();
    ?>
<?php
/**
 * bienvenida.php
 * La vista de bienvenida no tiene interaccion y simplemente es para darle
 la bienvenida al estudiante
 * o al maestro, dependiendo del tipo de usuario, te informa de que es lo
 que hace la pagina y te pide 1
 * inicies la evaluacion si eres alumno o en que mires tus resultados en el
 caso del maestro.
 */
if(!isset($nivel_dir)){
$nivel_dir="../";
}
?>
<div style="text-align:center;">
<?php if($persona->esEstudiante()==1){?>

<h2>Bienvenido al proceso de evaluaci&ocute;n de tus docentes</h2>
  <!--h2>Universidad de Navojoa<br-->
  
  <h3>
  Vicerrector&iacute;a acad&eacute;mica<br>
  Direcci&ocute;n de calidad educativa<br>
  </h3>
  <p class="lead">
  Para nosotros es muy valiosa tu opini&ocute;n
de manera responsable,
  </p>
      <p >
      ya que as&iacute; podemos mejorar la calidad
de la ense&#328;nza en nuestra instituci&ocute;n.
      </p>
      <p>
      Por favor sigue las siguientes recomendaciones
para el correcto llenado de esta evaluaci&ocute;n a
tus docentes.</p>
      <p>
      1.- Deber&aacute; responder la totalidad de los
&iacute;tems y/o preguntas para poder guardar su
evaluaci&ocute;n.
      </p>
      <p>
      2.- Si por alguna raz&ocute;n deja sin
responder alg&uacute;n campo, el sistema le indicara
cual de ellos debe llenar o falta.
      </p>

```

```

    <p>
    3.- Una vez llenados todos los campos o preguntas deber&acutes;
presionar el bot&acutes;n enviar.
</p>

```

```

    <blockquote>
    <p>
        A uno dio 5 talentos, y a otro 2, y a otro 1, a cada uno
dio conforme a su capacidad; y luego se fue lejos
    </p>
    <small>Mateo 25:15</small>
    </blockquote>

```

```

<?php } if($persona->esMaestro()==1){?>

```

```

<h2>Bienvenido al Sistema de evaluacion Docente</h2>

```

```

<!--h2>Universidad de Navojoa<br-->

<h3>

```

```

Vicerrector&iacute;a acad&eacute;mica<br>
Direcci&acutes;n de calidad educativa<br>
</h3>

```

```

<p class="lead">

```

```

    Aqui podras ver los resultados de tu evaluacion docente.

```

```

</p>

```

```

    <blockquote>

```

```

    <p>

```

```

    A uno dio 5 talentos, y a otro 2, y a otro 1, a cada uno dio conforme a su
capacidad; y luego se fue lejos

```

```

</p>

```

```

<small>Mateo 25:15</small>

```

```

</blockquote>

```

```

<?php }?>

```

```

</div><!-- /.box -->

```

```

<?php

```

```

/**

```

```

    * Editar.php

```

```

    * Esta vista es para editar los diferentes, catalogos, recibe variables
GET del url y las manda al controlador.

```

```

    * primero busca acciones y luego tipo.

```

```

    */

```

```

//Si aun no se define nivel dir definirlo a este directorio.

```

```

if(!isset($nivel_dir)){

```

```

    $nivel_dir="../";

```

```

}

```

```

//incluir controlador.

```

```

include_once($nivel_dir."controller/editar.controller.php");

```

```

//variable global del controlador

```

```

global $editar_controller;

```

```

//Content goes here....
//si existe la variable accion, mandarla al controlador y imprimir el
resultado
if(isset($_GET['accion'])){
echo $editar_controller->cargarAccion($_GET['accion'],$_GET['id']);

//si existe la variable tipo, mandarla al controlador y imprimir el
resultado
if(isset($_GET['tipo'])){
echo $editar_controller->cargarPagina($_GET['tipo']);
}

}
include_once ($nivel_dir."models/encuesta.model.php");
include_once ($nivel_dir."tutzi/class/template.class.php");
//contenido
global $encuesta;
global $template;
global $database;
$disponible=$encuesta->disponible();
if($disponible==1){
echo "Gracias por contestar la encuesta.";
}
else{
$string1 = "Bienvenido ". $encuesta->nombre."<br>". "Usted tiene:
".$encuesta->materias." Materias.";
echo $template->mostrarAdvertencia("exito", $string1);
?>
<form method="post" action="models/encuesta.model.php">
<input type="hidden" value="<?php echo $encuesta->id_estudiante;?>"
name="matricula" />
<input type="hidden" value="<?php echo $encuesta->escuela;?>"
name="escuela" />
<input type="hidden" value="encuesta" name="tipo" />
<?php
$rows= $encuesta->getCategorias();
foreach ( $rows as $cate ) {
    $no_catm=$cate['NombreCategoria'];
    $idCate=$cate['id_Categoria']."<br>";
?>
<div class="box box-info">
                                <div class="box-header">
                                    <h3 class="box-title"><?php echo
$no_catm=$cate['NombreCategoria'];?></h3>
                                </div>
                                <div class="box-body">

```

```

<div class="table-responsive">
table-hover text-center">
<?php
echo "<thead>";
echo "<tr>";
echo "<th>Pregunta</th>";
$x=1;
$content=$encuesta->getMaterias();
foreach ( $content as $rows ) {
    echo "<th>";
    echo $rows['descripcion'];
    echo "</th>";
    $x++;
}
echo "</tr>";
echo "</thead>";
echo "</thead>";
echo "<tbody>";
$i="a";
$z=0;
$preguntas=10;
$materias=$encuesta->materias;

$array1=$encuesta->getMatCat($idCate);
foreach ( (array) $array1 as $preg_array ) {
?>
<tr>
<td>

<?php echo $preg_array['descripcion'];?></td>
<?php

$materias=$encuesta->getMaterias();

foreach ( $materias as $mat_array ) {
?>
<td>

<select class="form-control input-xsmall" name="<?php echo
$preg_array['idPregunta']; ?>,<?php echo $mat_array['idMateria_Detalle'];
?>" required>
<option value=""> </option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
<option value="4">4</option>
<option value="5">5</option>
</select>
</td>

```

```

<?php
$z++;
    }
    echo "<tr>";
$i++;
$z=0;
} ?>
</tbody>
</table>
</div>
</div><!-- /.box-body -->
</div>
<?php}
?>
    <button type="submit" class="btn green">Enviar</button>

    </form>
<?php
}
?>
<?php
?>
<?php
if(!isset($nivel_dir)){
$nivel_dir="../";
}
$isindex=1;
include_once ($nivel_dir."tutzi/class/template.class.php");
include_once ($nivel_dir."tutzi/template/builder/form.template.php");

global $template;
global $form;
$template->incHead();
$template->cssrewrite("
    body {
        background-color: #222222;
    }
    html {
        background-color: #222222;
    }
");
$form->inicForm("controller/process.controller.php", "post");
$form->addhtml('<div class="body bg-gray">');
$form->addinput("text", "user", "form-control", "Usuario");
$form->addinput("password", "pass", "form-control", "Contraseña");
$form->addinput("hidden", "login", "form-control", "login");
$form->addhtml('</div>');
$form->addbutton("submit", "btn bg-olive btn-block", "Iniciar Session");
$content=$form->render();

```

```

$footer=' <p><a href="#">Olvide mi contraseña</a></p>';
$template->addFormbox("Inciar Session",$content,$footer);
$template->incTail();
echo $template->render();
?>
<?php
include ("view/inc/head.php");
include ("view/inc/header.php");
include ("view/inc/content.php");
//Content goes here....
?>
<?php
include ("view/inc/foot.php");
?>
<?php
/**
 * Database.php
 *
 * Esta clase esta echa para facilitar toda la comunicacion a la base de
datos. Aqui se incluiran clases
 * para editar, insertar, y editar clases.
 */
// incluir archivo con todas las variables necesarias
include ($nivel_dir . "config/conf.php");
class MySQLDB
//Inicio de clase
{
    var $connection; //La conexion a La base de datos
    /* constructor */
    function MySQLDB() {
        /* Crear la conexion a La base de datos */
        $this->connection = new mysqli(DB_SERVER, DB_USER, DB_PASS,
DB_NAME);
        if ($this->connection->connect_errno > 0) {
            die('Unable to connect to database [' . $this->connection-
>connect_error . ']');        }    }
        //Funcion para mandar queries de cualquier parte solo mandar a yamar
        $database->query("su query");
        function query($query) {
            global $database;
            //echo $query;
            $result = $this->connection->query($query);
            // $dbarray = mysql_fetch_array($result);
            if ($mysqli->error) {
                try {
                    throw new Exception("MySQL error $mysqli->error <br> Query:<br>
$query", $mysqli->errno);
                } catch(Exception $e ) {
                    echo "Error No: ".$e->getCode(). " - ". $e->getMessage() . "<br >";

```

```

        echo nl2br($e->getTraceAsString());return $result; } }
    //funcion para seleccionar toda la informacion de una tabla, regresa un
    array con toda la informacion
    function selectAll($table) {
        global $database;
        //echo $query;
        $q = ("SELECT * FROM `" . $table . "`");
        //echo $q;
        $result = $this->connection->query($q);
        //$dbarray = mysql_fetch_array($result);
        return $result; }
    // Esta funcion selecciona la informacion de un campo especifico.
    regresa un array
    function selectField($table, $field, $id) {
        global $database;
        //echo $query;
        $q = ("SELECT * FROM `" . $table . "` WHERE `" . $field . "` = '" .
$id . "'"); //echo $q;
        $result = $this->connection->query($q);
        return $result; }
    //Funcion para convertir el resultado de un query a un array.
    function toArray($data) {
        global $database;
        if(!$data==FALSE){
            while ($rows[] = $data->fetch_assoc());
            array_pop($rows);
            return $rows;}else{echo "ERROR not valid DATA!";} }
    //Esta funcion borra un campo especificandole la tabla el nombre del
    campo y el identificador.
    function deleteField($table, $field_name, $id) {
        global $database;
        $query = ("DELETE FROM " . $table . " WHERE " . $field_name . " = '"
. $id . "'");$result = $this->connection->query($query);
        return $result;}
    function updateField($table, $field, $value, $field2, $id) {
        $q = "UPDATE " . $table . " SET " . $field . " = '$value' WHERE
$field2 = '$id'";return $this->connection->query($q);}
    function insertValues($table, $values) {
        // "INSERT INTO ".TBL_USERS." VALUES ('$username', '$password', '0',
        $uLevel, '$email', $time)"
        $q = "INSERT INTO `" . $table . "` VALUES " . $values . ";";
        // echo $q;return $this->connection->query($q);}
    function exist($query){
        $result=$this->query($query);
        $nrows=$result->num_rows;
        if(!$result || $nrows < 1){
            return FALSE;
        }else if ($nrows>=1){
            return TRUE; }}

```



```

function count($query){
return $query->num_rows;}
function selectSingleField($table,$field,$where,$value){
$query='SELECT '.$field.' FROM '.$table.' WHERE '.$where.'='.$value;
return array_values(mysqli_fetch_array($this->query($query)))[0];}
function singleField($query){
return array_values(mysqli_fetch_array($this->query($query)))[0];}};
/* Crear conexion a base de datos */$database = new MySQLDB;?><?php/**
* Constants.php
* Esta clase esta echa para almacenar todas las direntes variables
globales. */
define("DB_SERVER", "");
define("DB_USER", "");//usuario
define("DB_PASS", "");//contrasena
define("DB_NAME", "");//nombre?>
<?php/**
* Process.php *
* Esta clase cumple con el trabajo de procesar todas las formas.
* */
include("session.class.php");
include("ejercicio.class.php");
include("catalogos.class.php");
//include("catalogos.class.php");

class Process{
/* Constructor de clase */
function Process(){
global $session;
/* Prosesar formas */
if(isset($_POST['login'])){
$this->login(); }
if(isset($_GET['timeout'])){
$this->timeout();
//echo "TIMEOUT!"; }
if(isset($_GET["logout"])){
$this->Logout(); }
if(isset($_GET["ejer_ins"])){
$this->ejer_ins(); }
if(isset($_GET["cuenta_ins"])){
$this->cuenta_ins(); }
if(isset($_GET["set_ejer"])){
$this->set_ejer(); }

function login(){
global $session;
$retval = $session->login($_POST['user'], $_POST['pass']);
if($retval){
header("Location:../main.php?encuesta"); }
else{

```

```

        header("location:../index.php?error=loginfail"); } }
    function Logout(){
    global $session;
    $retval = $session->logout();
    header("Location: ../index.php?logout"); }
function timeout(){
    global $session;
    $nom_usuario=$_SESSION['username'];
    $session->logout();
    header("Location: ../timeout.php?user=$nom_usuario"); }
function set_ejer(){
global $session;
$session->setEjersio($_GET['set_ejer']);
header("Location: ../main.php"); }
function ejer_ins(){
    global $ejercicio;
    $ejercicio-
>insEjercicio($_POST['nom_Empresa'],$_POST['descripcion'],$_POST['rfc'],$_P
OST['direccion'],$_POST['telefono']);
    header("Location: ../main.php?ejercicios"); }
    function cuenta_ins(){
    global $catalogo;
    $catalogo-
>insCuenta($_POST['nombre'],$_POST['descripcion'],$_POST['tipo'],$_POST['ej
ercicio']); }
$process = new Process;?>
<?php /**
 * session.php *
 * Esta clase se encarga de administrar todas las funciones de usuarios.
 */
include("user.class.php");

class Session//Inicio de clase {
    var $username; //Usuario
    var $password; //Contraseña
    var $name; //nombre
    var $userid; //id
    var $userlevel; //nivel
    var $logged_in; //Verdadero o Falso
    var $id_ejercicio; // La id del ejercicio seleccionado
    var $userinfo = array(); //Array con toda la informacion del usuario
        function Session(){
            $this->startSession();}
    function startSession(){
        global $database;
        session_start();
            $this->logged_in = $this->checkLogin(); }
        function checkLogin(){
            global $database,$user;

```

```

        if(isset($_SESSION['username']) && isset($_SESSION['id'])){
            if($user->confirmUserID($_SESSION['username'], $_SESSION['id'])
!= 0){
                unset($_SESSION['username']);
                unset($_SESSION['userid']); return false; }
                $this->userinfo = $user-
>getUserInfo($_SESSION['username']);
                $this->username = $this->userinfo['idUsuario'];
                $this->userid = $this->userinfo['idUsuario'];
                $this->userlevel = $this->userinfo['tipo'];
                return true;}
                else{return false;}}
        function logout(){
            unset($_SESSION['username']);
            unset($_SESSION['id']);
            $this->logged_in = false;}
        function login($subuser, $subpass){
            global $database,$user;
            $result = $user->confirmUserPass($subuser, md5($subpass));
            if($result == 1){
                $field = "user";
                return "wronguser";}
            else if($result == 2){
                $field = "pass";
                return "wronguser";}
            if($result>=1){
                return false;}
            $this->userinfo = $user->getUserInfo($subuser);
            $this->username = $_SESSION['username'] = $this-
>userinfo['idUsuario'];
            $this->userid = $_SESSION['id'] = $this->userinfo['idUsuario'];
            $this->userlevel = $this->userinfo['tipo'];
            return true; } };
        $session = new Session;
class Template
//Inicio de clase{
    var $siteName; //Nombre completo de La Persona
    var $title;
    var $HTML;
    function Template() {
        $this->HTML = " ";}
    function incHead() {
        $this->htmlAdd(file_get_contents('tutzi/inc/head.php'));}

```

```

function incTail() {$this-
>htmlAdd(file_get_contents('tutzi/inc/tail.php'));}
function cssrewrite($css) {$this->htmlAdd("<style>" . $css . "
</style>");}
function addFormbox($title, $header, $footer) {
    $this->htmlAdd('<div class="form-box">
        <div class="header">' . $title . '</div>' . $header . '
            <div class="footer">
                ' . $footer . '
            </div>
        </form>

        /div> /div> ');}
function buildSite() {
    include ("tutzi/inc/head.php");
    //CONTENT HERE
    include ("tutzi/inc/tail.php");    }
function buildLogin() {
    include ("tutzi/inc/head.php");
    include ("tutzi/inc/login.php");
    include ("tutzi/inc/tail.php");    }
function strHeaderMsgs($number) {
    $this->htmlAdd('<li class="dropdown messages-menu">
toggle="dropdown">
        <i class="fa fa-envelope"></i>
        <span class="label label-success">4</span>
    </a>
    <ul class="dropdown-menu">
        <li class="header">Tiene ');
    $this->htmlAdd($number . ' <li>
        <ul class="menu">');    }
function HeaderMsgs_addMsg($img, $title, $msg, $mins) {
    $this->htmlAdd('<li>
    <a href="#">
        <div class="pull-left">
            
                </div>
                <h4>' . $title . '
    <small><i class="fa fa-clock-o"></i>' . $mins . ' mins</small>
                </h4>
                <p>' . $msg . '</p>
            </a>
        </li>');
}
function HeaderMsgs_reder() {$this->htmlAdd('</ul>
</li>
<li class="footer"><a href="#">Mirar todos los mensajes</a></li> </ul>
    </li>');}

```

```

        // return this->$HTML;
        echo $this->HTML;    }
    function render() {
        return $this->HTML;
        $this->HTML = " ";    }
    function mostrarAdvertencia($tipo, $texto) {
        switch ($tipo) {
            case "peligro":
                return '<div class="alert alert-danger alert-dismissable">
<i class="fa fa-ban"></i>
<button type="button" class="close" data-dismiss="alert" aria-
hidden="true">x</button>' . $texto . '</div>';
                # code...
            break;
            case "info": return '<div class="alert alert-info alert-
dismissable">
<i class="fa fa-info"></i>
<button type="button" class="close" data-dismiss="alert" aria-
hidden="true">x</button>' . $texto . '</div>';
            break;
            case "advertencia":
                return '<div class="alert alert-warning alert-dismissable">
                    <i class="fa fa-warning"></i>
                    <button type="button" class="close"
data-dismiss="alert" aria-hidden="true">x</button>
                    ' . $texto . '</div>';
            break;
            case "exito":
                return '<div class="alert alert-success alert-dismissable">
                    <i class="fa fa-check"></i>
                    <button type="button" class="close"
data-dismiss="alert" aria-hidden="true">x</button>' . $texto . '</div>';
            break;
            default:
                # code...

            break;}}
    function htmlAdd($text) {
        $this->HTML = $this->HTML . $text; }}
$template = new Template();?><?php
//tools.class.php
class Tools{
function Tools(){}
function safe_redirect($url){
// Solo usar header redirec si no ah sido usado, para evitar error
//de headers ya redirecionadas.
if (!headers_sent()){
    header('HTTP/1.1 301 Moved Permanently');
    header('Location: ' . $url);

```

```

        // para IE
        header("Connection: close");}
    // Metodo HTML/JAVASCRIPT:
    // Si no funciona en php intentar otros metodos.
    print '<html>';
    print '<head><title>Te estamos redireccionando...</title>';
    print '<meta http-equiv="Refresh" content="0;url=' . $url . '" />';
    print '</head>';
    print '<body onload="location.replace(\'' . $url . '\')">';
    // Si javascript no funciona
    // el usuario podra dar click en un link
    print 'Debes de ser redireccionado a:<br />';
    print '<a href="' . $url . '">' . $url . '</a><br /><br />';
    print 'Si no funciona, porfavor dar click en link.<br />';
    print '</body>';
    print '</html>';
    // salir
    exit; } }
$tools= new Tools();?> <?php /** * User.php *
 * Esta clase se encarga de administrar todas las funciones de usuarios.
 */
// incluir archivo con todas las variables necesarias
include("database.class.php");
    class User//Inicio de clase {
        function addUser($user,$password,$deparment){
            global $database;
            $password=md5($password);
            $database->
>insertValues("users","(NULL,'$user','$password','$deparment')");
            return "El usuario a $user a sido agregado.";        }
        function confirmUserID($user,$id){
            global $database;
            $result=$database->selectField("Usuario", "idUsuario", $user);
            $nrows=$result->num_rows;
            if(!$result || $nrows < 1){
                return 1; //No se encontro el usuario en la base de datos.
                echo "NO SE ENCONTRO AL USUARIO";}
            $dbarray = $result->fetch_assoc();
            /* Validar que el usuario y id sean validas */
            if($id == $dbarray['idUsuario']){
                return 0; //son correctos}
            else{return 2; //ID invalida echo "CONTRASENA INVALIDA!";}
        }
    }
    function getUserInfo($username){
        global $database;
        $q = "SELECT * FROM Usuario WHERE idUsuario = '$username'";
        $result = $database->query($q);
        $nrows=$result->num_rows;

```

```

    /* Ocurrio un error */
    if(!$result || $nrows < 1){
        return NULL;    }
    /* Regresar todo el array con la informacion. */
    $dbarray = $result->fetch_assoc();
    return $dbarray;    }
function confirmUserPass($username, $password){
    global $database;
    $result=$database->selectField("Usuario","idUsuario",$username);
    $nrows=$result->num_rows;
    if(!$result || $nrows < 1){
        return 1;
        echo "No existe el usuario";}
    $dbarray = $result->fetch_assoc();
    if($password == $dbarray['clave']){
        return 0;
        echo "Clave Correcta!";}
    else{
        return 2;
        echo "Clave Inorrecta";}}
};$user = new User; ?>
<?php include_once "tutzi/class/database.class.php";?>
login.php
<body>
<style>
body {
    background-color: #222222;}html{background-color: #222222;}
</style><div class="form-box">
div class="header">Sign In</div><form action="../../index.html"
method="post"><div class="body bg-gray">
    <div class="form-group"> <input type="text" name="userid" class="form-
control" placeholder="User ID"/>
    </div>
    <div class="form-group">
    <input type="password" name="password" class="form-control"
placeholder="Password"/>
    </div>
    <div class="form-group">
    <input type="checkbox" name="remember_me"/> Remember me
    </div>
    </div>
    <div class="footer">
    <button type="submit" class="btn bg-olive btn-block">Sign me in</button>

<p><a href="#">I forgot my password</a></p>

</div>
</form>

```

```

        </div>
    </div>
    <!-- jQuery 2.0.2 -->
    <script
src="http://ajax.googleapis.com/ajax/libs/jquery/2.0.2/jquery.min.js"></scr
ipt>
    <!-- Bootstrap -->
    <script src="../../js/bootstrap.min.js"
type="text/javascript"></script>
</body>

```

head.php

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1" />

    <title><?php echo "TODO TITLE";?></title>
    <meta content='width=device-width, initial-scale=1, maximum-
scale=1, user-scalable=no' name='viewport'>
    <!-- bootstrap 3.0.2 -->
    <link href="tutzi/template/css/bootstrap.min.css" rel="stylesheet"
type="text/css" />
    <!-- font Awesome -->
    <link href="tutzi/template/css/font-awesome.min.css"
rel="stylesheet" type="text/css" />
    <!-- Ionicons -->
    <link href="tutzi/template/css/ionicons.min.css" rel="stylesheet"
type="text/css" />
    <!-- Morris chart -->
    <link href="tutzi/template/css/morris/morris.css" rel="stylesheet"
type="text/css" />
    <!-- jvectormap -->
    <link href="tutzi/template/css/jvectormap/jquery-jvectormap-
1.2.2.css" rel="stylesheet" type="text/css" />
    <!-- fullCalendar -->
    <link href="tutzi/template/css/fullcalendar/fullcalendar.css"
rel="stylesheet" type="text/css" />
    <!-- Daterange picker -->
    <link href="tutzi/template/css/daterangepicker/daterangepicker-
bs3.css" rel="stylesheet" type="text/css" />
    <!-- bootstrap wysihtml5 - text editor -->
    <link href="tutzi/template/css/bootstrap-wysihtml5/bootstrap3-
wysihtml5.min.css" rel="stylesheet" type="text/css" />
    <!-- Theme style -->

```



```

        <link href="tutzi/template/css/AdminLTE.css" rel="stylesheet"
type="text/css" />

        <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
        <!-- WARNING: Respond.js doesn't work if you view the page via
file:// -->
        <!--[if lt IE 9]>
            <script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
            <script
src="https://oss.maxcdn.com/libs/respond.js/1.3.0/respond.min.js"></script>
        <![endif]-->
</head><body>

```

form.template.php

```

<?php

class Form//Inicio de clase {
    var $HTML;
    function Form(){
        $this->HTML=" ";
    }
    function inicForm($action,$type){
        $this->htmlAdd('<form action="'. $action.'" method="'. $type.'">');
    }
    function addinput($type,$name,$class, $placeholder){
        $this->htmlAdd('<div class="form-group">
                        <input type="'. $type.'" name="'. $name.'"
class="'. $class.'" placeholder="'. $placeholder.'" />
                        </div>');
    }
    function addbutton($type,$class,$text){
        $this->htmlAdd('<button type="'. $type.'"
class="'. $class.'">' . $text . '</button>');
    }
    function addhtml($html){
        $this->htmlAdd($html);
    }
    function render(){
        return $this->HTML;
        $this->HTML=" ";
    }
    function htmlAdd($text){
        $this->HTML=$this->HTML.$text;
    }$form = new Form();?>

```

REFERENCIAS

- Amo, F. A., Martínez Normand, L. y Segovia Pérez, F. J. (2005). *Introducción a la ingeniería del software*. Málaga: Delta.
- Arce Orozco, J. M. (2010). Fines y modalidades de la evaluación docente en el nivel superior en México. *Revista Digital Universitaria*, 11(6), 3-11. Recuperado de <http://www.revista.unam.mx/vol.11/num6/art60/ar>
- Aumaille, B. (2002). *J2EE desarrollo de aplicaciones web*. Barcelona: Eni.
- Bass, L., Clements, P. y Kazman, R. (2003). *Software architecture in practice*. Reading, MA: Addison-Wesley.
- Bertoni, A., Poggi, M. y Teobaldo, M. (1993). *Evaluación: nuevos significados para una práctica compleja*. Buenos Aires: Kapelusz.
- Booch, G., Rumbaugh, J. y Jacobson, I. (2005). *The unified modeling language user guide*. Reading, MA: Addison-Wesley.
- Briggs, A. y Burke, P. (2005). *A social history of the media: From Gutenberg to the internet*. Malden, MA: Polity.
- Canales Sánchez, A. (2007). Evaluación educativa: la oportunidad y el desafío. *Reencuentro*, 48, 40-46.
- Canales Sánchez, A. (2008). La evaluación de la actividad docente: a la espera de iniciativas. *Revista Electrónica de Investigación Educativa*, 10, 1-20. Recuperado de <http://redie.uabc.mx/NumEsp1/contenido-canales.htm>
- Cancher Castro, C., Ponce de León Amador, P. y Saquete Boró, E. (2006). *Introducción a la programación orientada a objetos*. Universidad de Alicante: Servicio de Publicaciones.
- Catalán, J. y González, M. (2009). Actitud hacia la evaluación del desempeño docente y su relación con la autoevaluación del propio desempeño, en profesores básicos de Copiapó, La Serena y Coquimbo. *Psykhé*, 18(2), 97-112. doi:10.4067/S0718-222820090002000232
- Crovi Druetta, D. (2002). Sociedad de la información y el conocimiento. *La Revista Mexicana de Ciencias Políticas y Sociales*, 45(185), 13-34.
- Daft, L. R. (2007). *Teoría y diseño organizacional* (Vol. 9). Mexico: Cengage Learning.

- De San Martín Oliva, C. R. P. (s. f.). *Metodología ICONIX*. Recuperado de <http://www.portalhuarpe.com.ar/Seminario09/archivos/MetodologiaICONIX.pdf>
- Estudillo García, J. (2001). Surgimiento de la sociedad de la información. *Biblioteca Universitaria*, 4(2), 77-86.
- Fuentes Medina, M. E. y Herrero Sánchez, J. R. (1999). *Evaluación docente: hacia una fundamentación de la autoevaluación*. *Revista Interuniversitaria de Formación del Profesorado*, 2(1), 354-368. Recuperado de http://maxconn.aufop.com/aufop/uploaded_files/articulos/1224341555.pdf
- Horbath, J. E. y Gracia, M. A. (2014). La evaluación educativa en México. *Revista de Relaciones Internacionales, Estrategia y Seguridad*, 9(1), 59-85.
- International Organization for Standardization. (2011). *Concepto de calidad*. Recuperado de <http://iso-actual.blogspot.com/p/iso-8402.html>
- Izquierdo, L. R. (s. f.). *Introducción a la programación orientada a objetos*. Recuperado de <http://luis.izqui.org/resources/ProgOrientadaObjetos.pdf>
- Jalloul, G. (2004). *UML by example: sigs: Advances in object technology s*. Cambridge: Cambridge University Press.
- Loredo Enríquez, J., Romero Lara, R. e Inda Icaza, P. (2008). Comprensión de la práctica y la evaluación docente en el posgrado a partir de la percepción de los profesores. *Revista Electrónica de Investigación Educativa, Especial*. Recuperado de <http://redie.uabc.mx/NumEsp1/contenido-loredoromeroinda.html>
- Luján Mora, S. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web*. Alicante: Club Universitario.
- Lutz, M. (2009). *Learning python* (5ª ed.). Sebastopol, CA: O'Reilly Media.
- Márquez Jiménez, A. (2016). ¿Qué podemos esperar de la evaluación docente? *Perfiles Educativos*, 38(152), 3-11.
- Merino Balladares, K. V. y Aguirre Ponce, I. J. (2015). *Desarrollo de una aplicación web para la administración y control de los procesos de venta de tickets dirigido para teatros* (Tesis de licenciatura). Universidad Católica de Santiago de Guayaquil, Guayaquil, Ecuador.
- Morales, R. (2014). *Lenguajes de programación: ¿qué son y para qué sirven?* Recuperado de <https://colombiadigital.net/actualidad/articulos-informativos/item/7669-lenguajes-de-programacion-que-son-y-para-que-sirven.html>
- Organización para la Cooperación y el Desarrollo Económicos. (2009). *Evaluating and rewarding the quality of teachers*. Recuperado de https://my.vanderbilt.edu/performanceincentives/files/2012/10/Springer_Balch_OECD_Chapter1.pdf

- Osés Bargas, R. M., Duarte Briceño, E. y Esquivel Alcocer, L. A. (2007). Calidad del desempeño docente en una universidad pública. *Sapiens: Revista Universitaria de Investigación*, 8(1), 11-22.
- Palomo Duarte, M. (s. f.). *Programación en PHP a través de ejemplos*. Recuperado de http://servicio.uca.es/softwarelibre/publicaciones/apuntes_php
- Peres Núñez, W. e Hilbert, M. R. (2009). *La sociedad de la información en América Latina y el Caribe*. Santiago de Chile: CEPAL.
- Pérez González, A. M. (s.f.). Web 2.0 ¿Para qué? ¿Cómo nos puede ayudar la web 2.0 a cambiar las organizaciones? Recuperado de <http://files.sld.cu/bmn/files/2017/02/conferencia-web-2.0.pdf>
- Real Academia Española. (2014). *Diccionario de la lengua española* (23ª ed.). Recuperado de: <http://buscon.rae.es/drael/>
- Redondo Duque, M. Á. y Bravo Santos, C. (2005). *Sistemas interactivos y colaborativos en la web*. Castilla La Mancha: Universidad de Castilla La Mancha.
- Rosales Romero, C. E. (2017). *Prototipo de detección de expresiones corporales mediante visión artificial para mejorar la comunicación con niños que tienen parálisis cerebral infantil* (Tesis de licenciatura). Universidad Nacional de Loja, Loja, Ecuador.
- Rueda Beltrán, M., Elizalde Lora, L. y Torquemada González, A. D. (2003). La evaluación de la docencia en las universidades mexicanas. *Revista de la Educación Superior*, 32(127), 71-77.
- Santos Guerra, M. A. (1993). Evaluación educativa: un proceso de diálogo, comprensión y mejora. *Investigación en la Escuela*, 20, 23-35.
- Secretaría de Educación Pública. (2014). *Perfil, parámetros e indicadores para docentes y técnicos docentes y propuesta de etapas, aspectos, métodos e instrumentos de evaluación*. Recuperado de http://servicioprofesionaldocente.sep.gob.mx/content/ba/docs/parametros_indicadores/Completo.pdf
- Sommerville, I. (2005). *Ingeniería del software*. Madrid: Pearson Educación.
- Valencia Rodríguez, W. A. y Vallejo Cardona, J. D. (2015). La evaluación educativa: más que una acción, una cuestión ética. *Revista Virtual Universidad Católica del Norte*, 45, 210-234. Recuperado de <http://revistavirtual.ucn.edu.co/Index.php/RevistaUCN/article/view/668/1199>
- Van Bon, J., De Jong, A., Kolthof, A., Pieper, M., Tjassing, R., Van der Veen, A. y Verheijen, T. (2008). *Fundamentos de la gestión de servicios de TI basada en ITIL® V3*. (Q. Wellington, Trad.). Zaltbommel: Van Haren Publishing (versión en línea). Recuperado de <http://books.google.com.mx/books?id=QHYS9yWDRsQC&printsec=frontcover#v=onepage&q&f=false>

White, E. (1971). *Consejos para los maestros*. Miami: APIA.

White, E. (1995). *Primeros escritos*. Montemorelos: APIA.

White, E. (2005). *Consejos para los maestros*. México: APIA.