



Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

Aplicaciones con la Tecnología Internet

Aplicación web para la gestión de las actividades académico-administrativas, de investigación y de extensión del Instituto de Desarrollo Experimental de la Construcción de la Universidad Central de Venezuela

Trabajo Especial de grado presentado ante la ilustre
Universidad Central de Venezuela

Por el bachiller

Luis Daniel Solórzano Trías

Tutores:

Profa. Yosly Hernández Bieliukas

Prof. Luis Rosales (Fac. de Arquitectura)

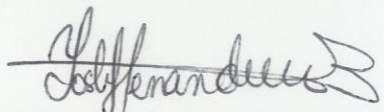
Caracas, julio / 2016

ACTA

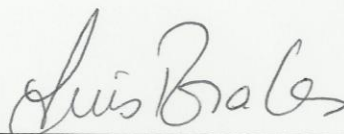
Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para examinar el Trabajo Especial de Grado titulado “**Aplicación web para la gestión de las actividades académico-administrativas, de investigación y de extensión del Instituto de Desarrollo Experimental de la Construcción de la Universidad Central de Venezuela.**” y presentado por el Bachiller Luis Daniel Solórzano Trias C.I: 15.831.532, a los fines de optar al título de Licenciado en Computación, dejamos constancia de lo siguiente:

Leído como fue dicho trabajo, por cada uno de los miembros del jurado, se fijó el día 25 de julio de 2016 a la 2:00pm horas, para que el autor lo defendiera en forma pública, lo que este hizo en el aula PAIII de la Escuela de Computación, mediante una presentación oral de su contenido, luego de lo cual respondió a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobar con la nota de 20 puntos.

En fe de lo cual se levanta la presente Acta, en Caracas el día 25 de julio de 2016



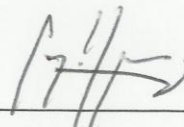
Prof. Yosly Hernández
(Tutora)



Prof. Luis Rosales
(Tutor)



Prof. Luis Hernández
(Jurado)



Prof. Argenis Lugo
(Jurado)

Resumen

El Instituto de Desarrollo Experimental de la Construcción (I.D.E.C) de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela, es un centro de investigación, desarrollo e innovación dedicado a la investigación, la docencia y la extensión.

Los investigadores del I.D.E.C. deben realizar un registro de sus actividades, este registro se basaba en completar unos formularios (diseñados por los docentes del I.D.E.C.), en los cuales se describen las actividades realizadas de investigación, docencia y extensión. Estos formularios son entregados a los jefes de departamento en cada año en los cierres de gestión. Posteriormente los docentes encargados de cada departamento, con la información proporcionada por los investigadores, desarrollarán un informe que refleje y ponga en conocimiento las actividades de coordinación, control, dirección, etc., ejecutadas durante el año, como también los logros obtenidos y dificultades que se presentaron durante la ejecución de tales proyectos, permitiendo reflejar la evolución del Instituto y las perspectivas que se tiene para el siguiente período académico.

Este proceso genera un gran costo para los jefes de departamento ya que deben manipular y clasificar una gran cantidad de información por rubro de cada formulario, para que estos a su vez puedan realizar el informe de gestión del departamento al que pertenecen.

Tomando en cuenta esta situación, el objetivo del presente Trabajo Especial de Grado fue desarrollar una aplicación web de gestión de actividades para docentes del I.D.E.C. (AGADI, por sus siglas en español), que permita optimizar el proceso y permita facilitar el trabajo realizado por los investigadores y principalmente el de los jefes de departamento.

Para la implementación y desarrollo de la aplicación web se aplicó el método de desarrollo de software AgilUs, que está enfocado en el usuario final y en la incorporación de la usabilidad desde etapas tempranas del desarrollo de software.

Palabras Claves: Aplicación web, gestión, actividades, académico-administrativa, Modelado ágil. Metodología de desarrollo AgilUs, extensión, investigación.

Índice General

Introducción	1
Capítulo I. Problema de investigación	4
Contexto de investigación.....	4
1.1 Problema de investigación	8
1.2 Justificación	15
1.3 Objetivo General	16
1.4 Objetivos Específicos.....	16
1.5 Antecedentes	17
1.6 Alcance	17
1.6 Metodología de desarrollo	18
1.6.1 Etapa I: Requisitos.....	18
1.6.2 Etapa II: Análisis	19
1.6.3 Etapa III: Prototipaje	19
1.6.4 Etapa IV: Entrega	19
Capítulo II. Marco Teórico	20
2.1 Aplicaciones Web	20
2.1.1 Definición	20
2.1.2 Ventajas	21
2.1.3 Desventajas	21
2.1.4 Ejemplos de aplicaciones web	22
2.2 Arquitectura Cliente-Servidor.....	25
2.3 Tecnologías web.....	26
2.3.1 Tecnologías del lado del Cliente	26
2.3.2 Tecnologías del lado del Servidor	32
2.3.3 Otras tecnologías	36

2.4 Licencias Creative Commons	38
2.4.1 Licencias de Creative Commons Venezuela	39
2.4.2 Combinación de Licencias de Creative Commons	40
Capítulo III. Marco aplicativo.....	42
3.1 Etapa I: Requisitos	42
3.1.1 Tormenta de ideas	42
3.1.2 Perfiles de usuarios.....	43
3.1.3 Requerimientos funcionales y no funcionales.....	44
3.2 Etapa II: Análisis	45
a) Modelo de caso de uso.....	46
b) Modelo de objetos del dominio	49
c) Modelo físico de la base de datos	51
d) Prototipo en papel.....	54
e) Guía de estilos	56
3.3 Etapa III: Prototipaje.....	58
a) Prototipo rápido	59
b) Evaluación heurística.....	62
3.4 Etapa IV: Entrega.....	65
a) Pruebas de aceptación.....	65
Capítulo IV. Resultados	69
5.1 Funcionalidad de inicio de sesión de AGADI.....	69
5.2 Funcionalidades principales de usuario	71
Conclusiones	87
Recomendaciones.....	89
Referencias.....	90
Anexos A.....	95

Índice de Figuras

Figura 1. Organigrama I.D.E.C.....	5
Figura 2. Formulario de actividades de investigación	10
Figura 3. Formulario de actividades de docencia.....	11
Figura 4. Formulario de actividades de extensión.	12
Figura 5. Formulario de actividades de mejoramiento y capacitación.....	13
Figura 6. Formulario de actividades académico administrativas.	13
Figura 7. Aplicación web de Banesco.....	23
Figura 8. Aplicaciones web del Seniat.....	24
Figura 9. Aplicaciones web de Cinex	25
Figura 10. Arquitectura Cliente-Servidor. Recuperado de:	26
Figura 11. Arquitectura cliente-servidor. Ciclo de vida.....	35
Figura 12. Licencias Creative Commons Venezuela.	41
Figura 13. Tormenta de ideas.....	43
Figura 14. Caso de uso nivel 0.....	47
Figura 15. Caso de uso – nivel 1	47
Figura 16. Actores investigador o profesor / jefe de departamento	49
Figura 17. Modelo de objeto del dominio.....	51
Figura 18. Modelo físico de la base de datos	52
Figura 19. Prototipo de papel del inicio de sesión	55
Figura 20. Prototipo en papel de entrada al portal de usuario.....	56
Figura 21. Menú navbar-inverse de bootstrap.....	56
Figura 22. Paleta de colores menú navbar-inverse de bootstrap.....	57

Figura 23. Vista de botones usados por bootstrap.....	57
Figura 24. Diseño de botones por defecto de Bootstrap	58
Figura 25. Prototipo de la funcionalidad de gestión de actividades.....	59
Figura 26. Prototipo de la funcionalidad de buscar actividad.....	60
Figura 27. Prototipo de la funcionalidad generar reportes.....	61
Figura 28. Prototipo de la funcionalidad de gestión de usuarios.	62
Figura 29. Prueba de aceptación para evaluar la usabilidad de AGADI.....	66
Figura 30. Prueba de aceptación con las primeras cuatro (4) preguntas.....	67
Figura 31. Prueba de aceptación con las cuatro (4) ultimas preguntas	68
Figura 32. Interfaz inicio de sesión.....	70
Figura 33. Portal de bienvenida del usuario administrador.....	71
Figura 34. Funcionalidad gestión de actividades	73
Figura 35. Registro de actividad	74
Figura 36. PDF generado de la funcionalidad de gestión de actividades.....	75
Figura 37. Funcionalidad buscar actividad	76
Figura 38. Filtro de búsqueda por investigador	77
Figura 39. Filtro de búsqueda por nombre de actividad.....	78
Figura 40. Filtro de búsqueda por período.....	79
Figura 41. Filtro de búsqueda por rango de fecha.....	79
Figura 42. Funcionalidad generar reportes.....	80
Figura 43. Ejemplo de reporte de resumen totalizador de proyectos.....	81
Figura 44. Ejemplo de reporte de gráfico de barra para proyectos de investigación	82
Figura 45. Ejemplo de reporte de grafico de tota para proyectos de investigación	83
Figura 46. Funcionalidad administrar, menú gestión de usuarios.....	84

Figura 47. Reporte generado con lista de investigadores.....	85
Figura 48. Formulario de registro de investigador.....	85
Figura 49. Interfaz administrar, opción gestionar períodos	86

Índice de Tablas

Tabla 1. Resultados de la evaluación Heurística.....	64
Tabla 2. Descripción caso de uso de nivel 0 – actor administrador.....	95
Tabla 3. Descripción caso de uso de nivel 0 – actor jefe de departamento.....	95
Tabla 4. Descripción caso de uso de nivel 0 – actor investigador	95
Tabla 5. Descripción opción 1 - caso de uso de nivel 1	95
Tabla 6. Descripción opción 2 - caso de uso de nivel 1	96
Tabla 7. Descripción opción 3 - caso de uso de nivel 1	96
Tabla 8. Descripción opción 4 - caso de uso de nivel 1	97
Tabla 9. Descripción opción 5 - caso de uso de nivel 1	97
Tabla 10. Descripción opción 2.1 - caso de uso de nivel 1	98
Tabla 11. Descripción opción 2.2 - caso de uso de nivel 2.....	98.....96
Tabla 12. Descripción opción 2.3 - caso de uso de nivel 2.....	99
Tabla 13. Descripción opción 2.4 - caso de uso de nivel 2.....	99

Introducción

Internet ha tenido gran auge en las últimas décadas, modificado nuestro estilo de vida, lo cual ha incrementado el desarrollo de aplicaciones web, que ofrecen numerosos beneficios ya que pueden ser accedidos desde cualquier computador conectado a Internet. Todo esto ha provocado que empresas y organizaciones sustituyan sus programas de escritorio por aplicaciones web. (Mora, 2002)

Arias y Portela (1997) afirma: Las organizaciones son concebidas como entidades procesadoras de información, independientemente de su actividad, ya que todas tienen necesidad de obtener y analizar información sobre mercados, costos, ventas y procesos de producción a todo nivel. Esta información procede de fuentes internas y externas, y una vez procesada y utilizada, genera nueva información que será difundida dentro y fuera de la organización. Los sistemas de software que son diseñados para la automatización de procesos, son cada vez más demandados para el control de actividades y procesos dentro de cada organización. Esto se debe a la eficiencia y eficacia que ofrecen estos sistemas para la realización de las actividades que conforman el núcleo principal de las empresas.

Una aplicación web es un programa informático que en lugar de ejecutarse en un ordenador personal, se ejecuta parcialmente en un servidor remoto, al que se accede a través de internet por medio de un navegador web. Con la implantación y popularización del Internet, con el paso del tiempo las aplicaciones web están siendo cada día más usada por la población. Su uso es tan extendido que todos empleamos aplicaciones web casi sin darnos cuenta, como por ejemplo, con el acceso a una cuenta bancaria, en la consulta y envío de correo electrónico, declaración de impuestos, como también consulta y pagos de servicios, entre muchas más utilidades del día a día.

Las dependencias de la Universidad Central de Venezuela (U.C.V.), deben entregar anualmente un informe de gestión con la evaluación académica y administrativa de lo realizado y ejecutado ese año, así como la proyección de planes y actividades futuras, con el fin de que la

U.C.V. en su conjunto rinda cuentas a las autoridades y a la sociedad y a la vez pueda autoevaluarse y proyectarse como institución.

Este trabajo especial de grado se enfocó en el Instituto de Desarrollo Experimental de la Construcción (I.D.E.C), específicamente, en la manera en que los docentes investigadores pertenecientes a dicha dependencia, realizan el registro de sus actividades para posteriormente presentar el informe de gestión de Instituto.

Los docentes investigadores pertenecientes al I.D.E.C. vieron la necesidad de hacer uso de la tecnología con el objetivo de respaldar todas las actividades realizadas por ellos durante cada año, debido a que, en base a esta información, se genera el informe de gestión de cada departamento perteneciente al I.D.E.C. A fin de reducir costos en tiempo y en recursos humanos al momento de procesar tal volumen de información de los cuarenta y cinco (45) investigadores del Instituto, se planteó la realización de este trabajo especial de grado, con la finalidad de brindarle a los docentes de una aplicación web con la que puedan registrar y organizar de manera fácil y eficaz sus actividades y que a su vez luego facilite el procesamiento y análisis de dicha información por los jefes de departamento y el director del I.D.E.C. para desarrollar el informe de gestión.

Este trabajo se encuentra estructurado en cuatro (4) capítulos, tal como se describe a continuación.

Capítulo 1: Problema de investigación. Describe el problema y la solución planteada, además se definen el objetivo general, los objetivos específicos, el alcance y la metodología de desarrollo para el presente trabajo de investigación.

Capítulo 2: Marco Teórico. Se muestran las bases en las que se fundamenta el desarrollo de este trabajo. Se describen las aplicaciones web cliente/servidor, su funcionamiento, arquitectura y características principales. Se presentan las herramientas tecnológicas de software libre utilizadas para realizar esta aplicación, explicando las tecnologías del lado del cliente y del servidor necesarias para el desarrollo de software web.

Capítulo 3: Marco Metodológico. Describe el proceso de desarrollo de esta aplicación a través de la metodología AgilUs, el cual incorpora diferentes fases y actividades de este proceso.

Capítulo 4: Resultados. Muestra los resultados obtenidos una vez finalizado el Trabajo Especial de Grado, describiendo cada una de las funcionalidades e interfaces de la aplicación.

Capítulo I. Problema de investigación

Este capítulo describe la situación que se identificó en el I.D.E.C. en cuanto al registro de actividades llevado a cabo en el Instituto por los docentes que pertenecen al mismo, a su vez se detallan el objetivo general y los específicos, los planteamientos que justificaron la importancia de detectar el problema y el alcance de este TEG.

Contexto de investigación

La presente investigación se llevó a cabo en el I.D.E.C, adscrito a la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela (U.C.V.). “El I.D.E.C. es un centro de investigación, desarrollo e innovación dedicado a la investigación, la docencia y la extensión del entorno construido en las siguientes áreas” (I.D.E.C, 2015):

- Desarrollo Tecnológico
- Habitabilidad de las Edificaciones
- Economía de la Construcción

El I.D.E.C. (2015) publica en su página web la siguiente información del Instituto:

Misión

El I.D.E.C es una institución universitaria dedicada a contribuir e impulsar el desarrollo tecnológico de la construcción bajo criterios de calidad, sostenibilidad y responsabilidad social (I.D.E.C, 2015).

Visión

El I.D.E.C es un Instituto de excelencia de la U.C.V., con amplio reconocimiento en el ámbito nacional e internacional en lo relativo al estudio, la investigación, la innovación y la docencia de la tecnología de la construcción, que posee un capital humano altamente capacitado e instalaciones permanentemente actualizadas para responder a su misión. Se encuentra comprometido en la generación de conocimientos, proyectos, servicios y productos tecnológicos

competitivos que contribuyen al mejoramiento de la calidad de vida de la sociedad y su desarrollo sostenible (I.D.E.C, 2015).

En la figura 1 podemos apreciar el organigrama del I.D.E.C y como está conformado.



Figura 1. Organigrama I.D.E.C (<http://idec.fau.ucv.ve/wordpress/sample-page/organigrama/>).

El I.D.E.C, como se puede apreciar en el organigrama, existen cuatro (4) departamentos, como los son departamento de investigación, el departamento de docencia, el departamento de extensión y el departamento de administración académica. Para este trabajo, nos enfocaremos en los tres primeros departamentos, que son donde los docentes del Instituto se desenvuelven en el día a día. A continuación describirá un poco.

Departamentos de trabajo de los docentes dentro el I.D.E.C:

➤ Departamento de Investigación

El I.D.E.C. (2015) publica en su página web lo siguiente:

El I.D.E.C ha sido precursor en proponer alternativas constructivas económicas, competitivas y de calidad para la producción de edificaciones, desarrollando sistemas y componentes bajo las premisas y estrategias del concepto sistemas de edificaciones y sostenibilidad de la construcción.

Así mismo, la investigación en el Área de Desarrollo Tecnológico ha permitido al I.D.E.C introducir componentes, sistemas constructivos y procesos de producción innovadores aplicados a diferentes en proyectos emblemáticos (I.D.E.C, 2015).

Entre las actividades realizadas por los investigadores del I.D.E.C. en este departamento son:

Actividades:

- a) Realización de proyectos de investigación.
- b) Elaboración de publicaciones arbitradas o no arbitradas.
- c) Realización de otros productos de investigación como informes técnicos, patentes, prototipos, planes pilotos.
- d) Redes de investigación.

➤ **Departamento de Docencia**

El I.D.E.C. (2015) publica en su página web lo siguiente:

El Programa Docente del Instituto de Desarrollo Experimental de la Construcción data de 1986, cuando dio inicio la primera cohorte del Postgrado en Desarrollo Tecnológico de la Construcción. El programa está dirigido a la formación de profesionales de alto nivel, en el área, mediante el desarrollo de estudios de cuarto nivel y el apoyo a la docencia de pregrado (I.D.E.C, 2015).

Se dictan algunas Asignaturas que integran el Plan de Estudios de la Carrera de Arquitectura en la Facultad de Arquitectura y Urbanismo de la U.C.V. Sin embargo el énfasis del programa está en una oferta variada de cursos de postgrado, dirigida a reforzar la formación profesional de Arquitectos, Ingenieros y otros profesionales afines, con alcance nacional e internacional (I.D.E.C, 2015).

Entre las actividades realizadas por los investigadores del I.D.E.C. en este departamento son:

Actividades:

- a) Dictado de asignaturas de postgrado y pregrado.
- b) Cursos de ampliación de conocimientos.
- c) Tutorías en postgrado, pregrado y en pasantías.
- d) Servicio comunitario
- e) Elaboración de material didáctico.
- f) Jurado de trabajos de grado, postgrado y ascensos, arbitraje o evaluación de cursos.
- g) Elaboración de material docente.

➤ **Departamento de Extensión**

El I.D.E.C. (2015) publica en su página web lo siguiente:

Desde su fundación el I.D.E.C se ha distinguido tanto por su novedosa concepción institucional como por sus aportes a favor de la superación de las graves diferencias habitacionales y de servicios públicos del país, a través del desarrollo de tecnologías innovadoras apropiadas al medio social, económico y ambiental, utilización de materias primas nacionales, racionalización de procesos, difusión y formación de personal de alto nivel en esta área. En consecuencia, el marco institucional ha favorecido la exploración de amplios mecanismos orientados a establecer una permanente vinculación nacional e internacional con los ámbitos académicos, socioeconómicos y productivos involucrados con el sector construcción, especialmente con aquellos cuyo objetivo es contribuir a resolver los déficit del hábitat popular (I.D.E.C, 2015).

Entre las actividades realizadas por los investigadores del I.D.E.C. son:

Actividades:

- a) Elaboración de proyectos de aplicación.

- b) Realización de conferencias o ponencias en eventos nacionales o internacionales
- c) Difusión: elaboración de información y materiales divulgativos, exposiciones y eventos.
- d) Comités organizadores de eventos.
- e) Arbitrajes en artículos, ponencias y proyectos.
- f) Comisiones u organizaciones externas representando a la U.C.V.
- g) Redes.
- h) Asistencias a eventos nacionales o internacionales.

1.1 Problema de investigación

Los tres (3) departamentos del I.D.E.C. encargados de las actividades académicas son los de investigación, docencia y extensión. El I.D.E.C cuenta con cuarenta y cinco (45) investigadores, entre investigadores activos, jubilados y contratados. Tres (3) de ellos son jefes de los departamentos antes mencionados.

A fin de cumplir con la entrega del informe de gestión “memoria y cuenta del I.D.E.C.” ante las autoridades del Instituto y de la Facultad de Arquitectura y Urbanismo, los investigadores deben registrar las actividades realizadas durante el año y hacerle llegar esta información a los jefes de cada departamento.

Los investigadores a fin de unificar el formato de registro de sus actividades, implementaron cinco (5) formularios, tres (3) de los departamentos antes mencionados (investigación, docencia y extensión), un cuarto para registrar las tareas de mejoramiento y capacitación de cada investigador y el último dedicado para registrar las funciones académico administrativas.

Estos formularios para ser llenados constan de tres (3) campos, los cuales se detallan y se clasifican por tipo. Por ejemplo, si la actividad a registrar es de investigación se describe la actividad y se especifica si es un proyecto, redes de investigación o una publicación, entre otros tipos de actividades que existen. El siguiente campo a llenar es el promedio de horas

utilizadas para la realización de la actividad, y por último el campo de fecha, en el que se especifica la fecha de inicio y la fecha de fin en que se desarrolló la actividad.

En la figura 2. Se puede observar el formato utilizado para registrar las actividades de investigación. Este permite clasificar la actividad en seis (6) tipos de actividades como son: Proyecto, publicaciones arbitradas, publicaciones no arbitradas, otros productos de investigación (entrando en esta clasificación los informes técnicos, patentes, prototipos, planes pilotos, etc.), otras actividades de investigación y por último redes de investigación.

ACTIVIDADES AÑO 2014

Apellidos:	Nombres:
Dedicación:	Adscrito: IDEC-FAU-UCV
Cargo:	
Area: Desarrollo Experimental de la Construcción	

Categoría actual:	Fecha último ascenso:
Años de Servicio :	Grado Académico:
Fecha Entrega:	

Actividad	Promedio (horas/sem.)	Fecha aprox. (inicio-fin)
-----------	-----------------------	---------------------------

1.- INVESTIGACION:		
1.1. Proyectos:		
1.2. Publicaciones arbitradas (<i>artículos, libros</i>)		
1.3. Publicaciones no arbitradas		
1.4. Otros productos de investigación: (<i>Informes técnicos, patentes, prototipos, planes pilotos, etc.</i>)		
1.5. Otras actividades de investigación		
1.6. Redes de investigación		

TOTAL HORAS AL MES/mes

Actividad	Promedio (horas/sem.)	Fecha aprox. (inicio-fin)
-----------	-----------------------	---------------------------

2.- DOCENCIA:		
----------------------	--	--

Figura 2. Formulario de actividades de investigación

A continuación, en la figura 3, se muestra el formulario de actividades de docencia, el cual, para ser llenados consta de las siguientes clasificaciones por tipo: Dictado de asignaturas (pudiendo ser de postgrado o pregrado), cursos de ampliación de conocimientos, tutorías (de postgrado, pregrado o pasantías), servicio comunitario, elaboración de material

didáctico, jurado de trabajos de grado, postgrado y ascensos, arbitraje o evaluación de cursos, otras actividades y materiales docentes elaborados.

2.1. Dictado de Asignaturas 2.1.1. Postgrado 2.1.2. Pregrado		
2.2. Cursos de ampliación de conocimientos		
2.3. Tutorías: 2.2.1. Postgrado: 2.2.2 Pregrado: 2.2.3 Pasantías		
2.4. Servicio comunitario:		----
2.5. Elaboración de material didáctico:		
2.6. Jurado de trabajos de grado, postgrado y ascensos, arbitraje o evaluación de cursos		
2.7. Otras actividades y materiales docentes elaborados:		
TOTAL HORAS AL MEShoras aprox./mes		

Figura 3. Formulario de actividades de docencia.

En la figura 4 se observa el formato de funciones de extensión el cual se clasifica de la siguiente manera, por proyectos de aplicación, conferencias o ponencias en eventos nacionales o internacionales, asistencia a eventos nacionales o internacionales, comités

organizaciones externas representando a la U.C.V., arbitrajes (de artículos, ponencias, proyectos, etc.), comisiones u organizaciones externas representando a la U.C.V., redes y servicios de divulgación y/o asistencias técnicas.

Actividad	Promedio (horas/sem.)	Fecha aprox. (inicio-fin)
3.- EXTENSION		
3.1. Proyectos de aplicación	---	---
3.2. Conferencias o Ponencias en eventos nacionales o internacionales: Conferencia: Ponencia:		
3.3. Asistencia a eventos nacionales o internacionales		
3.4. Comités organizadores de eventos		
3.5. Arbitrajes (<i>artículos, ponencias, proyectos, etc.</i>)		
3.6. Comisiones u organizaciones externas representando a la UCV.		
3.7. Redes		
3.8. Servicios de divulgación y/o asistencias técnicas.		
TOTAL HORAS AL MES		Horas/mes

Figura 4. Formulario de actividades de extensión.

La figura 5 muestra el formato de actividades de mejoramiento y capacitación. Este clasifica las tareas por estudios de postgrado, cursos de ampliación o extensión, pasantías.

Actividad	Promedio (horas/sem.)	Fecha aprox. (inicio-fin)
4.- MEJORAMIENTO Y CAPACITACION:	---	---
4.1. Estudios de Postgrado	---	---
4.2. Cursos de ampliación o extensión	---	---
4.3. Pasantías	---	---
4.4. Otros	---	---
TOTAL HORAS AL MES horas/mes		

Figura 5. Formulario de actividades de mejoramiento y capacitación.

La figura 6 muestra el formulario de registro de actividades Académico Administrativas, el cual se clasifica en cargos de dirección, cargos de elección, comisiones y asignaciones especiales, entre otras.

Actividad	Promedio (horas/sem.)	Fecha aprox. (inicio-fin)
5.- ACTIVIDADES ACADEMICO ADMINISTRATIVAS:		
5.1. Cargos de Dirección		
5.2. Cargos de elección:		
5.3. Comisiones:		
5.4. Asignaciones especiales		
5.5. Otros		
TOTAL HORAS AL MESHoras aprox/mes		

Firma _____
Profesor

Aprobado _____
Director

Figura 6. Formulario de actividades académicas administrativas.

H
a
y

q
u

e resaltar que para el llenado del formulario son usadas herramientas ofimáticas como el editor de texto Word, para que después el investigador envíe el archivo por correo electrónico a los jefes de cada departamento del Instituto. Ya en este punto cada jefe de departamento clasifica las actividades y extrae las actividades que coordina para desarrollar el informe de gestión de su departamento, el cual será entregado al director(a) del Instituto para su revisión, en base a estos informes, completar el informe de gestión del Instituto.

De este proceso surgen varias dificultades para los investigadores del I.D.E.C., como el manejo del gran volumen de información de los cuarenta y cinco investigadores (45) que deben clasificar y procesar los jefes de departamento, requiriendo gran cantidad de su tiempo y esfuerzo. Otro de los problemas se presenta a nivel de cada investigador, debido que al no contar con una herramienta que les facilite realizar el proceso de registro de sus actividades, no registran dichas tareas mensualmente, lo cual acarrea que se olviden de registrar algunas actividades, influyendo en las estadísticas del informe final.

Una de las dificultades que se les presenta a los docentes del Instituto es que al no contar con un repositorio donde puedan guardar las actividades registradas de años anteriores, se dificulta en gran medida el desarrollo de los informes de gestión, debido a que no se le pueden hacer seguimiento de manera fácil a las actividades con duración de más de un año y necesitando una gran cantidad de tiempo y esfuerzo de recurso humano para publicar estadísticas e información históricas en el informe.

Considerando los problemas antes mencionados en como los investigadores usan herramientas ofimáticas, para registrar sus actividades para luego entregar los formularios en físico o por correo a los jefes de departamento, surge la siguiente interrogante:

¿Cómo mejorar al proceso asociado a la gestión de actividades de los investigadores del I.D.E.C., manteniendo el flujo de trabajo actual y que permita minimizar el tiempo utilizado?

1.2 Justificación

Debido a los problemas que surgen del proceso de registro de actividades, como la gran cantidad de tiempo que deben invertir los investigadores con cargo de jefes de departamento para analizar el gran volumen de información provista por todos los investigadores del I.D.E.C. y el problema que se le presenta a los investigadores al momento de registrar las actividades realizadas durante el año, ya que al transcurrir, en algunos casos hasta meses, desde su realización hasta el momento de registro de la actividad, se olvidan de registrar alguna actividad o tienen dificultad para detallar la tarea realizada o especificar su fecha de inicio, fecha de fin o hasta las horas invertidas. Pudiendo este último problema influenciar en las estadísticas del informe de memoria y cuenta del I.D.E.C.

El desarrollo de la Aplicación Web de gestión de actividades para los docentes del I.D.E.C. (AGADI) es de gran importancia, ya que permite a los investigadores del Instituto contar con un repositorio donde puedan registrar las actividades realizadas durante el año y les permita gestionarlas, pudiendo así crear, editar, visualizar y eliminar alguna actividad antes del mismo año. Gracias a esta facilidad los investigadores pueden crear alguna actividad y en cualquier momento ir complementando la información de esta.

Una de las ventajas que traerá la aplicación AGADI es que al contar con un repositorio con información histórica se podrán realizar análisis a la información previamente registrada a través de los años y generar estadísticas que complementen el informe de gestión. La aplicación AGADI proveerá de utilidades como la generación de reportes en tablas y gráficos de los rubros o campos de los formularios que se puedan cuantificar, facilitando así el proceso de desarrollo del informe de gestión anual y acortando el tiempo utilizado por los jefes de departamento para esta tarea.

Entre los beneficios que la aplicación traerá para los investigadores se pueden mencionar:

- Ahorro en el tiempo de elaboración del informe de gestión anual.

- Un portal donde los investigadores puedan registrar, mantener y corroborar sus actividades.
- Permitirá a los jefes de departamento un mayor control sobre las actividades culminadas y pendientes por culminar.
- La aplicación permitirá a los jefes de departamento realizar búsquedas por años, por investigador, por rango de fechas y brindará la opción generar reportes de las búsquedas realizadas.
- Cabe destacar que por ser una aplicación web, los investigadores podrían acceder desde cualquier computador con conexión a internet.

1.3 Objetivo General

Desarrollar una aplicación Web para la gestión de las actividades académicas-administrativas, de investigación, docencia y de extensión del I.D.E.C.

1.4 Objetivos Específicos

Los objetivos específicos para dar cumplimiento al objetivo general son:

- a) Realizar el levantamiento de la información e identificar los requerimientos funcionales y no funcionales.
- b) Seguir un método de desarrollo de software ágil, para lograr un diseño claro, limpio y que permita una aplicación usable.
- c) Determinar la arquitectura necesaria para la solución propuesta.
- d) Diseñar una base de datos que permita almacenar la información generada por los investigadores al registrar sus actividades, basado en la información de los formularios físicos de registro de actividades diseñados previamente por los docentes del I.D.E.C.
- e) Plantear un módulo que permita gestionar las actividades realizadas por los docentes del I.D.E.C.

- f) Desarrollar un módulo que permita a los jefes de departamento realizar búsquedas sobre un departamento y filtrar por campos claves de la actividad.
- g) Implementar un módulo para generar reportes en formato PDF.
- h) Aplicar pruebas de evaluación heurística y pruebas de aceptación de la aplicación, con la finalidad de comprobar el correcto funcionamiento de la aplicación web desarrollada.

1.5 Antecedentes

Indagando sobre trabajos previos, se encontró con un proyecto desarrollado durante el período 2008 -2011, denominado como Base de Datos Calder, el cual es un sistema de información para la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela. El cual tuvo como objetivo incluir todas las coordinaciones y dependencias de dicha Facultad, de manera tal que estas pudiesen compartir la información que cada una maneja con las demás, para facilitar las tareas y procesos administrativos que cada una realiza de manera independiente, además de automatizar todos los procedimientos administrativos que son realizados de forma manual. La Base de Datos Calder cuenta con seis (6) módulos, los cuales son: Módulo del administrador, módulo de la coordinación de investigación, módulo de la coordinación de docencia, módulo del profesor, módulo de la coordinación administrativa y por último el módulo del decano.

1.6 Alcance

El presente trabajo de investigación consiste en el desarrollo de funcionalidades que permitan que los investigadores del I.D.E.C. realicen un registro de todas las actividades realizadas durante el año, del departamento de investigación, docencia y extensión, además registrar actividades de mejoramiento y capacitación y actividades académico administrativas. En vista de las inquietudes planteadas por el grupo de docentes del Instituto, se plantea cubrir las necesidades de los jefes de departamento e implementar funcionalidades que doten a los mismos de herramientas para buscar, organizar y clasificar el gran volumen de actividades, y a su vez permitir agrupar tareas y generar reportes exportándolas en formato PDF, igualmente se plantea

facilitar el trabajo de generación del informe de gestión permitiendo generar reportes en tablas y gráficos, acortando en gran medida el tiempo utilizado para realizar estos reportes. Además la aplicación en vista que contará con un repositorio dedicado a registrar cada una de las actividades, proporcionará y facilitará el análisis de información histórica, lo cual influirá drásticamente en la toma de decisiones de las autoridades del Instituto. Siendo el desarrollo de estas funcionalidades el objetivo de este trabajo especial de grado.

1.7 Metodología de desarrollo

Para el desarrollo de la aplicación basada en tecnología web para la gestión de actividades de los docentes investigadores (AGADI) del I.D.E.C de la Facultad de Arquitectura y Urbanismo de la U.C.V, se optó por aplicar la metodología AgilUs, con el objetivo de que el desarrollo de la aplicación fuera iterativo e incremental y diera paso a prototipos de alta fidelidad con interacción continua del usuario, propiciando un intercambio cara a cara, lo cual resulta fundamental al momento de realizar evaluaciones de usabilidad y es esta retroalimentación la que guiará el desarrollo del prototipo de la aplicación y dará como resultado una aplicación adaptada a las necesidades de los investigadores del I.D.E.C con un alto índice de usabilidad.

Acosta (2011) propone las siguientes cuatro (4) etapas para la implementación de la metodología AgilUs:

1.6.1 Etapa I: Requisitos

Se realiza el análisis global del problema a solucionar, se estudia productos similares existentes, se genera el perfil de usuario, y se define la lista de requerimientos a desarrollar. Esta etapa es importante en el desarrollo del software, ya que un mal análisis de requisitos traería como consecuencia un software que no cumple con las necesidades e incluso con las expectativas del usuario.

Los artefactos propuestos por la metodología en la etapa de requisitos son: tormenta de ideas, encuestas, entrevistas, perfiles de usuario, y requerimientos funcionales y no funcionales.

1.6.2 Etapa II: Análisis

En esta etapa se realiza un análisis de la solución a desarrollar, se emplean diagramas de casos de uso y modelo de objetos del dominio, siguiendo la notación UML, para definir las funcionalidades que tendrá el producto a desarrollar.

Los artefactos propuestos por la metodología para la etapa de análisis son: modelo de casos de uso, modelo de objetos del dominio, prototipos en papel, guías de estilos.

1.6.3 Etapa III: Prototipaje

Se busca implementar un prototipo rápido de la interfaz de usuario a partir de los patrones de interacción, el cual va evolucionando hasta convertirse en el producto final, se determina la guía de estilo, y se realizan evaluaciones de usabilidad apropiadas a esta etapa: las evaluaciones heurísticas y la lista de comprobación.

Los artefactos propuestos por la metodología para la etapa de prototipaje son: prototipo rápido, evaluación heurística.

1.6.4 Etapa IV: Entrega

Se aplican las pruebas al sistema para certificar que la aplicación desarrollada sea un software usable y sin errores, finalmente se pone en producción la aplicación.

Los artefactos propuestos por la metodología para la etapa de entrega son: pruebas de aceptación y aplicación a liberar.

En este capítulo se presentó el problema antes identificado sobre el manejo del gran volumen de información que debe ser clasificada y procesada por los jefes de cada departamento del Instituto, con base a lo que se planteó en los objetivos para dar respuesta y desarrollar la aplicación web. Seguidamente se detalla, en el Capítulo II, las bases conceptuales de las aplicaciones, tecnologías, frameworks y licencias que permitieron el desarrollo de AGADI.

Capítulo II. Marco Teórico

En este capítulo se definirá qué son las aplicaciones web, sus características, como también se enunciarán las ventajas y desventajas de dichas aplicaciones. Se detallará la arquitectura cliente-servidor utilizada para este trabajo especial de grado, como el patrón de desarrollo de aplicaciones web modelo, vista, controlador (MVC) utilizado por el *framework* usado para el desarrollo de la aplicación, se describirán las tecnologías utilizadas para crear la aplicación web AGADI (Aplicación de Gestión de Actividades para Docentes del IDEC), del lado del cliente como, por ejemplo, HTML, CSS y JavaScript y del lado del servidor, el frameworks Ruby on Rails. Y por último, se explicará el sistema manejador de base de datos utilizado.

2.1 Aplicaciones Web

Internet y la web han influido enormemente tanto en el mundo de la informática como en la sociedad en general. Si nos centramos en la web, en poco menos de diez (10) años ha transformado los sistemas informáticos: Ha roto las barreras físicas (debido a la distancia), económicamente y lógicas (debido al empleo de distintos sistemas operativos, protocolos, etc.) y ha abierto todo un abanico de nuevas posibilidades. Una de las áreas que más expansión está teniendo en la web en los últimos años son las aplicaciones web.

Las aplicaciones web permiten la generación automática de contenido, la creación de páginas personalizadas según el perfil del usuario o el desarrollo del comercio electrónico. Además, una aplicación Web permite interactuar con los sistemas informáticos de gestión de una empresa, como puede ser gestión de clientes, contabilidad o inventario, a través de una página Web (Luján, 2002).

2.1.1 Definición

Una aplicación web consiste en una aplicación en la cual un usuario, por medio de un navegador, realiza peticiones a una aplicación remota accesible a través de Internet (o a través de una intranet) y que recibe una respuesta que se muestra en el propio navegador (Mora, 2002).

Actualmente se ha incrementado el desarrollo de aplicaciones para el entorno web en función de diversos lenguajes de programación, utilizando las distintas alternativas propuestas y tecnologías presentes. Los cambios y las mejoras realizadas en una aplicación web son incorporados de manera casi instantánea (a diferencia de las aplicaciones de escritorio, las cuales deben instalarse y actualizarse en cada computadora de manera individual).

Las aplicaciones web se desarrollan bajo una arquitectura cliente- servidor, donde el cliente, es el equipo que solicita los recursos, equipado con una interfaz de usuario (generalmente un navegador web) para la presentación. El servidor (también denominado software intermedio), es el encargado de proporcionar los recursos solicitados

2.1.2 Ventajas

Según Mora, (2002) las ventajas de las aplicaciones web son las siguientes:

- a) Se reduce drásticamente el problema de gestionar el código en el cliente. Todos los cambios, tanto de interfaz como de funcionalidad, que se deseen realizar a la aplicación se realizan cambiando el código que reside en el servidor web.
- b) La segunda ventaja relacionada con la anterior, es que se evita la gestión de versiones. Se evitan problemas de inconsistencias en las actualizaciones, ya que no existen clientes con distintas versiones de la aplicación.
- c) Otra de las ventajas es que si la empresa ya está usando internet, no necesita comprar ni instalar herramientas adicionales para los clientes.
- d) Una de las ventajas importantes es la independencia de plataforma. Para que una aplicación web se pueda ejecutar en distintas plataformas (hardware y sistemas operativos), solo se necesita disponer de un navegador para cada una de las plataformas, y no es necesario adaptar el código de la aplicación a cada una de ellas. (p. 54).

2.1.3 Desventajas

Dentro de las desventajas se pueden apreciar las planteadas por Mora (2002).

- a) Una de las desventajas, que sin embargo está desapareciendo rápidamente, es que la programación en la web no es tan versátil o potente como la tradicional. El lenguaje HTML presenta varias limitaciones. Sin embargo con HTML5 y la aparición de nuevas tecnologías de desarrollo JavaScript, esta limitación tiende a desaparecer.
- b) La compañía, teóricamente, puede rastrear todo lo que los usuarios hacen. Esto puede causar problemas de privacidad.
- c) Las aplicaciones Web requieren absolutamente navegadores web compatibles. Si un vendedor navegador decide no aplicar una determinada característica, o abandona una determinada plataforma o versión del sistema operativo, esto puede afectar a un gran número de usuarios. (p. 54).

2.1.4 Ejemplos de aplicaciones web

Las aplicaciones web son de gran importancia, ya que son una herramienta de uso global y de amplia difusión gracias al internet, debido a que una aplicación de este tipo puede estar disponible las veinticuatro (24) horas del día, los treientos sesenta y cinco (365) días del año. Las aplicaciones basadas en web no necesitan ser descargadas, instaladas y configuradas, proporcionando inmediatez al momento de acceder a las funcionalidades de la aplicación. Las aplicaciones son sin lugar a dudas muy versátiles debido a que existen diferentes aplicaciones con distintas funcionalidades dirigidas a varios perfiles de usuarios.

A continuación se muestran tres ejemplos de aplicaciones web, las cuales son bastante conocidas y utilizadas en Venezuela.

a) Aplicación web del banco Banesco

El banco Banesco es una institución financiera que funciona en países como Venezuela, Panamá, Puerto Rico, Colombia, República Dominicana, Estados Unidos y España. Banesco dispone de una aplicación web llamada BanescOnline, la cual es un servicio de banca en línea que te permite realizar operaciones bancarias a través de cualquier dispositivo electrónico con acceso a Internet (<http://www.banesco.com/>).

En la figura 7 se observa el inicio de sesión de la aplicación del banco Banesco.



Figura 7. Aplicación web de Banesco (<https://www.banesconline.com/mantis/Website/Login.aspx>)

b) Aplicación web del Servicio Nacional Integrado de Administración Aduanera y Tributaria

El Servicio Nacional Integrado de Administración Aduanera y Tributaria (SENIAT), es el órgano de ejecución de la Administración Tributaria Nacional de Finanzas, para mejorar la recaudación nacional. Su principal función es la recaudación y control de los impuestos o tributos; disminuir la evasión fiscal, reducir la morosidad tributaria (<http://cef.seniat.gob.ve>)

El SENIAT cuenta con una aplicación web en la cual se publican noticias, además de permitir con previa autenticación, actualizar tus datos, consultar y actualizar el Registro Único de Información Fiscal de cada persona, como también declarar el impuesto sobre la renta, entre otras funcionalidades.

A continuación se muestra en la figura 8 el portal de inicio de la aplicación web del seniat.



Figura 8. Aplicaciones web del Seniat. (http://declaraciones.seniat.gob.ve/portal/page/portal/PORTAL_SENIAT)

c) Aplicación web de Cinex

Cinex es una cadena de cines de Venezuela con mayor alcance a nivel nacional, que cuenta con veintisiete complejos cinematográficos ubicados en las principales ciudades del país (<http://www.cinex.com.ve>).

En esta aplicación web, Cinex publica noticias, promociones y realizar consultas de películas por estado, por cine, por películas y formato de la película. Cinex brinda la posibilidad de comprar en línea las entradas para la película de preferencia y en algunos casos elegir el asiento.

Se puede observar en la figura 9 el inicio de la aplicación web de Cinex.



Figura 9. Aplicaciones web de Cinex (<http://www.cinex.com.ve/index.php>)

2.2 Arquitectura Cliente-Servidor

Esta arquitectura cliente-servidor es una arquitectura de red en la que cada ordenador o proceso en la red es cliente o servidor. Las aplicaciones web se basan en una arquitectura cliente-servidor. Por un lado está el cliente (el navegador) y por el otro lado el servidor (el servidor web) (mora, 2002).

El cliente es un programa con el que interacciona el usuario para solicitar a un servidor web el envío de los recursos que desea obtener mediante HTTP (mora, 2002).

Un servidor es un computador que ofrece sus servicios al resto de equipos conectados. Suelen tener una presencia estable en la red, lo que se concreta en tener asignadas direcciones IP permanentes (Cobo, Gómez, Pérez y Rocha, 2005).

En la Figura 10 se muestra una representación de la interacción entre el cliente y el servidor, en la cual ambos están conectados a través de internet, el cliente inicia el ciclo haciendo una petición http, la cual recibe y procesa el servidor, para luego enviar su respuesta http a través de internet hasta llegar al cliente.

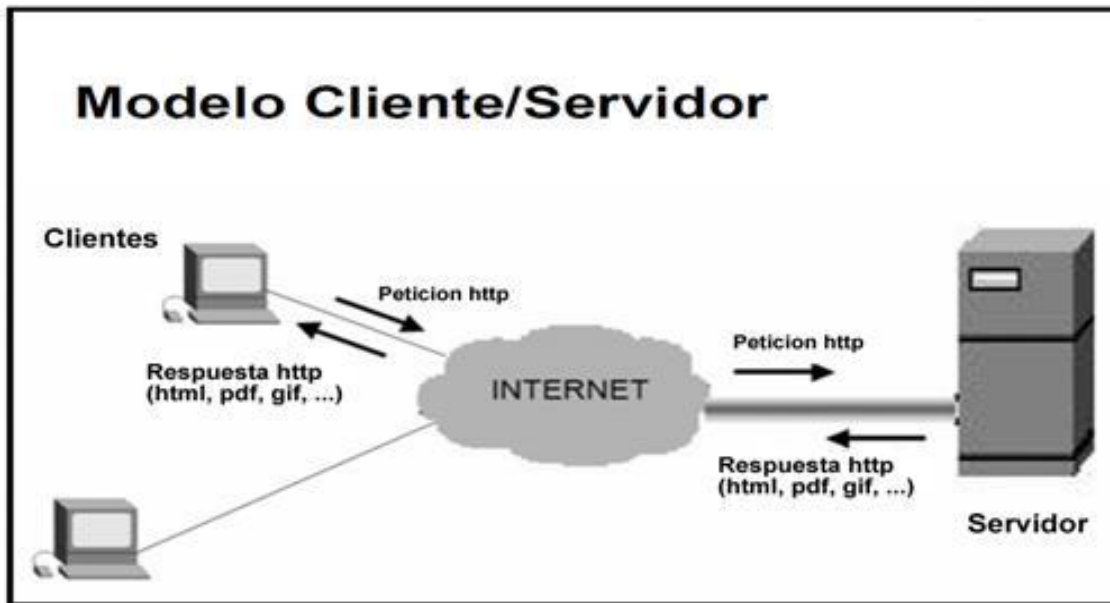


Figura 10. Arquitectura Cliente-Servidor. Recuperado de:
<https://eddiewebdotorg.wordpress.com/2012/09/19/modelo-cliente-servidor/>

Generalmente un servidor puede tratar múltiples peticiones (múltiples clientes) al mismo tiempo. Si se aplica el principio de distribución, cada máquina puede cumplir el rol de servidor para algunas tareas y el rol de cliente para otras. Los clientes y los servidores pueden estar conectados a una red local o una red amplia, como la que se puede implementar en una empresa o a una red mundial como lo es Internet.

2.3 Tecnologías web

A continuación se presentan las tecnologías que son utilizadas para desarrollar la Aplicación Web de este trabajo (AGADI). Entre ellas se encuentran:

2.3.1 Tecnologías del lado del Cliente

Son las que están insertadas en la página HTML del cliente y son interpretadas y ejecutadas por el Navegador. Estas son utilizadas fundamentalmente para mostrar la información y dar estética al Sitio Web. Las tecnologías utilizadas en el desarrollo de AGADI son HTML, CSS y JavaScript.

Estas tres (3) tecnologías fueron de gran importancia para el desarrollo de AGADI, debido a que el HTML aporta la estructura que debe seguir el navegador para interpretar el contenido por lo

cual es lo fundamental a la hora de desarrollar una aplicación de este estilo, el CSS define el aspecto y la presentación de la aplicación, permitiendo separar el contenido de la presentación, lo cual ayuda a desarrollar aplicaciones más ordenadas y facilita el trabajo. Por último tenemos JavaScript el cual aporta dinamismo a las aplicaciones, permitiendo que la aplicación responda a eventos activados por el usuario, como por ejemplo menú desplegable. A continuación se detallarán más a fondo estas tres (3) tecnologías.

2.3.1.1 Lenguaje de Marcado de Hipertexto

HTML es el acrónimo de Lenguaje de Marcado de Hipertexto (*HyperText Markup Language*, por sus siglas en inglés) y es el lenguaje que se utiliza para crear las páginas web. Es uno de los principales componentes de la plataforma web abierta (W3C, 2015).

HTML es un lenguaje de etiquetas. Indica al navegador como tiene que mostrar el contenido. El HTML separa el "contenido" (palabras, imágenes, audio, video, etc.) de la "presentación" (la definición del tipo de contenido y las instrucciones de cómo esos contenidos tienen que mostrarse). El HTML emplea un conjunto de elementos predefinidos que permiten identificar los distintos tipos de elementos. Estos elementos contienen una o más etiquetas que contienen o expresan el contenido. Estas etiquetas suelen ir encapsuladas entre los símbolos <>, y las etiquetas de cierre (que indican el final de un determinado contenido) están precedidas por una barra / (developer mozilla Network).

El lenguaje HTML indica a los navegadores cómo deben mostrar el contenido de una página web. Los navegadores se encargan de interpretar el código HTML de los documentos, y de mostrar a los usuarios las páginas web resultantes del código interpretado. HTML se creó en un principio con objetivos meramente divulgativos. No se pensó que la web llegaría a ser un área de ocio con carácter multimedia, de modo que, el HTML se creó sin dar respuesta a todos los posibles usos que se le iba a dar y a todos los colectivos de gente que lo utilizarían en un futuro. Sin embargo, pese a esta deficiente planificación, se han ido incorporando modificaciones poco a poco con el tiempo, y ahora todo eso encapsulado se transformó en estándares del HTML.

HTML5 Es la nueva versión del lenguaje de marcado que se usa para estructurar páginas web, actualmente en desarrollo, que surge como una evolución lógica de las especificaciones anteriores con los siguientes objetivos: separar totalmente la información, y la forma de presentarla, resumir, simplificar y hacer más sencillo el código utilizado, incorporar nuevas etiquetas semánticas, páginas compatibles con todos los navegadores web incluyendo los de los teléfonos móviles y otros dispositivos utilizados en la actualidad para navegar en Internet. En definitiva representa un importante paso adelante hacia el "diseño universal", es decir, diseño que puedan utilizar todas las personas, en la mayor medida posible, sin necesidad de adaptación ni diseño especializado (Diez, Domínguez, Martínez, Sáenz, 2012).

2.3.1.2 Hojas de Estilo en Cascada

CSS es el acrónimo de Cascading Style Sheet, es decir hojas de estilo en cascada, según la (W3C) “Es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos”.

La aplicación más común de CSS es permitir la separación de los contenidos de los documentos escritos en HTML, XML, XHTML, SVG, o XUL de la presentación del documento con las hojas de estilo, incluyendo elementos tales como los colores, fondos, márgenes, bordes, tipos de letra..., modificando la apariencia de una página web de una forma más sencilla, permitiendo a los desarrolladores controlar el estilo y formato de sus documentos.

Una de las características más potentes de la programación con hojas de estilo consiste en definir los estilos de todo un sitio web. Esto se consigue creando un archivo donde tan sólo colocamos las declaraciones de estilos de la página y enlazando todas las páginas del sitio con ese archivo. De este modo, todas las páginas comparten una misma declaración de estilos y, por tanto, si la cambiamos, cambiarán todas las páginas (Álvarez, Monteiro, Méndez, 2003).

Ventajas de usar CSS: (Schulz, 2009)

- Menos código y más transparente.
- Facilita las actualizaciones.
- Separación entre el contenido y la presentación, lo que permite modificar la presentación sin alterar el contenido.
- Optimización del tiempo de carga y de tráfico en el servidor.
- Usabilidad, CSS contiene muchas funciones y métodos con los que proporcionar funciones ampliadas a sus visitantes

CSS3 es la última versión de CSS, entre sus ventajas principales en esta nueva versión es la inclusión de nuevas propiedades especialmente en cuanto al aspecto gráfico aunque se prevean mejoras sustanciales en otros medios como el de sonido. La actualización incluye los tan esperados bordes redondeados, textos con sombras, la capacidad de asignar múltiples fondos, un mejor manejo de tablas incluyendo el estilo cebra, multi-columnas, etc. El modelo conservará muchas de las actuales propiedades y trabajará con nuevos selectores.” Según Álvarez (2008) “consiste en la incorporación de nuevos mecanismos para mantener un mayor control sobre el estilo con el que se muestran los elementos de las páginas, sin tener que recurrir a trucos, que a menudo complicaban el código de las web”

Está dividido en módulos y cada uno de estos añade nuevas funciones a las que estaban definidas en CSS2, es decir que las funciones que poseía CSS2 se preservan.

En el 2011 la W3C convirtió en recomendaciones oficiales tres módulos de CSS3 los cuales son selectores, espacios de nombres y color. Entre las nuevas propiedades de CSS3 están los bordes, fondos, color, texto, selectores, modelo de caja básico, degradados, animaciones, propiedades orientadas a discurso o lectura automática de páginas web, creación de múltiples columnas de texto y media queries.

2.3.1.3 JavaScript

JavaScript es el lenguaje interpretado más utilizado, principalmente en la construcción de páginas Web, con una sintaxis muy semejante a Java y a C. Pero, al contrario que Java, no se trata de un lenguaje orientado a objetos propiamente dicho, sino que éste está basado en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad (Flanagan, 2011).

El núcleo de JavaScript contiene un conjunto central de objetos, tales como Array (arreglos), Date (fechas) y Math (objetos matemáticos), además de un conjunto central de elementos del lenguaje tales como los operadores, estructuras de control y sentencias. El núcleo de JavaScript puede ser extendido para una variedad de propósitos complementándolo con objetos adicionales; por ejemplo (Mozilla Developer Network):

- JavaScript del lado Cliente extiende el núcleo del lenguaje proporcionando objetos para el control del navegador (Navigator o cualquier Web browser) y su Modelo Objeto Documento [Document Object Model] (DOM). Por ejemplo, las extensiones del lado del cliente permiten a una aplicación ubicar elementos en un formulario HTML y responder a los eventos de usuario tales como los clics del mouse, entradas del formulario y navegación de páginas.
- JavaScript del lado Servidor extiende el núcleo del lenguaje proporcionando objetos relevantes para la ejecución de JavaScript en un servidor. Por ejemplo, las extensiones del lado del servidor permiten que una aplicación se comunique con una base de datos relacional, proporcionar continuidad de la información desde una invocación de la aplicación a otra o efectuar la manipulación de archivos en un servidor.

JavaScript es el lenguaje de programación de la Web. La inmensa mayoría de los sitios web modernos utilizan JavaScript, y todos los navegadores web modernos en los escritorios, tabletas y teléfonos inteligentes-incluyen intérpretes de JavaScript, haciendo a JavaScript el

lenguaje de programación más omnipresente en la historia. JavaScript es parte de la tríada de tecnologías que todos los desarrolladores web deben aprender: HTML para especificar el contenido de páginas web, CSS para especificar la presentación de las páginas web y JavaScript para especificar el comportamiento de las páginas web (Flanagan, 2011).

JavaScript se está convirtiendo también en el lenguaje "integrador". Lo encontramos en muchos ámbitos, ya no solo en Internet y la Web, también es nativo en sistemas operativos para ordenadores y dispositivos, del lado del servidor y del cliente. Aquella visión de JavaScript "utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página web" se ha quedado muy pequeña.

En el contexto de un sitio web, con JavaScript puedes hacer todo tipo de acciones e interacción. Antes se utilizaba para validar formularios, mostrar cajas de diálogo y poco más. Hoy es el motor de las aplicaciones más conocidas en el ámbito de Internet: Google, Facebook, Twitter, Outlook... absolutamente todas las aplicaciones que disfrutas en tu día a día en la Web tienen su núcleo realizado en toneladas de JavaScript. La Web 2.0 se basa en el uso de JavaScript para implementar aplicaciones enriquecidas que son capaces de realizar todo tipo de efectos, interfaces de usuario y comunicación asíncrona con el servidor por medio de Ajax (DesarrolloWeb).

2.4.1.4 Framework JQuery

JQuery es una biblioteca JavaScript rápida, pequeña y rica en funciones. Hace las cosas como HTML recorrido y manipulación de documentos, gestión de eventos, animación y Ajax mucho más simple con un API fácil de usar que funciona a través de una multitud de navegadores. Con una combinación de versatilidad y capacidad de ampliación, JQuery ha cambiado la forma en que millones de personas escriben JavaScript (Jquery).

Cuando un desarrollador programa aplicaciones web utilizando el lenguaje JavaScript, es común que deba preocuparse por la compatibilidad de los scripts con respecto a los

navegadores, incluyendo código que permita detectar el navegador utilizado por el usuario para realizar una u otra acción dependiendo del mismo. Con JQuery no hay necesidad de tomar en cuenta este problema, ya que esta librería funciona de igual forma en cualquier navegador, esto gracias a la incorporación de una serie de clases que permiten al desarrollador programar sin tomar en cuenta el navegador en el cual se desplegará la aplicación, además el uso de esta librería es no intrusivo ya que está separado de la estructura HTML de la aplicación.

Es importante comentar que jQuery no es el único framework que existe en el mercado. Existen varias soluciones similares que también funcionan muy bien, que básicamente nos sirven para hacer lo mismo. Como es normal, cada uno de los frameworks tiene sus ventajas e inconvenientes, pero jQuery es un producto con una aceptación por parte de los programadores muy buena y un grado de penetración en el mercado muy amplio, lo que hace suponer que es una de las mejores opciones. Además, es un producto serio, estable, bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización del framework. Otra cosa muy interesante es la dilatada comunidad de creadores de plugins o componentes, lo que hace fácil encontrar soluciones ya creadas en jQuery para implementar asuntos como interfaces de usuario, galerías, votaciones, efectos diversos, etc. (DesarrolloWeb).

2.3.2 Tecnologías del lado del Servidor

Permite construir código que se ejecuta en el Servidor Web justo antes de que se envíe la página al cliente a través de la red, existen diversidad de tecnologías del lado del servidor tales como PHP, JSP y Ruby on Rails, en el presente trabajo se destaca Ruby on Rails por ser una tecnología innovadora que facilita la implementación de aplicaciones Web AGADI y minimizando el tiempo de desarrollo.

2.3.2.1 Ruby on Rails

“Ruby on Rails o Rails es un framework para aplicaciones web desarrollado por David Heinemeier Hansson, basado en el lenguaje de programación Ruby” (Ruby-Lang).

La primera versión fue liberada en el año 2004 y actualmente se encuentra en desarrollo de nuevas versiones. Resulta un framework bastante práctico porque ayuda a construir sitios web de manera rápida, con código ordenado y que es fácil de mantener.

“Ruby es un lenguaje de programación interpretado, dinámico, reflexivo y orientado a objetos”. Su creador, Yukihiro “Matz” Matsumoto, mezcló partes de sus lenguajes favoritos (Perl, Python, Smalltalk, Eiffel, Ada, y Lisp) para formar un nuevo lenguaje que incorporara tanto la programación funcional como la programación imperativa, y una sintaxis inspirada en los lenguajes mencionados. Su implementación oficial es distribuida bajo una licencia de software libre; esto quiere decir que no solamente es gratis, sino que también es libre para usarse, copiarse, modificarse y distribuirse. Es considerado un lenguaje flexible, ya que permite a sus usuarios alterarlo libremente; un ejemplo ello es que las partes esenciales de Ruby pueden ser quitadas o redefinidas (Ruby on Rails).

Ruby on Rails es de código abierto y multiplataforma. Proporciona una capa de abstracción de bases de datos de gran alcance llamada Active Record, la cual trabaja con los sistemas manejadores de bases de datos más empleados. Provee un sistema multicapa para organizar los diferentes archivos que conforman la aplicación.

Rails tiene convenciones específicas sobre la estructura de directorios, los nombres de archivos, las estructuras de datos, los argumentos de un método, etc. Cuando se escribe una aplicación utilizando Rails, deben seguirse dichas convenciones estipuladas por el framework.

El framework emplea un enfoque bajo el patrón de diseño MVC el cual será descrito a continuación.

2.3.2.2 Patrón Modelo-Vista-Controlador (MVC) Aplicado en Rails

El patrón MVC separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos: modelo, vista y controlador, los cuales se tratan como entidades separadas y son capaces de diseñarse y probarse de manera aislada.

El modelo consiste en la representación de los datos y las reglas del negocio presentes en la aplicación, los cuales, una vez colocados en contexto del sistema, proveen de información a la aplicación o al usuario. La vista representa el modelo en forma gráfica, es decir, la interfaz de usuario. El controlador se encarga de dirigir y responder todas las solicitudes realizadas por el usuario (Teixidor, 2010).

Como resultado de esta organización en base a tres componentes, el patrón de diseño MVC consigue separar la lógica de negocio (el modelo) y la presentación (la vista), aportando como ventaja un mantenimiento más sencillo de las aplicaciones.

Otra ventaja presente es que al momento de incorporar un cambio en el modelo, no tiene por qué afectar a las vistas, al igual que un cambio en una vista no tiene por qué tener efecto en el modelo. “... un cambio en una aplicación MVC tiende a ser localizado y de bajo impacto, facilitando considerablemente el mantenimiento mientras aumenta el nivel de reutilización entre los componentes” (Carneiro, 2010).

A continuación, en la figura 11, se expone gráficamente el proceso de solicitud-respuesta en un enfoque cliente-servidor, aplicando el esquema MVC.

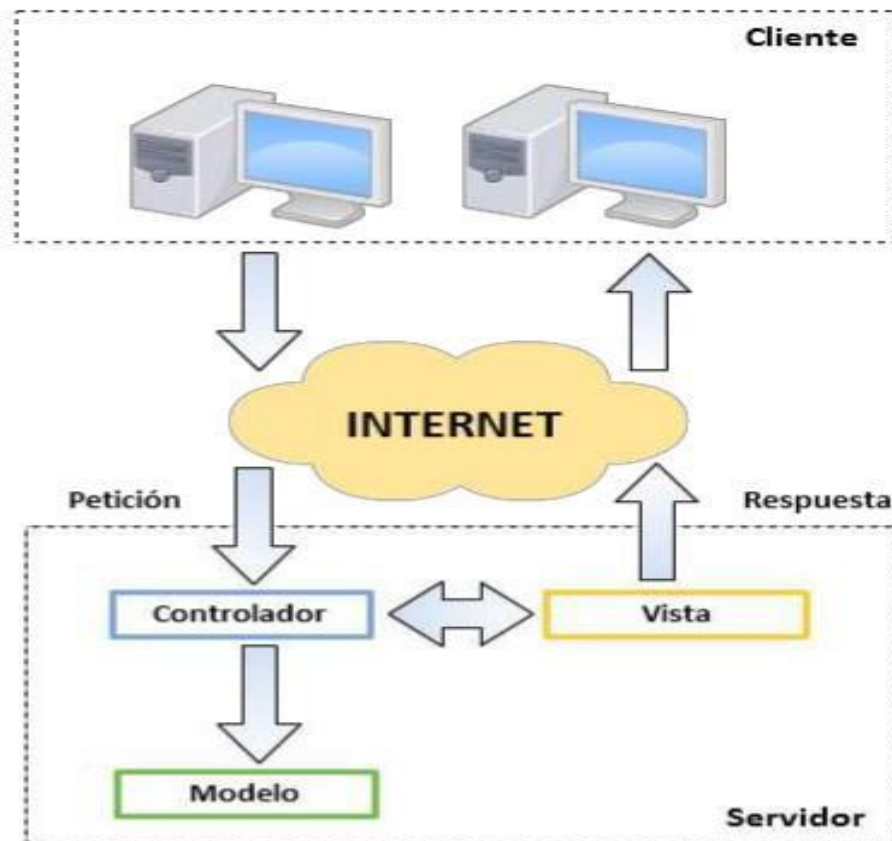


Figura 11. Arquitectura cliente-servidor. Ciclo de vida

Recuperado de: <http://desarrolladorsenior.blogspot.com/2010/10/patron-mvc-modelo-vista-controlador-en.html>

Como se aprecia en la figura 11, el ciclo inicia cuando un usuario se conecta a Internet, a través del protocolo HTTP 4, y realiza una solicitud al controlador. Es tarea del controlador atender dicha solicitud y delegar la tarea al modelo, el cual es el ente encargado de realizar operaciones sobre la información que maneja para cumplir con lo que le delegó el controlador. Una vez que el modelo finaliza su labor, emite una respuesta al controlador con la información obtenida luego de realizar las operaciones necesarias. El controlador debe redirigir dicha respuesta a la vista, quien se encargará de transformar la respuesta en información entendible por el usuario. Una vez realizado esto, la vista transmite la representación gráfica de los datos al controlador, quien se encargará de transmitírsela al usuario que emitió la solicitud. Este ciclo puede llevarse a cabo cuantas veces lo requiera el usuario.

Cada capa de MVC se corresponde con un módulo definido en Rails:

- **Modelo (Active Record):** se encarga de mantener la relación entre un objeto y la base de datos. Es implementado a través de la librería Active Record, la cual provee una interfaz entre las Tablas en una base de datos relacional y el código de la aplicación desarrollada en Ruby, para manipular los registros de la base de datos. Los nombres de los métodos en Ruby son generados automáticamente de los nombres de los campos en las Tablas de la base de datos.
- **Vista (ActionView):** consiste en la representación de los datos en un formato particular. Se implementa mediante la librería ActionView que se encuentra embebida en Ruby, basada en un sistema para definir plantillas de presentación para mostrar la información.
- **Controlador (ActionController):** provee una facilidad dentro de la aplicación para dirigir el tráfico de las solicitudes, por un lado accediendo a los datos a través del modelo, y por otro lado, organizando los datos en una plantilla suministrada por la vista para una presentación de los mismos. Implementa un ActionController, el cual se sitúa entre el Active Record (la interfaz de la base de datos) y el ActionView (el motor de presentación).

2.3.3 Otras tecnologías

2.3.3.1 Sistema Manejador de Base de Datos PostgreSQL

Los objetos generados durante este trabajo especial de grado, mediante el uso de las tecnologías HTML, CSS, JavaScript y demás objetos creados por el framework Ruby on rails,

deben ser almacenados de alguna manera para su posterior consulta, edición y publicación, por lo que es necesario el uso de una base de datos.

Una base de datos es un conjunto estructurado de datos que representa las entidades y sus interrelaciones. La representación es única e integrada, a pesar de que debe permitir utilizaciones diversas y simultáneas (Paré, 2002).

Las bases de datos dependen de un software el cual es el que soportará lógicamente esta estructura, a este software se le conoce como Sistema Manejador de Bases de Datos (SMBD).

Un sistema manejador de bases de datos provee herramientas y servicios que permiten ver, manipular y manejar las bases de datos, servicios tales como manipulación de los datos que se encuentran en la base de datos física, manejo multiusuario de las bases de datos, manejo de las transacciones que se realizan en la base de datos, soporte de comandos que permiten editar y ver datos (comandos llamados queries), respaldo de las bases de datos y mecanismos de seguridad para prever que solo usuarios autorizados vean y manipulen datos (Paré, 2002).

PostgreSQL es un gestor de bases de datos orientadas a objetos (SGBDOO o ORDBMS en sus siglas en inglés) muy conocido y usado en entornos de software libre porque cumple los estándares SQL92 y SQL99, y también por el conjunto de funcionalidades avanzadas que soporta, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales (Ginesta Marc & Mora Oscar, 2007).

PostgreSQL se distribuye bajo licencia BSD, lo que permite su uso, redistribución, modificación con la única restricción de mantener el copyright del software a sus autores, en concreto el PostgreSQL Global Development Group y la Universidad de California. PostgreSQL puede funcionar en múltiples plataformas (en general, en todas las modernas basadas en Unix) y a partir de la versión 8.0, también en Windows de forma nativa.

Para la realización de este Trabajo Especial de Grado se determinó utilizar el sistema manejador de bases de datos PostgreSQL, por ser software libre, evitando así un costo

significativo en la implementación esta AGADI. Además para su elección se tomó en cuenta su amplísima lista de prestaciones que lo hacen capaz de competir con cualquier SGBD comercial:

Está desarrollado en C, con herramientas como Yacc y Lex.

- La API de acceso al SGBD se encuentra disponible en C, C++, Java, Perl, PHP, Python y TCL, entre otros.
- Cuenta con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario.
- Su administración se basa en usuarios y privilegios.
- Sus opciones de conectividad abarcan TCP/IP, sockets Unix y sockets NT, además de soportar completamente ODBC.
- Es altamente confiable en cuanto a estabilidad se refiere.
- Control de concurrencia multi-versión, lo que mejora sensiblemente las operaciones de bloqueo y transacciones en sistemas multi-usuario.
- Soporte para vistas, claves foráneas, integridad referencial, disparadores, procedimientos almacenados, subconsultas y casi todos los tipos y operadores soportados en SQL92 y SQL99.

2.4 Licencias Creative Commons

Las aplicaciones web en las mayorías de los casos son elaboradas por usuarios y no por grandes empresas, especialmente en el ámbito educativo. El objetivo de publicar el contenido en dichas aplicaciones es permitir a los demás usarlos sin restricciones, pero era necesario un marco legal que permitiese su uso sin arriesgarnos a tener futuros problemas con los derechos de la obra. Para llenar este vacío legal nacieron las licencias Creative Commons que veremos en el punto siguiente.

2.4.1 Licencias de Creative Commons Venezuela

Creative Commons diseñó seis tipos diferentes de licencias para garantizar la flexibilidad que no ofrece copyright en el uso de las creaciones, bienes y conocimiento; asegurando el mínimo de restricciones en ellos según sea el caso.

De esta manera CC proporciona un sistema que automatiza la búsqueda de contenidos “comunes” o bajo sus licencias. Así, al licenciar su obra, el creador establece condiciones generales que quedan incorporadas digitalmente a la obra, de manera que un motor de búsqueda puede identificarlas y escoger la que más le convenga.

Para lograr lo anterior CC estableció el conjunto de condiciones, que se mencionan a continuación (CreativeCommons, 2015):

Atribución: permite a otros copiar, distribuir, exhibir, y realizar su trabajo con derechos de autor y trabajos derivados basados en ella - pero sólo si ellos dan crédito de la manera que usted solicite.

Compartir Igual: permite que otros distribuyan trabajos derivados sólo bajo una licencia idéntica a la licencia que rige el trabajo original.

No Comercial: permite a otros copiar, distribuir, exhibir, y realizar su trabajo y trabajos derivados basados en el original pero con fines no lucrativos.

Sin obras derivadas: permite a otros copiar, distribuir, exhibir, y realizar sólo la copia literal del trabajo original, por lo cual no está permitida la generación trabajos derivados basados en él.

Las licencias Creative Commons se diseñaron con la intención de establecer un equilibrio entre los derechos de autor establecidos en las leyes y los derechos del colectivo, lo cual permitió forjar el camino para ir desde “Todos los derechos reservados” a “Algunos derechos reservados” (CreativeCommons, 2015).

Por ser AGADI una aplicación con la finalidad ser utilizada por docentes de una institución educativa, se vio la necesidad de establecer un marco legal que permitiese su uso sin arriesgar tener a futuro problemas con los derechos de esta obra digital. En vista a las facilidades provistas por las licencias CC para ser utilizadas y adaptadas a las leyes venezolanas en su versión 3.0. Siendo estas las licencias de acceso libre altamente compatibles con lo que es el derecho de autor y la legislación nacional. Debido a estas ventajas que proporcionan estas licencias, se determinó la elección de su uso.

2.4.2 Combinación de Licencias de Creative Commons

Creative Commons diseñó seis tipos diferentes de licencias para garantizar la flexibilidad que no ofrece copyright en el uso de las creaciones, bienes y conocimiento, asegurando el mínimo de restricciones en ellos según sea el caso.

En la figura 12 se puede observar los tipos de licencias ofrecidos por Creative Commons

TIPOS DE LICENCIAS CREATIVE COMMONS (CC)	
<p>Reconocimiento (BY)</p>  <p>Permite cualquier explotación de la obra, incluyendo una finalidad comercial, así como la creación y distribución de obras derivadas sin ninguna restricción.</p>	<p>Reconocimiento - NoComercial (BY-NC)</p>  <p>Permite la generación de obras derivadas sin uso comercial de la obra original.</p>
<p>Reconocimiento - NoComercial - CompartirIgual (BY-NC-SA)</p>  <p>No se permite uso comercial de la obra original ni de las posibles obras derivadas, cuya distribución debe hacerse con una licencia igual a la que regula la obra original.</p>	<p>Reconocimiento - NoComercial - SinObrasDerivadas (BY-NC-ND)</p>  <p>No se permite un uso comercial de la obra original ni la generación de obras derivadas.</p>
<p>Reconocimiento - CompartirIgual (BY-SA)</p>  <p>Se permite el uso comercial de la obra y de las posibles obras derivadas, cuya distribución debe hacerse con una licencia igual a la que regula la obra original.</p>	<p>Reconocimiento - SinObrasDerivadas (BY-ND)</p>  <p>Se permite el uso comercial de la obra pero no la generación de obras derivadas.</p>

Venezuela.

Figura 12. Licencias Creative Commons Venezuela.

Para la aplicación implementada en este trabajo especial de grado se le otorgó la licencia CC de reconocimiento – no comercial – compartir igual en donde se le atribuye a otros, el derecho a modificar y ampliar la obra para fines no comerciales, siempre y cuando se le de crédito a sus autores y la licencia de las nuevas obras creadas a partir de la original, posean iguales términos y condiciones a la licencia de la obra original.

A continuación se procede al Capítulo III referente al marco aplicativo en donde se muestran las fases de la metodología AgilUs implementadas en el desarrollo de este trabajo especial de grado.

Capítulo III. Marco aplicativo

En este capítulo se describe el proceso de desarrollo de la aplicación AGADI, el cual se llevó a cabo a través de la metodología AgilUs. Este capítulo se divide en cuatro fases: Requerimientos, Análisis, Prototipaje y Entrega. En primer lugar se describen las actividades y artefactos elaborados en la fase de requerimientos, entre los cuales se tienen tormenta de ideas, perfil de usuario y requerimientos funcionales y no funcionales. En segundo lugar se describen las actividades elaboradas durante la fase de análisis, tales como modelos de casos de uso, objetos del dominio, prototipo en papel y guía de estilo. En tercer lugar se describe el proceso de prototipaje de AGADI, en el cual se describen las dos actividades elaboradas. Por último se describe la fase de entrega, en la cual se colocó en producción la aplicación AGADI, sobre el cual se aplicaron las pruebas de aceptación de la aplicación.

3.1 Etapa I: Requisitos

En la primera fase del desarrollo de software se identifica los requerimientos del usuario a través de la elaboración de los artefactos propuestos por la metodología, los cuales para esta investigación comprenden tormentas de ideas, evaluación de sistemas existentes, como por ejemplo la aplicación Registro nacional de innovación e investigación (RNII, por sus siglas en español), la cual permite registrar los datos fundamentales de todas aquellas personas e instituciones que realizan actividades de innovación e investigación y el sistema de gestión académica (CONEST), el cual permite automatizar la gestión académica de la Facultad de Ciencias de la U.C.V.. Otros artefactos que comprende esta fase es la creación de perfiles de usuario y requerimientos funcionales y no funcionales. A continuación se presentan los resultados obtenidos en esta fase.

3.1.1 Tormenta de ideas

El grupo de desarrollo de la aplicación discutió acerca de las características que se requieren para el desarrollo de interfaces de usuario y se generó un conjunto de características funcionales que AGADI debería proveer al usuario.

En la figura 13 se muestra un resumen de las propuestas que surgieron al aplicar esta técnica.

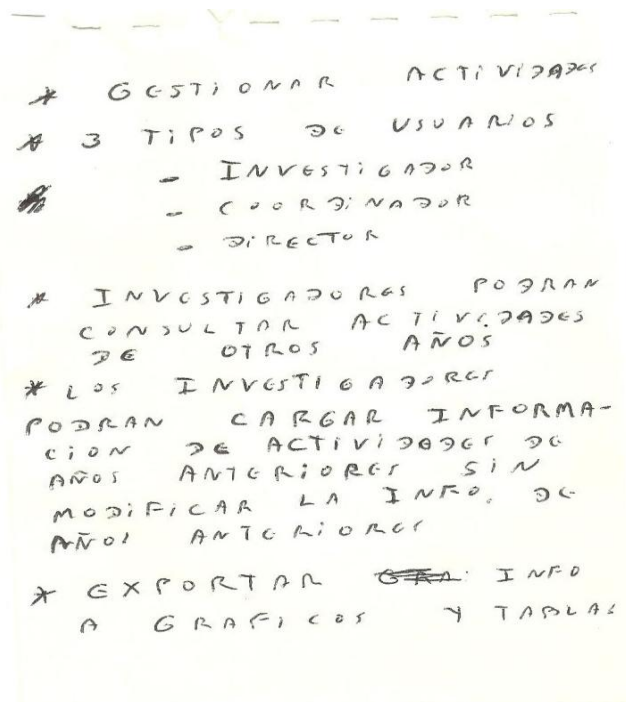


Figura 13. Tormenta de ideas

De aquí se extrajeron las ideas principales acerca de las funcionalidades que debe tener AGADI.

3.1.2 Perfiles de usuarios

Se identificaron los diferentes usuarios potenciales que tendrá la aplicación. AGADI será usada por todos los docentes investigadores pertenecientes al I.D.E.C, cuyos miembros se dividirá en tres grupos de usuarios, los cuales serán investigadores, jefe de departamento y director(a).

a) Perfil investigador:

Este perfil de usuario será el de menos privilegios, los investigadores con este perfil se les mostrará en su portal solo tres (3) funcionalidades principales, como la funcionalidad de gestionar sus actividades, en la cual podrán crear, editar, visualizar y eliminar sus actividades, la segunda funcionalidad principal será la de consulta de sus

actividades, en la que podrán consultar sus actividades aplicando ciertos filtros de búsquedas como consultar actividades por rango de fechas, por períodos o años en los que se creó la actividad, por tipo de actividad de un departamento específico y por último podrán consultar las actividades que concuerden con alguna palabra clave introducida.

La tercera funcionalidad principal permitirá gestionar todo lo relacionado con la cuenta del docente, permitiendo consultar y modificar la información personal del investigador, modificar la contraseña y cerrar la sesión.

b) Perfil jefe de departamento:

Este perfil de usuario tendrá privilegios intermedios, permitirá cuatro (4) funcionalidades principales, como las tres (3) descritas anteriormente (gestionar actividades, cuentas y buscar actividades), y la cuarta funcionalidad brindará la opción de producir reportes en forma de tablas y gráficos los cuales serán importantes a la hora de generar el informe de gestión del departamento, además que concederá realizar esta acción con un ahorro significativo de tiempo.

c) Perfil administrador:

De preferencia se espera que este perfil sea manejado por el director(a), ya que contará con las cuatro funcionalidades del perfil jefe de departamento, además se le concederá una quinta funcionalidad principal, como administrador de la aplicación, donde podrá gestionar los usuarios de la aplicación, pudiendo crear, editar, consultar y eliminar usuarios, además de gestionar los períodos para las actividades, ya que tendrá la opción de crear, editar y eliminar el campo *período*. Cabe destacar que este campo permitirá al administrador controlar el año o período de las actividades que puedan crear los investigadores.

3.1.3 Requerimientos funcionales y no funcionales

A continuación se describen los requerimientos funcionales y no funcionales que acompañan la solución propuesta.

Entre los requerimientos funcionales se tienen los siguientes:

- a) Cada investigador podrá crear, modificar y eliminar actividades del departamento de investigación, docencia, extensión, mejoramiento y capacitación y por último las actividades de académicas administrativas del año en curso.
- b) Los investigadores podrán consultar sus actividades del año en curso como realizar búsquedas históricas de sus actividades.
- c) Crear perfil personal.
- d) Generar reportes luego de realizar las búsquedas de actividades.
- e) Los jefes de departamento podrán consultar sus actividades, como las actividades de los investigadores.
- f) Los jefes de departamento tendrán la posibilidad de realizar consultas, por año, por investigador y por departamento o algún campo específico de dicho departamento.

Entre los requerimientos no funcionales se puede mencionar los siguientes:

Usabilidad, de manera que el usuario pueda comprender y usar con facilidad el módulo; Disponibilidad, se debe permitir usar el módulo en cualquier momento y desde cualquier lugar con acceso a internet; Mantenibilidad, con el objetivo de poder realizar los cambios necesarios en el menor tiempo posible a fin de garantizar la disponibilidad de la aplicación; Portabilidad, para que pueda ser utilizado independientemente de la plataforma que se esté utilizando. Los usuarios deberán quedar satisfechos al emplear la aplicación, gracias a la facilidad y simplicidad de uso de sus pantallas. Seguridad, dicha aplicación web debe reflejar patrones de seguridad teniendo en cuenta la alta sensibilidad de la información que maneja. Además de autenticar a cada usuario que requiera acceso a la aplicación.

3.2 Etapa II: Análisis

Para el desarrollo de la aplicación web (AGADI) fue utilizado el framework Ruby on Rails como herramienta tecnológica para la construcción de sus diferentes módulos de gestión de actividades, debido a que este Framework es muy intuitivo y se cuenta con experiencia previa en

el uso de esta herramienta, lo cual se convierte en una ventaja. Otra de las razones es que por ser software libre contribuye a un desarrollo económico y respaldado debido a la gran cantidad de fuentes de información disponible acerca de esta tecnología.

Debido a que Ruby on Rails utiliza el patrón de diseño modelo MVC (modelo, vista, controlador), conlleva una gran ventaja para la aplicación, ya que permite un mantenimiento sencillo de dicha aplicación, además que facilita el incorporar o modificar una funcionalidad de la aplicación, sin alterar las demás funcionalidades, lo cual aumenta la escalabilidad de la aplicación.

Cabe resaltar que este Framework es completamente compatible con las tecnologías de desarrollo web de uso común como HTML5, CSS, JavaScript, además de ser compatible con el manejador de base de datos Postgresql el cual muy conocido y usado en entornos de software libre, el cual está diseñado para entornos con altos volúmenes de tráfico/transacciones. Otras de las ventajas de usar PostgreSQL es que es multiplataforma.

En esta etapa se lleva a cabo el análisis de los requerimientos obtenidos en la etapa anterior y se generaron algunos artefactos, entre ellos:

a) Modelo de caso de uso

El modelo de casos de uso describe los requisitos funcionales del sistema en términos de las secuencias de acciones, incluyendo las variantes que el sistema u otra entidad puede realizar interactuando con actores del sistema” (Zapata, Carlos, Tamayo, Paula, Arango, Fernando, 2007).

En la figura 14 se muestra el caso de uso nivel 0 generado.

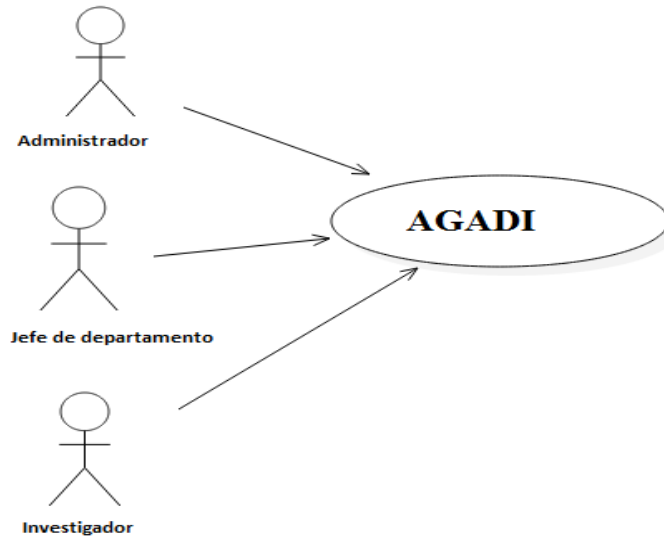


Figura 14. Caso de uso nivel 0

En el anexo A en la página 87 se incluyen las tablas correspondientes al caso de uso nivel 0 para los diferentes actores que van a interactuar con la aplicación. Cada Tabla describe las funcionalidades generales que tendrá cada actor con la aplicación web de gestión de actividades.

En las figura 15 se muestran los casos de uso correspondientes al nivel 1 de abstracción, los cuales presentan las principales funcionalidades de la aplicación y su interacción con los actores.

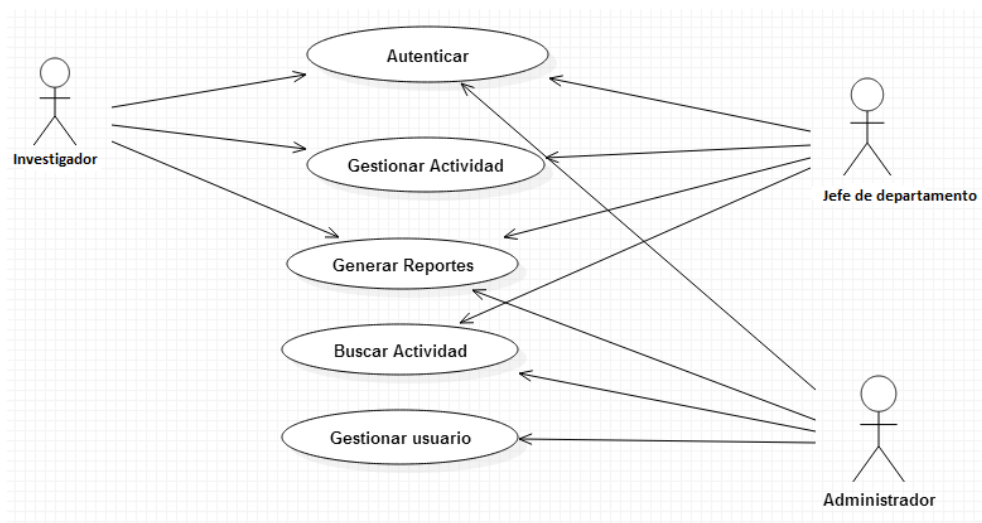


Figura 15. Caso de uso – nivel 1

En el anexo A en las páginas N° 88 y N° 89, se observan las tablas que describen las siguientes interacciones.

El actor investigador y jefe de departamento con la funcionalidad de gestión de actividad, donde dichos actores podrán gestionar las actividades (consultar, eliminar, modificar, crear), luego de señalar el departamento a la cual pertenece la actividad a gestionar.

Actor profesor, jefe de departamento y administrador, con la opción generar reportes, en la cual dichos actores podrán generar reportes de la información de alguna actividad.

La acción de buscar actividad con el actor jefe de departamento o administrador, los cuales podrán búsquedas de actividades de algún formulario o algún campo de dicho formulario.

Y por último la acción de búsqueda con el actor administrador, el cual podrá gestionar usuarios (crear, modificar, eliminar, consultar) y podrá asignarle un perfil de usuario (jefe de departamento o profesor).

En el nivel 2 del caso de uso es la opción gestionar actividades, previamente eligiendo de que departamento es la actividad (investigación, docencia, extensión, mejoramiento y capacitación o actividades académico administrativas). Se despliegan una serie de opciones para poder realizar las funcionalidades que permitirá a la aplicación gestionar dichas actividades, como por ejemplo, poder registrar la actividad, dando como resultado crear la actividad correspondiente. En la función de modificar se podrá actualizar la información de alguna actividad del año en curso (las actividades de años anteriores permanecerán bloqueadas para estas cuatro funcionalidades). En la tercera funcionalidad se podrán eliminar las actividades y por último en la cuarta funcionalidad cada profesor podrá consultar sus actividades.

En las figura 16 se muestra el caso de uso correspondientes al nivel 2 de abstracción, el cual presentan las funcionalidades antes descritas.

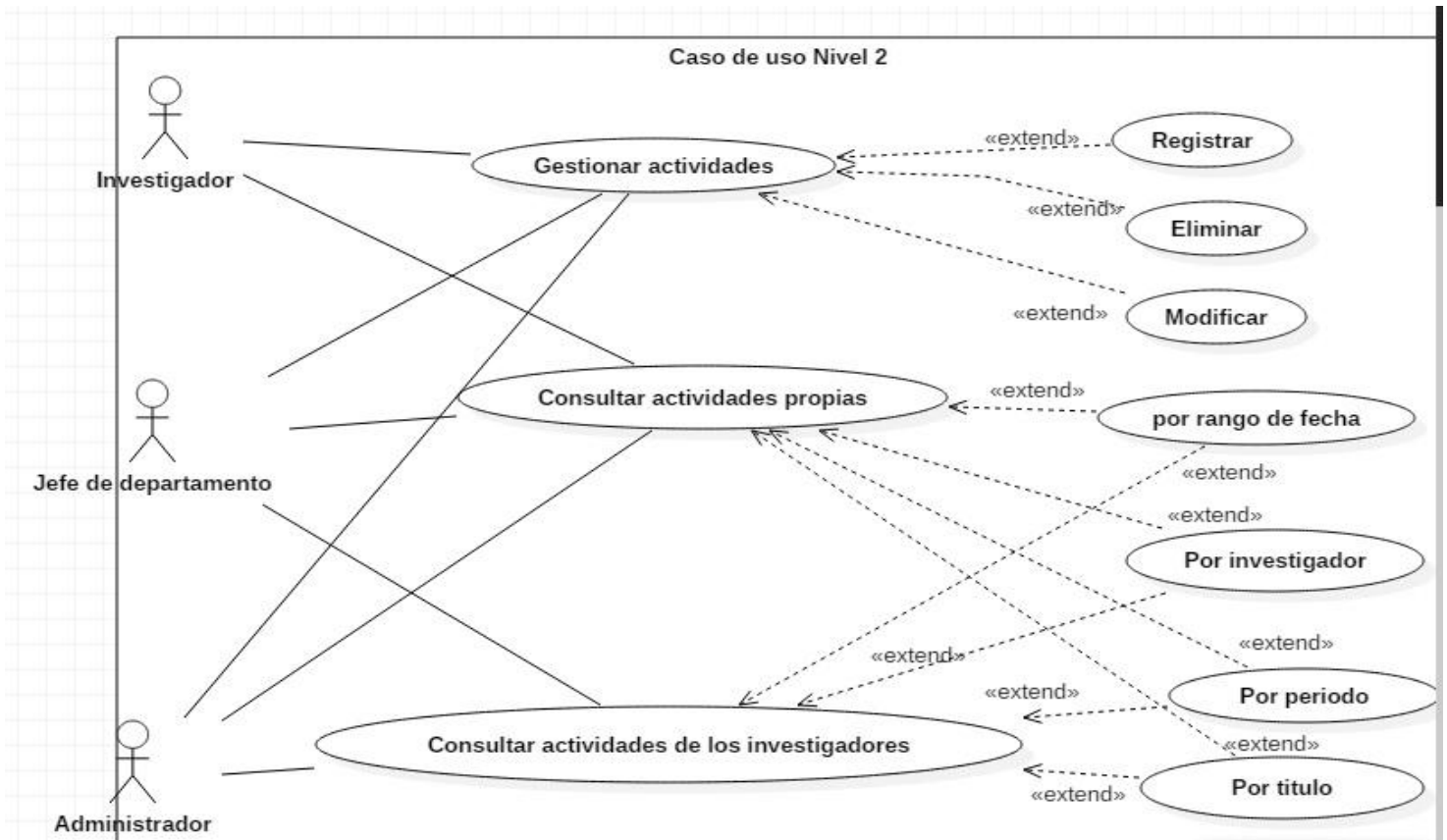


Figura 16. Actores investigador o profesor / jefe de departamento

En el anexo A, en las páginas N° 89 y N° 90, se pueden observar las tablas que detallan las funcionalidades de registrar, modificar, eliminar y consultar de la funcionalidad principal de *gestión de actividad*, además de especificar la precondición y el actor o actores que pueden ejecutar estas funcionalidades. Es válido resaltar que estas funcionalidades están disponibles para las actividades de todos los departamentos.

b) Modelo de objetos del dominio

Un modelo del dominio es una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés. Utilizando la notación UML, se representa con un conjunto de diagramas de clases en los que no se define ninguna operación (Larman, 2003).

Pueden mostrar:

- d) Objetos del dominio o clases conceptuales.
- e) Asociaciones entre las clases conceptuales.
- f) Atributos de las clases conceptuales.

El objetivo del modelado del dominio es comprender y describir las clases más importantes dentro del contexto de la aplicación.

A continuación se muestra el modelo de objetos del dominio que se genera en esta etapa en función al módulo de gestión de actividades. En este modelo se detalla la clase *usuarios*, la cual se clasificará en tres (3) roles, los cuales son investigador, jefe de departamento (coordinador) y administrador. Esta clase *usuario* se relaciona con la aplicación, específicamente la funcionalidad de *gestión de actividades*, la cual refleja la información planteada de los cinco (5) formularios de los diferentes departamentos. A su vez esta clase puede contener una gran cantidad de actividades y cada una de las actividades pertenecen solo a un formulario o departamento. Como se detalla en la figura 17, cada actividad contará con las funcionalidades de consultar, modificar, eliminar, registrar y generar reporte. Lo cual permitirá a los usuarios gestionar cada una de sus actividades.

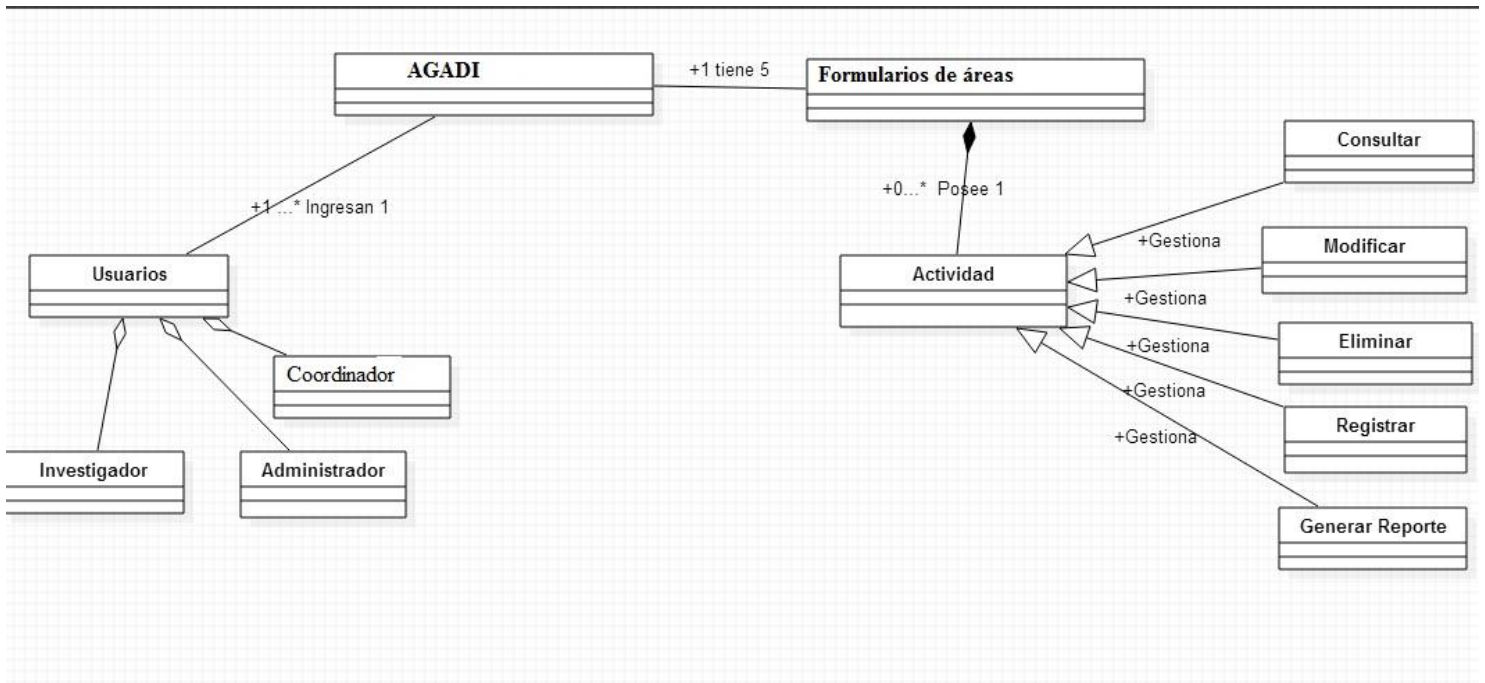


Figura 17. Modelo de objeto del dominio

c) Modelo físico de la base de datos

A continuación, en la figura 18, se puede observar el modelo físico de base de datos a emplear para solventar la problemática antes descrita.

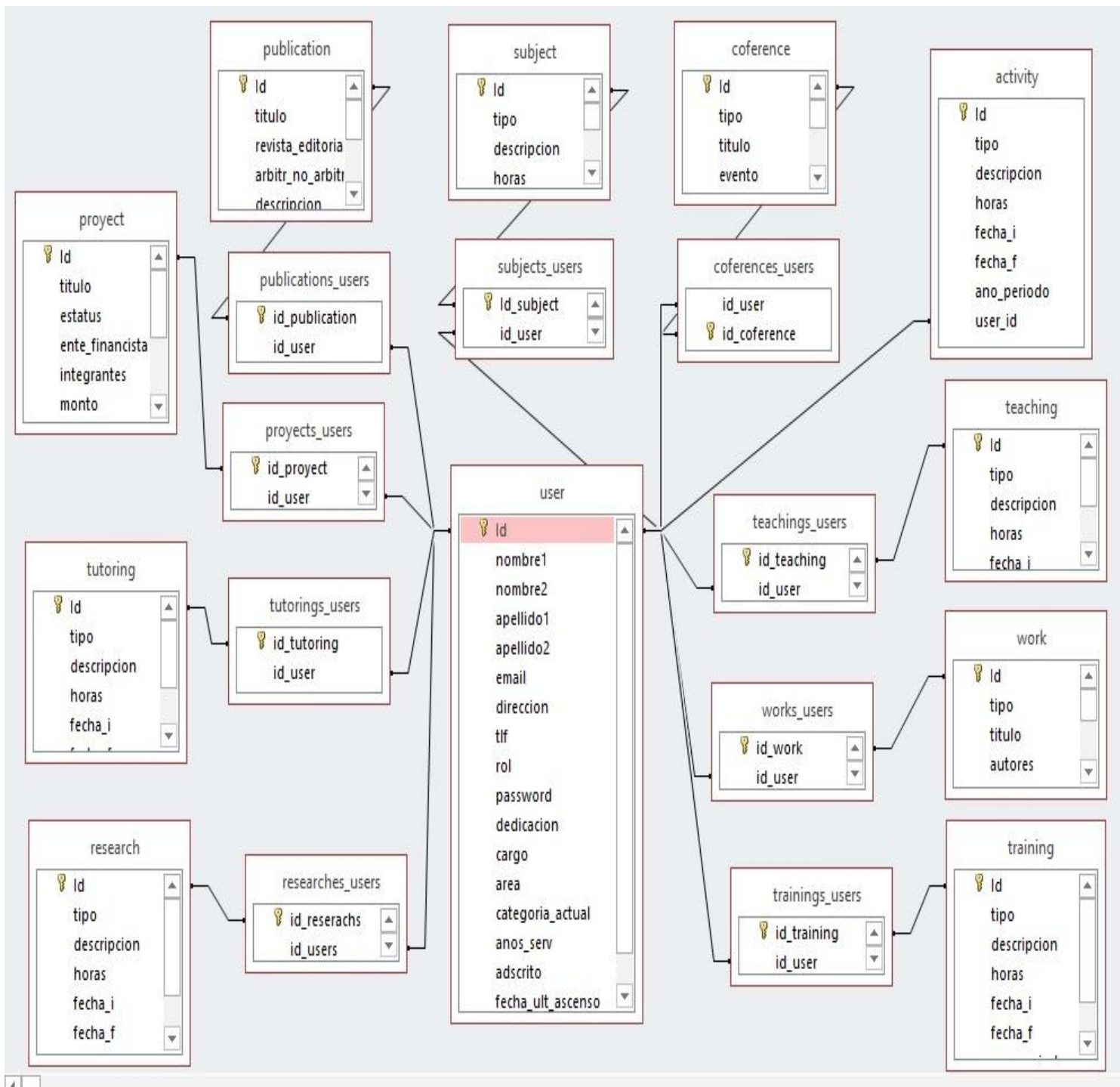


Figura 18. Modelo físico de la base de datos

En el diagrama anterior se muestra en el centro la tabla de *usuarios*, siendo esta la tabla principal, que contiene la información de los usuarios y esta a su vez está conectada con las

demás tablas, a través de la clave primaria *id*, que en las demás tablas es llamada *user_id* y es clave foránea en todas las demás tablas representan las actividades de los diferentes departamentos que realizan los investigadores.

Del departamento de investigación se tienen las tablas *proyect*, *publication* y *research*. La tabla *proyect* guardará toda la información referente a los proyectos realizados por el investigador, la tabla *publication*, tendrá la información de las publicaciones y la tabla *research* guardará la información referente a las demás actividades de investigación.

En el departamento de docencia encontramos las tablas *subject*, *tutoring*, *work* y por último la tabla *teaching*. La tabla *subject* contendrá la información de las asignaturas dictadas, la tabla *tutoring* tendrá las actividades de tutorías, la tabla *work* describirá la información en las cuales el docente-investigador ha servido de jurado en trabajos de grado, postgrado, entre otros, y por último, se tiene la tabla *teaching*, la cual registrará las actividades de los demás campos del departamento de docencia.

El departamento de extensión se dividió en dos tablas, una llamada *conference* en la cual se registrará la información de conferencias o ponencias en eventos nacionales o internacionales y la segunda llamada *extensión*, la cual contendrá las actividades restantes del departamento de extensión.

Por último las tablas de *training* y *activity*, la primera guardará las actividades de mejoramiento y capacitación, y la segunda, las actividades académico administrativas.

Se buscó crear varias tablas de un departamento en algunos casos, como por ejemplo en el departamento de investigación debido a que los campos como los de guardar proyectos serán campos que se consultaran más, además de requerir guardar otros campos como *monto del proyecto*, que no vienen al caso en los otros campos del departamento de investigación. Y al registrar una actividad que no sea de proyectos, quedaría ese campo de *monto de proyecto* en blanco como muchos otros campos.

Otro problema que se encontró al diseñar el diagrama de la base de datos es que en una actividad pueden participar varios investigadores, por tal motivo se crearon tablas intermedias llamadas *projects_users*, *publications_users*, *researches_users*, *subjects_users*, *tutorings_users*, *works_users*, *teachings_users*, *conferences_users*, *extensions_users* y por último *trainings_users*. Esto con el objetivo de que estas tablas intermedias guarden la relación de una actividad con el investigador, dando como resultado que las tablas de actividades no registren la misma actividad por cada investigador que participe en la misma actividad.

También se implementó una tabla de nombre *periods*, la cual estará conformada por dos campos que son el *año* o *período* y el *estatus* de ese período. El objeto de esta tabla es brindarle al administrador un control sobre la fecha de inicio de las actividades que los investigadores puedan registrar, debido a que un ciclo de trabajo abarca parte de dos años, es decir, esta intercalado en medio dos años, final de un año y principio del siguiente año.

d) Prototipo en papel

En la figura 19 se puede observar el prototipo en papel inicial de la interfaz diseñada para el requerimiento de Inicio de Sesión, en donde se detalla un banner superior con el logo de la aplicación AGADI, en el menú a la derecha se encuentra la opción para cerrar sesión, más abajo se encuentra el formulario planteado para que los usuarios se autentiquen en la aplicación.



Figura 19. Prototipo de papel del inicio de sesión

En la figura 20 se detalla el prototipo inicial de la vista diseñada para el inicio del portal donde se encuentran funcionalidades como buscar actividades por diferentes campos, gestionar actividades y gestionar usuarios (funcionalidad solo visible para usuarios administrador), en estas dos (2) funcionalidades anteriores, se contará con acciones como mostrar, editar, registrar y eliminar. Por último se tendrá una funcionalidad para generar los reportes en formato PDF y generar tablas, con información de vital importancia para el informe de gestión.



Figura 20. Prototipo en papel de entrada al portal de usuario

e) Guía de estilos

Bootstrap es un framework front-end desarrollado y liberado por Twitter, en el cual se involucran desarrolladores y diseñadores que manejan las tendencias web del momento. El tamaño de letra (font-size) por defecto en este framework en su versión N° 3 es 14px y el interlineado (line-height) es 1.428. El tipo de fuente utilizado es “sans-serif”. Fue dejado por defecto ya que se busca con este trabajo de grado, una aplicación con un diseño agradable, sencillo y sea del gusto de todos los investigadores del Instituto.

El menú utilizado fue el “navbar-inverse” el cual se muestra en la figura 22.



Figura 21. Menú navbar-inverse de bootstrap

La paleta de colores predefinida para el menú navbar-inverse de bootstrap se muestra en la siguiente figura.



Figura 22. Paleta de colores menú navbar-inverse de bootstrap

Los colores utilizados para los botones de la aplicación, fueron los colores de bootstrap por defecto para este tipo de componente, en la siguiente figura se muestra el color utilizado.



Figura 23. Vista de botones usados por bootstrap.

A continuación se muestran el diseño de los botones con la paleta de la figura 24.



Figura 24. Diseño de botones por defecto de Bootstrap

Otro de los componentes de bootstrap utilizado es el panel “panel-primary” el cual contiene una cabecera con el color #367884, el mismo utilizado por botón “button-primary” mostrado en la figura 24 anterior.

3.3 Etapa III: Prototipaje

La aplicación AGADI fue desarrollada utilizando, aplicaciones y tecnologías libres, dentro de las cuales podemos mencionar:

- Postgresql versión 9.4 como Sistema Manejador de Bases de Datos.
- Ruby on Rails en su versión 2.2.4.
- Javascript versión estable, como lenguaje del lado del cliente que provee dinamismo.
- JQuery versión 2.2.2, como framework de Javascript para la programación del lado del cliente.
- HTML5, como lenguaje del lado del cliente para darle estructura a la aplicación.
- CSS3, como lenguaje del lado del cliente para darle estilo a la aplicación web.
- Bootstrap versión 3.3.4 como framework para crear interfaces y diseños web responsive basados en HTML5 y CSS3, entre otros.

AGADI se desarrolló sobre la plataforma del Sistema Operativo Linux Mint en su versión N° 17, el cual es un sistema operativo basado en UNIX, en el que se encuentra instalado el framework para el desarrollo de aplicaciones Web del lado del servidor Ruby on Rails y el Sistema Manejador de Bases de Datos PostgreSQL. Adicionalmente para definir la apariencia de las interfaces de la aplicación se utilizó el framework CSS Bootstrap.

En esta etapa, se obtuvo algunos artefactos y se realizaron evaluaciones con el fin que el usuario no tuviera que esperar hasta el final del desarrollo de la aplicación para comenzar a usarlo y comunicar sus aportes como miembro del equipo de desarrollo de software.

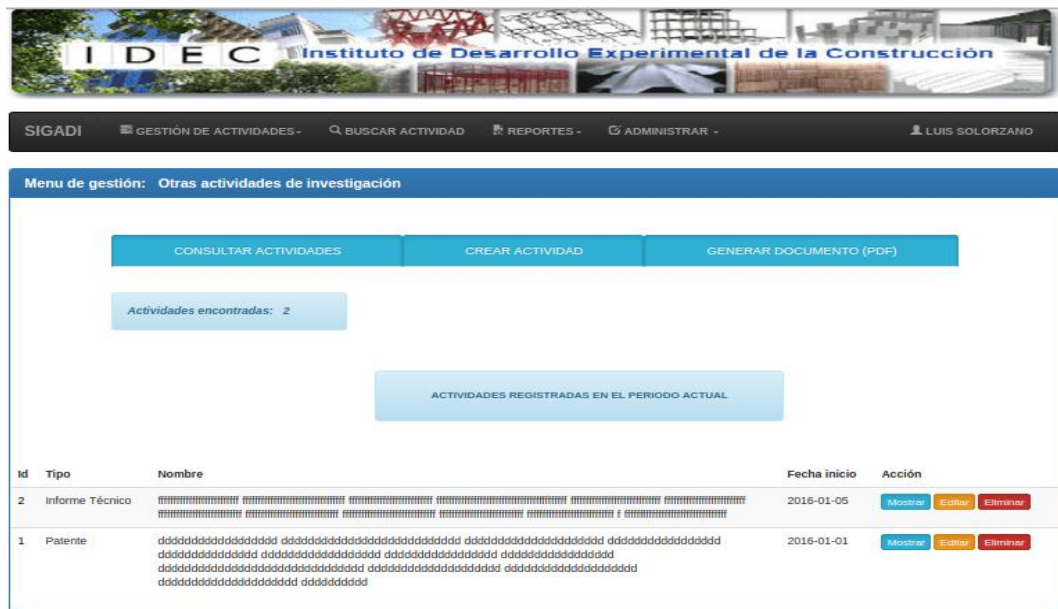
Algunos de ellos son:

a) Prototipo rápido

Se desarrolla un prototipo ejecutable a partir de los artefactos generados en las etapas anteriores, con la finalidad que el usuario pueda interactuar con los módulos que conforman la aplicación web, sin la necesidad que todos hayan sido desarrollados completamente. En esta etapa se desarrolla un prototipo ejecutable para las funcionalidades principales como lo son: Gestionar de actividades, buscar actividades, generar reportes y gestionar usuarios.

A continuación se mostrarán algunos prototipos para estas funcionalidades.

En la figura 25 mostrada a continuación, pertenece a la funcionalidad de gestión de actividades (específicamente pertenece a la gestión de otras actividades del departamento de



© 2015 IDEC, Instituto de Desarrollo Experimental de la Construcción.

investigación).

Figura 25. Prototipo de la funcionalidad de gestión de actividades.

En esta funcionalidad los investigadores podrán revisar todas las actividades del departamento y del tipo de actividad que fue previamente seleccionada. AGADI permite registrar

una nueva actividad y generar un reporte con la lista de actividades. Además pondrán contar con una serie de acciones sobre las actividades ya registradas como lo son: Mostrar en detalle, modificar algún campo de la actividad y eliminar actividad.

En la figura 26 se presenta la funcionalidad de buscar actividad.

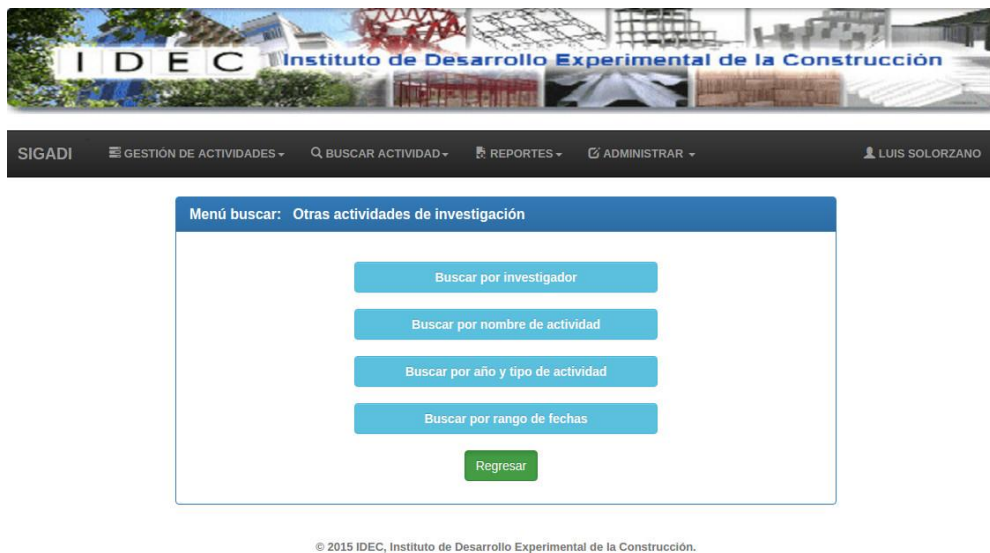


Figura 26. Prototipo de la funcionalidad de buscar actividad

En dicha funcionalidad el investigador contará con cuatro (4) tipos de filtro que permitirán búsquedas más específicas. Estos filtros son: *Buscar por investigador*, *buscar por nombre de actividad*, *buscar por año y tipo de actividad*, y por último, *buscar por rango de fecha*. Al igual que la funcionalidad anterior el investigador elegirá previamente el departamento y tipo de actividad.

A continuación, en la figura 27, se observa el prototipo de la tercera funcionalidad principal, que es generar reporte.



© 2015 IDEC, Instituto de Desarrollo Experimental de la Construcción.

Figura 27. Prototipo de la funcionalidad generar reportes

En la figura anterior del prototipo se observa un menú con varias opciones de tablas como por ejemplo: Actividades realizadas a través del departamento de extensión, tabla de jefes de departamento del I.D.E.C., tabla resumen totalizador de investigaciones, entre otras.

La figura 28 muestra la cuarta funcionalidad, que es la gestión de usuarios.

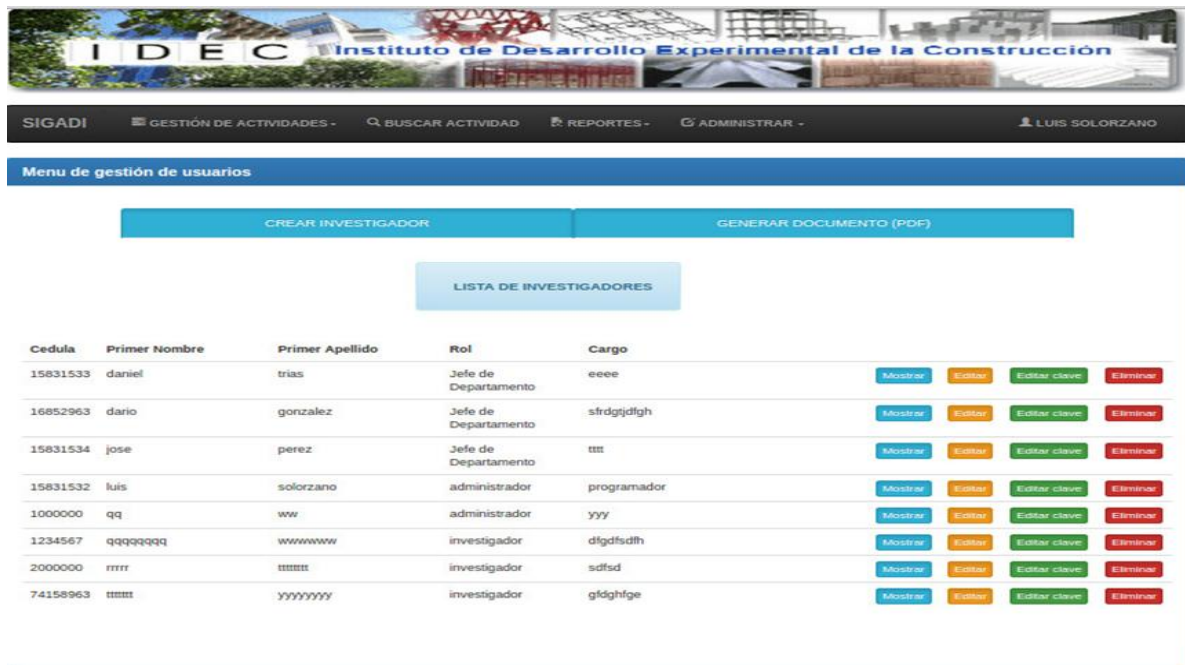


Figura 28. Prototipo de la funcionalidad de gestión de usuarios.

La funcionalidad de gestión de usuarios permitirá listar todos los investigadores registrados en la aplicación, registrar un nuevo usuario, además de aplicar sobre cada docente de la lista, varias acciones, como modificar información, mostrar en detalle la información, cambiar clave y eliminar.

b) Evaluación heurística

Las pruebas heurísticas se realizan con el fin de encontrar problemas de usabilidad en la interfaz de usuario de un sistema (Nielsen, 1995). Para ello, se utilizan las heurísticas creadas por Jakob Nielsen, las cuales se describen a continuación. Las heurísticas que se tomaron en cuenta para evaluación son las siguientes:

H1. Visibilidad de sistema: el sistema debe informar a los usuarios del estado del sistema, dando una retroalimentación apropiada en un tiempo razonable.

H2. Utilizar el lenguaje de los usuarios: el sistema debe utilizar el lenguaje de los usuarios, con palabras o frases que le sean conocidas, en lugar de los términos que se utilizan en el sistema, para que no resulte difícil utilizar el sistema.

H3. Control y libertad para el usuario: en caso que los usuarios elijan una opción del sistema por error, éste debe contar con las opciones de deshacer y rehacer para proveer al usuario de una salida fácil sin tener que utilizar un diálogo extendido.

H4. Consistencia y estándares: el usuario debe seguir las normas y convenciones de la plataforma sobre la que está implementando el sistema, para que no se tenga que preguntar el significado de las palabras, situaciones o acciones del sistema.

H5. Prevención de errores: es más importante prevenir la aparición de errores que de generar buenos mensajes de error.

H6. Minimizar la carga de la memoria del usuario: en la interfaz se deben mantener los objetos, acciones y opciones visibles para que el usuario no pierda el tiempo recordando la información de una parte del diálogo a otra.

H7. Flexibilidad y eficiencia de uso: las instrucciones para el uso del sistema tienen que ser visibles o fácilmente accesibles cada vez que se necesiten, para que al usuario inexperto se le facilite aprender a utilizar el sistema. Es importante que el sistema permita personalizar acciones frecuentes, para acelerar el uso del sistema.

H8. Los diálogos estéticos y diseño minimalista: la interfaz no debe contener información que no sea relevante o se utilice raramente, pues cada unidad adicional de información en un diálogo compete con las unidades relevantes de la información y disminuye su visibilidad relativa.

H9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores: los mensajes de error deben expresarse en un lenguaje claro y deben indicar exactamente el problema.

H10. Ayuda y documentación: en caso que el sistema necesite disponer de ayuda y documentación, ésta tiene que ser fácil de encontrar, centrada en las tareas del usuario, tener información de las etapas a realizar y que no sea demasiado extensa.

La escala de evaluación va de 0 a 4 con los siguientes valores: (0) No es un problema de Usabilidad, (1) Problema sin importancia, (2) Problema de poca importancia, (3) Problema grave y (4) Catástrofe es obligatorio arreglarlo.

En la Tabla 1 se presenta el ejemplo de la tabla usada para aplicar la prueba heurística y algunos de los resultados obtenidos al aplicar dicha prueba.

Tabla 1. Resultados de la evaluación Heurística

PROBLEMA	HEURISTICA	VALORACION	SOLUCION
Se requiere iniciar explicando el proyecto en sus iniciales y a quien va dirigido	H4	5	Colocando al inicio ¿Qué es AGADI?
Posibilidad de graficar datos históricos por cada investigador	H7	4	Generar gráficos de torta y de barra
Si uno olvida la contraseña no aparece el link para recuperarla	H9	4	Incluir olvido de contraseña
No indica el período que se va a registrar. Es decir si es para la gestión de cada año. Debe indicarlo	H1	2	Cargar al período actual las actividades de años anteriores.
Al registrar un proyecto, el campo Co-investigador no debe ser obligatorio	H5	5	El campo Co-investigador no debe ser obligatorio o cuando se seleccione proyecto individual desaparece la opción de co-investigador
Mejorar el diseño gráfico. Hacerlo más sobrio	H8	4	Hacerlo en concertación con el cliente
Al registrar una tutoría no quedo registrada	H9	4	Revisar y acomodar la programación
Al registrar una actividad y ocurrir un error en algún campo del formulario, hay que volver a seleccionar los campos desplegables	H9	3	Que los campos desplegables guarden la opción ya elegida
Algunos textos presentan detalles de sintaxis y ortografía.	H2	4	Revisar los textos de la aplicación y corregir

Los problemas encontrados durante la aplicación de esta evaluación heurística se solventaron oportunamente, permitiendo al grupo de desarrollo evaluar las sugerencias del especialista y realizar los ajustes necesarios que permiten obtener un producto de software que se ajuste a las necesidades del usuario, obteniendo un software refinado y con altos estándares de calidad.

3.4 Etapa IV: Entrega

En esta fase se aplican diferentes pruebas a la aplicación, con el fin de certificar que el sistema desarrollado cumple satisfactoriamente con los requerimientos funcionales y no funcionales, además verifica que posee los niveles de usabilidad presentes en un software de calidad. Adicionalmente, las pruebas de aceptación verifican el grado de satisfacción de los usuarios finales en relación a los requerimientos funcionales de la aplicación.

Es importante mencionar que luego de implementar las funcionalidades de la aplicación AGADI y realizar las pruebas correspondientes, se entregó el software y fue puesto en producción con todas las funcionalidades explicadas antes. Con la finalidad de realizar pruebas piloto se les pidió a un grupo de investigadores pertenecientes al I.D.E.C. que registraran sus actividades y evaluar el comportamiento de la aplicación.

A continuación se presenta el artefacto generado en esta etapa de la metodología AgilUs para evaluar la aplicación.

a) Pruebas de aceptación

Para llevar a cabo la verificación del software, se realizó una prueba de aceptación de todas las funcionalidades que ofrece la aplicación en su versión de producción, con el objetivo de detectar posibles fallos e inconsistencias. Se aplicó esta prueba a través de un cuestionario.

En la figura 29 se observa el cuestionario de usabilidad aplicado a los investigadores del I.D.E.C. el cual evalúa la aceptación de la aplicación por el grupo de investigadores del Instituto.

Cuestionario de Usabilidad de AGADIE

El presente cuestionario tiene como finalidad conocer su opinión con respecto a la usabilidad de **AGADIE**. La cual se refiere a la efectividad, eficiencia y satisfacción con la que la aplicación pueda ser comprendida, utilizada y ser atractiva al usuario.

Se agradece seleccionar la puntuación que corresponda a su opinión de acuerdo a la siguiente escala:

Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
5	4	3	2	1

N°	Item	5	4	3	2	1
1	Los títulos, menús y textos de la aplicación son legibles					
2	Los colores utilizados contribuyen a la percepción adecuada de la aplicación.					
3	Los descriptores de la aplicación son claros y precisos					
4	Se ofrece una organización que facilita la navegación					
5	El diseño de la aplicación favorece su uso					
6	La aplicación se puede ejecutar de manera intuitiva					
7	Es una aplicación fácil de utilizar					
8	Los elementos de ayuda disponibles en la aplicación son útiles					

Figura 29. Prueba de aceptación para evaluar la usabilidad de AGADI

El grupo de muestra al que se le aplicó la encuesta se encuentra formado por 5 miembros del I.D.E.C. con diferentes roles dentro de la aplicación, los cuales llevaron a cabo esta encuesta de manera individual, con una escala de evaluación sobre su apreciación con valores entre 1 y 5, siendo 1 el nivel de apreciación más bajo y 5 el nivel de apreciación más alto.

A continuación, en la figura 30 se pueden observar los resultados de la aplicación de esta prueba con las cuatro (4) primeras preguntas a cinco (5) de los investigadores.

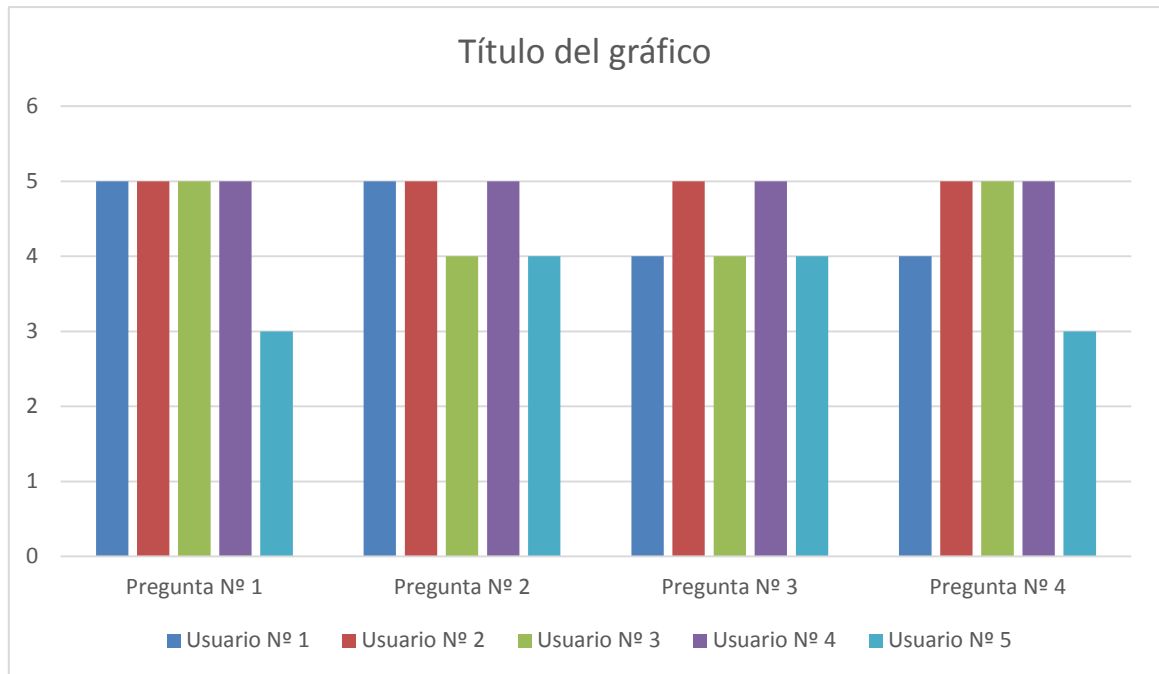


Figura 30. Prueba de aceptación con las primeras cuatro (4) preguntas.

A continuación, en la figura 30 se pueden observar los resultados de la aplicación de esta prueba con las cuatro (4) últimas preguntas a los cinco (5) mismos investigadores.

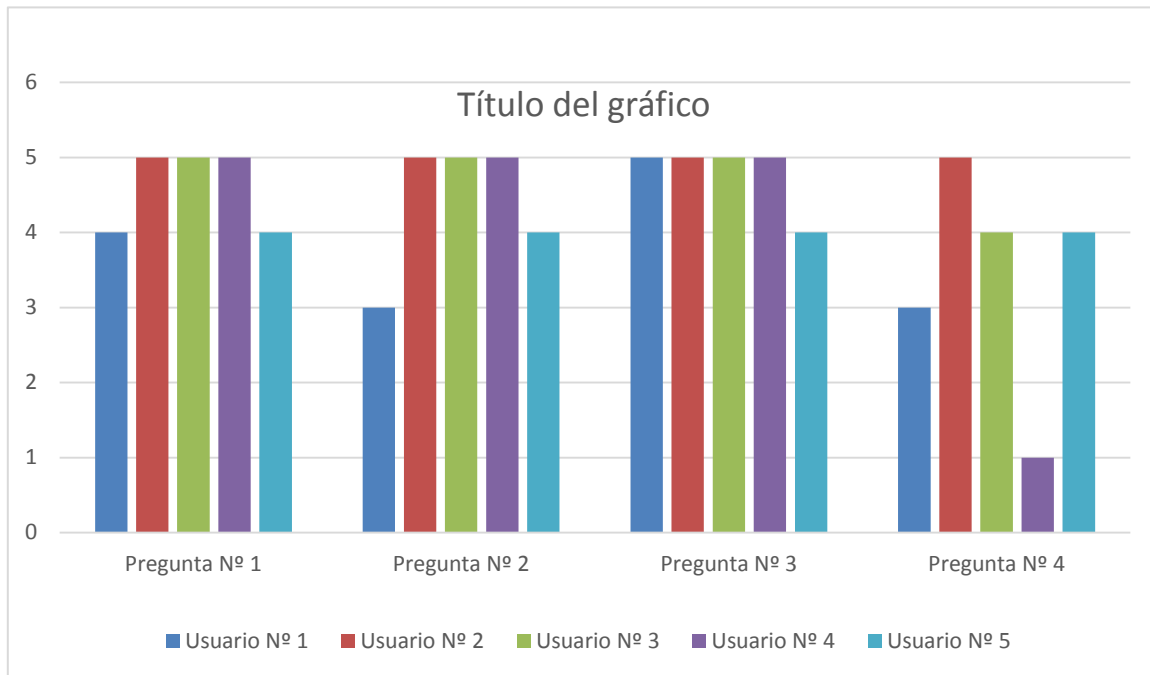


Figura 31. Prueba de aceptación con las cuatro (4) últimas preguntas

Como resultado de la aplicación de esta prueba de aceptación es posible concluir que en general la aplicación AGADI posee un alto nivel de aceptación por parte del grupo de investigadores del I.D.E.C., con el 88% de aceptación, cumpliendo con las necesidades y requerimientos del usuario, a través de una aplicación usable y con altos estándares de calidad. Por medio de la aplicación de la metodología de desarrollo AgilUs fue posible llevar a cabo exitosamente el proceso de desarrollo de la aplicación AGADI, aplicando durante cada una de las etapas, diferentes técnicas y elaborando artefactos para la consecución de la usabilidad.

En este capítulo se presentó la etapa de requisitos, de análisis, prototipaje y entrega, las cuales conforman la metodología AgilUs, y que permitieron guiar el desarrollo de la aplicación y adaptarla a los cambios. Para culminar, se procede al Capítulo IV referente a los resultados que se obtuvieron a lo largo del desarrollo de AGADI.

Capítulo IV. Resultados

En este capítulo se presentarán los resultados obtenidos, luego de utilizar la metodología AgilUs para el desarrollo de la aplicación AGADI y cada una de sus funcionalidades.

AGADI cuenta con cuatro (4) funcionalidades principales, las cuales son gestión de actividades, búsqueda de actividades, generación de reportes y gestión de usuarios. Además la aplicación cuenta con tres (3) niveles o rol de usuarios que son investigador, jefe de departamento y administrador.

El rol investigador contará en su portal con acceso a las funcionalidades de gestión de actividades y búsqueda de actividades. Adicionalmente cada usuario con rol de investigador se le permitirá modificar la información de su perfil y exportar en formato PDF la lista de actividades o la información detallada de alguna actividad.

El rol jefe de departamento se le permitirá en su portal el acceso a las funcionalidades de gestión de actividades, búsqueda de actividades, por último, dispondrá de una tercera funcionalidad que es generar reportes, con la cual los jefes de departamento podrán exportar la información de los investigadores a tablas referentes a distintos departamentos y generar gráficos con estadísticas de las actividades de los investigadores.

A continuación se describirá la vista de inicio de sesión y las cuatro (4) funcionalidades principales de AGADI.

5.1 Funcionalidad de inicio de sesión de AGADI

La interfaz de inicio de sesión es la primera vista de la aplicación AGADI, esta vista está compuesta en la parte superior por un banner, el cual es una imagen que contiene las iniciales del Instituto, además incluye las iniciales de la aplicación y el significado de AGADI. Seguido de este se encuentra la barra principal del menú, esta barra, contiene a la derecha las opciones de cuenta del usuario registrado previamente y permite cerrar la sesión del usuario.

En la parte central de esta vista de inicio se encuentra un recuadro el cual contiene el formulario de autenticación para crear una sesión en la aplicación y poder entrar al portal del rol que le fue asignado al ser registrado. El formulario cuenta con dos campos de texto, que son la cédula del usuario y contraseña, en esta funcionalidad la aplicación comparará los datos introducidos con los datos ya guardados previamente. Si los datos son correctos la aplicación presentará la interfaz de bienvenida al portal o si no son correctos los datos, se presentará un mensaje de alerta momentáneo que indicara que dichos datos son incorrectos.

Para finalizar, en la parte inferior de la vista se encuentra el pie de página el cual indica el nombre completo del Instituto, el año en que fue desarrollada la aplicación AGADI, la dirección y los teléfonos de contacto del I.D.E.C., por último en la esquina derecha del pie de página se encuentra el logo de las licencias Creative Commons asociadas. Las licencias asociadas son atribución – no comercial –compartir igual.

En la siguiente figura 32 se observa la primera vista de AGADI de inicio de sesión.

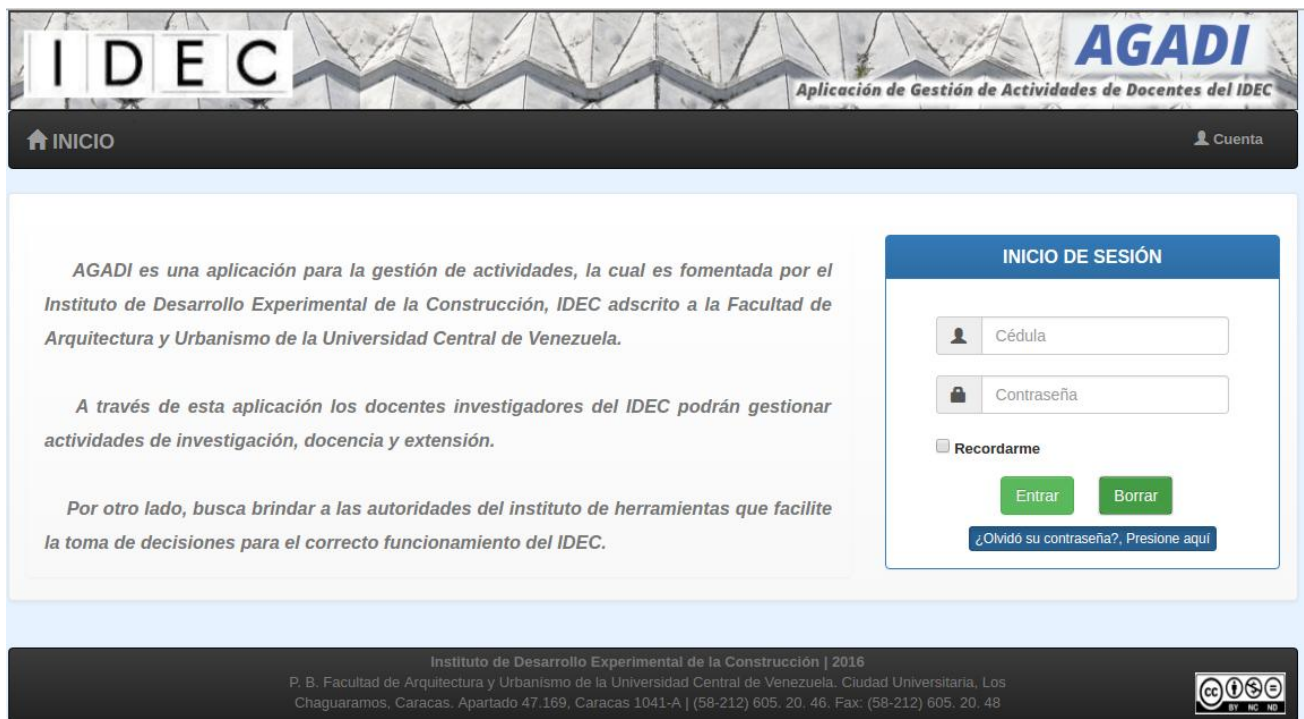


Figura 32. Interfaz inicio de sesión.

Cabe mencionar que todas las vistas de AGADI estarán conformadas por el banner, menú principal y en la parte inferior, el pie de página.

5.2 Funcionalidades principales de usuario

Como ya se mencionó, AGADI cuenta con cuatro (4) principales funcionalidades, estas funcionalidades aparecerán en el menú principal dependiendo del rol de usuario autenticado.

En la figura 33 se muestra el portal de bienvenida del usuario administrador el cual dispondrá en el menú principal, de todas las funcionalidades.

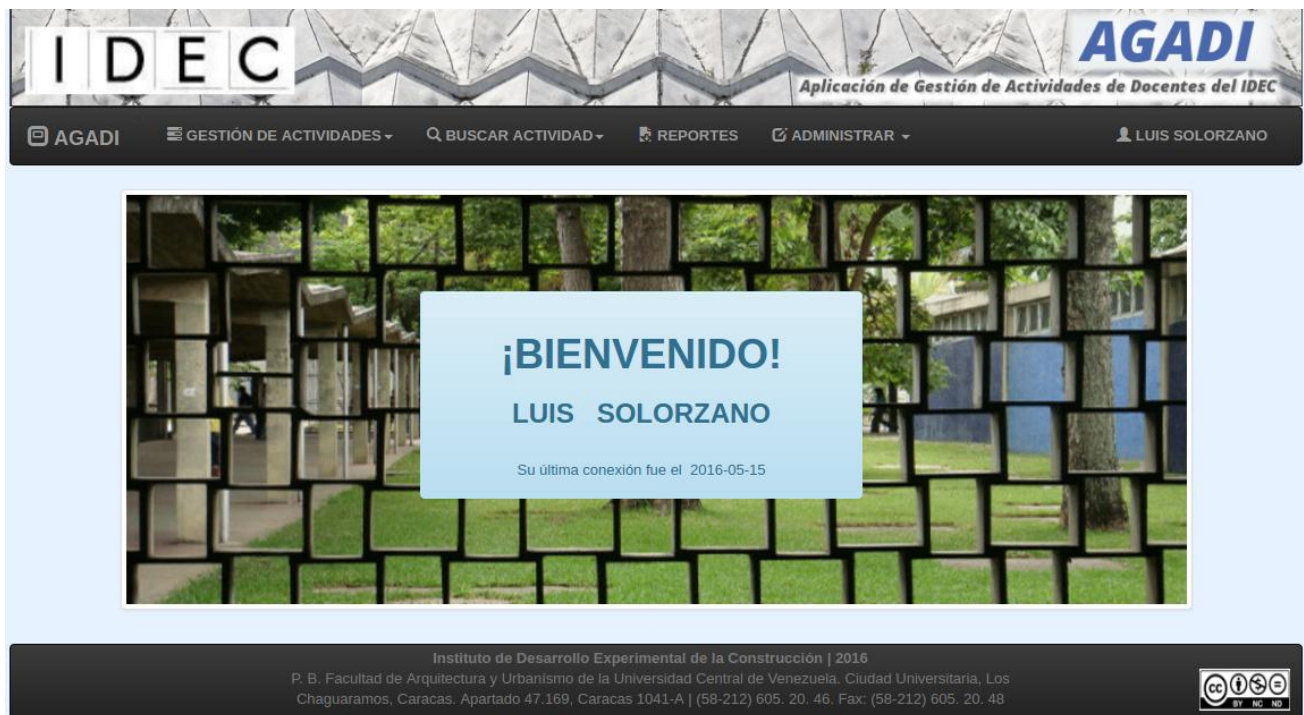


Figura 33. Portal de bienvenida del usuario administrador

Seguido de la barra de menú se encuentra un panel de bienvenida con el nombre del investigador que se autentique en el momento, con la fecha de su última conexión.

A continuación se describirán las funcionalidades principales de AGADI.

5.2.1 Funcionalidad de gestión de actividades

El usuario en esta funcionalidad podrá seleccionar el departamento de la actividad a gestionar y luego podrá especificar el tipo de la actividad, dentro del departamento previamente seleccionado. AGADI cuenta con once (11) modelos de actividades, los cuales se especifican a continuación:

- Departamento de investigación
 - Proyectos
 - Publicaciones
 - Otras actividades de investigación

- Departamento de docencia
 - Dictado de asignaturas
 - Tutorías
 - Jurado (jurado de trabajos de grado, postgrado y ascensos, arbitraje o evaluación de cursos).
 - Otras actividades de docencia

- Departamento de extensión
 - Conferencias o ponencias en eventos
 - Otras actividades de extensión

- Actividades de mejoramiento y capacitación
- Actividades académico administrativas

A continuación se presenta la figura 34, la cual contiene el menú de gestión de proyectos de investigación, los demás módulos de los otros tipos de actividades son similares.

Id	Titulo	Estatus	Fecha inicio	Acción
9	p163	Culminado	2016-02-19	Mostrar Editar Eliminar
7	p124	En desarrollo	2016-02-19	Mostrar Editar Eliminar
6	p123	Culminado	2016-02-11	Mostrar Editar Eliminar
3	p3	En desarrollo	2016-02-06	Mostrar Editar Eliminar
1	p1	En Espera de Aprobación	2016-02-04	Mostrar Editar Eliminar
4	p57	En Espera de Aprobación	2016-02-04	Mostrar Editar Eliminar

Figura 34. Funcionalidad gestión de actividades

En esta funcionalidad, el investigador podrá consultar todas las actividades previamente registradas por él, pertenecientes al período activo. También podrá registrar nuevas actividades y exportar la lista de actividades a un PDF, además cada actividad en la parte derecha contará con una serie de acciones, las cuales permitirán mostrar en detalle la actividad seleccionada, editar o eliminar.

En la figura 35 se observa la acción de crear actividad, para el ejemplo de actividades de proyectos de investigación. Esta acción cuenta con un formulario el cual el investigador debe completar para describir la actividad que desea registrar. Una vez registrada la actividad, esta aparecerá en la lista de actividades mostrada en la figura 34.

Los campos con (*) son obligatorios.

Cargar actividad de años anteriores

Título:

 (*)

Estatus: (*)

Ente financista: (*)

Monto financiado: Bs 1000.00

Fecha inicio: (*)

Fecha fin: (*)

Horas dedicadas: **Periodo:** (*)

Seleccione si es en grupo o individual: (*)

Investigador responsable: (*)

Integrantes
 Integrante 1, integrante 2, integrante 3 ...

Figura 35. Registro de actividad

En la siguiente figura 36 tenemos un ejemplo de la funcionalidad generar PDF, esta funcionalidad permite volcar todas las actividades resultado de la consulta anterior a un archivo en formato PDF, desde el cual se podrá imprimir o guardar en el destino de su preferencia.

ACTIVIDADES DEL ÁREA DE INVESTIGACIÓN

Lista de proyectos del área de investigación
Fecha de emisión: 15/05/2016

LISTA DE ACTIVIDADES:

ID	TITULO	ESTATUS	FECHA INICIO
9	p163	Culminado	2016-02-19
7	p124	En desarrollo	2016-02-19
6	p123	Culminado	2016-02-11
3	p3	En desarrollo	2016-02-06
1	p1	En Espera de Aprobación	2016-02-04
4	p57	En Espera de Aprobación	2016-02-04
8	p145	En Espera de Aprobación	2016-02-03
2	f f f f f	En Espera de Aprobación	2016-02-01

Figura 36. PDF generado de la funcionalidad de gestión de actividades

5.2.2 Funcionalidad de búsqueda de actividades

Esta funcionalidad permitirá al investigador hacer consultas históricas una vez seleccionado el departamento de la actividad. El investigador contará con cuatro (4) filtros de búsquedas, los cuales son: Buscar por investigador, por nombre, por período y rango de fecha de la actividad.

Es importante señalar que el resultado de las consultas serán diferentes dependiendo del rol del investigador que aplique los filtros, por ejemplo, los usuarios con rol investigador, tendrán como resultados de búsquedas actividades en las que él es creador o este asociado, en cambio si

el usuario es administrador o jefe de departamento el resultado de la búsqueda será actividades de toda la población del Instituto según el filtro aplicado.

En la figura 37 se observa la funcionalidad de buscar actividades, en la que el investigador podrá seleccionar cuatro (4) tipos de filtro para realizar la consulta.

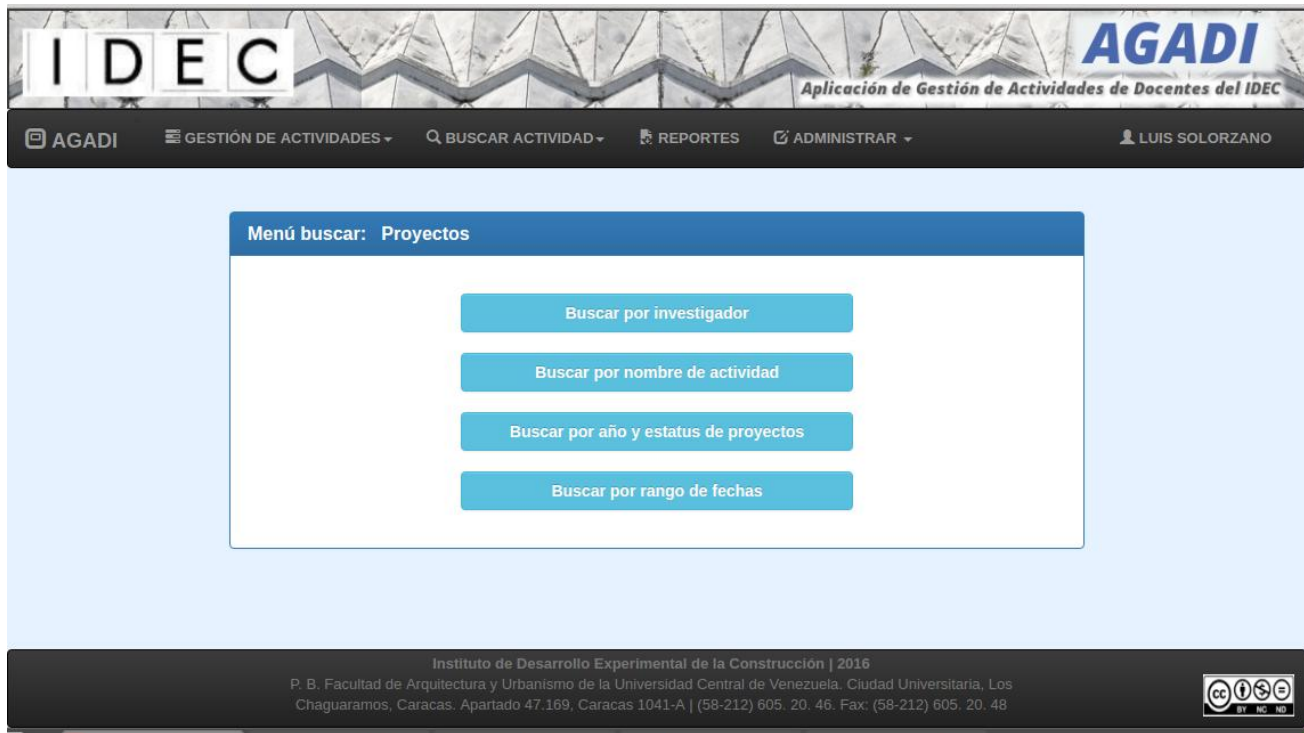


Figura 37. Funcionalidad buscar actividad

En el **filtro por investigador** el usuario podrá seleccionar el investigador el cual es el creador o está asociado a las actividades encontradas. Si el usuario que realiza la búsqueda es jefe de departamento o es administrador, este podrá seleccionar cualquier investigador registrado en la aplicación, en cambio si el usuario es de rol investigador, en la selección solo aparecerá su nombre, por lo tanto solo podrá consultar sus actividades.

En la siguiente figura 38 se muestra el filtro de búsqueda por investigador de una actividad de proyectos del departamento de investigación.

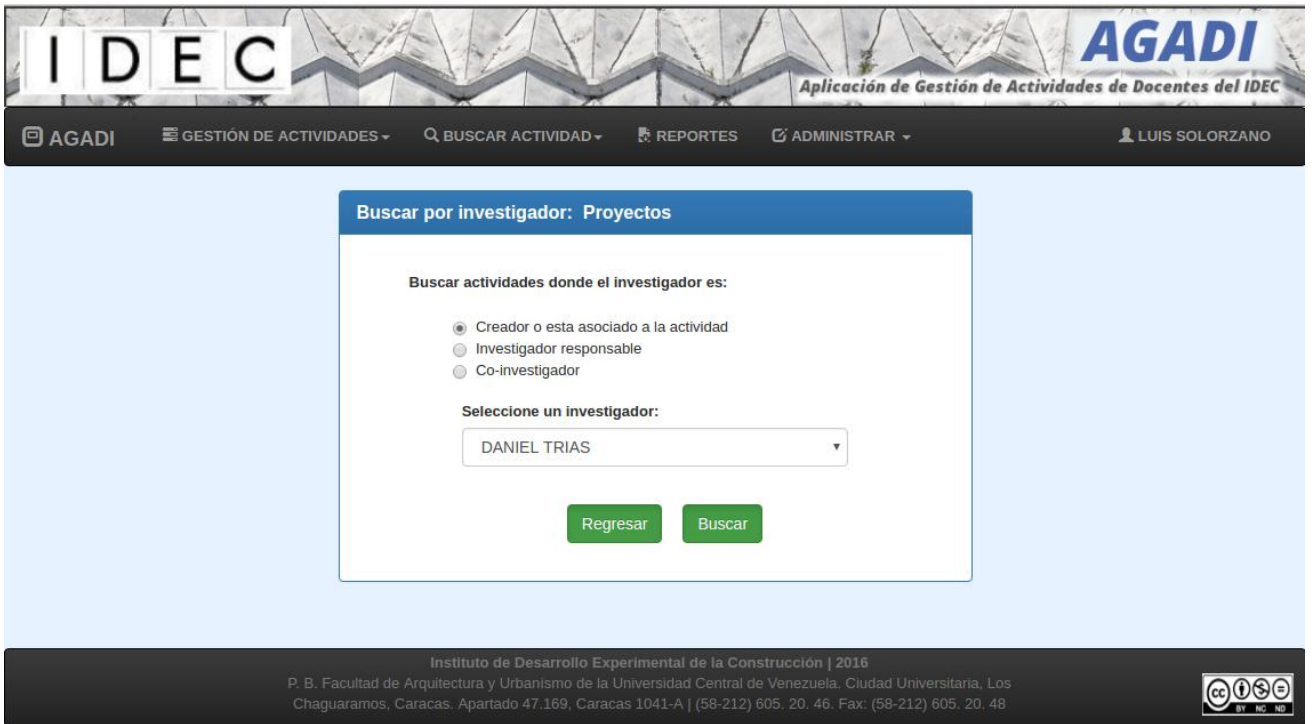


Figura 38. Filtro de búsqueda por investigador

El **filtro por nombre de actividad** se caracteriza por un campo de texto donde el investigador colocará alguna palabra clave del nombre de la actividad a buscar, la aplicación buscará el nombre colocado entre todas las actividades del tipo de actividad seleccionado y las actividades creadas de cualquier investigador. El resultado de la consulta puede resultar de una o más actividades, dependiendo de la palabra clave introducida, en la vista de presentación de resultados, brindará la opción de exportar a PDF las actividades encontradas. Otra característica de esta vista es que los administradores y jefes de departamento podrán buscar actividades entre todos los investigadores, ya que el usuario con rol investigador, solo se le presentará resultados de actividades en las cuales es creador o está asociado a dicha actividad.

A continuación se presentará la figura 39, en la que se muestra el filtro de búsqueda por nombre de actividad de proyectos del departamento de investigación.

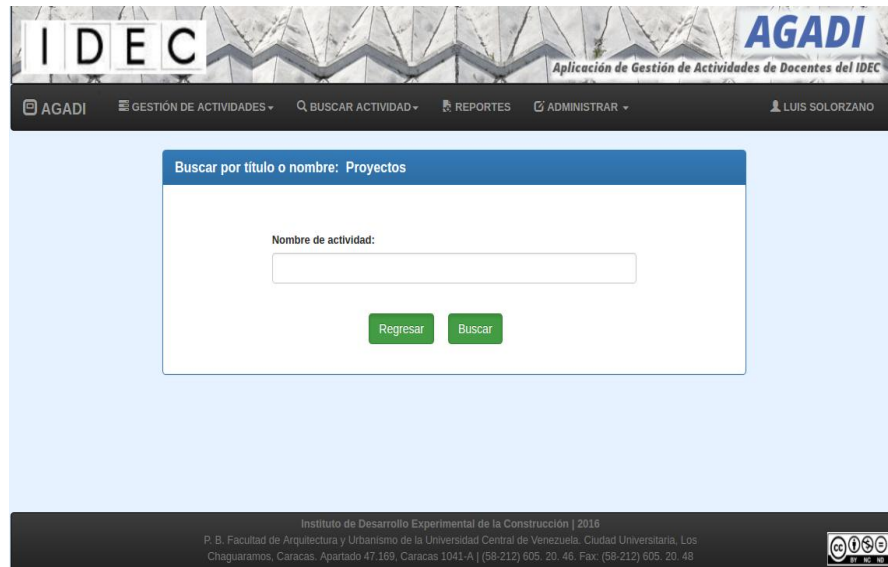
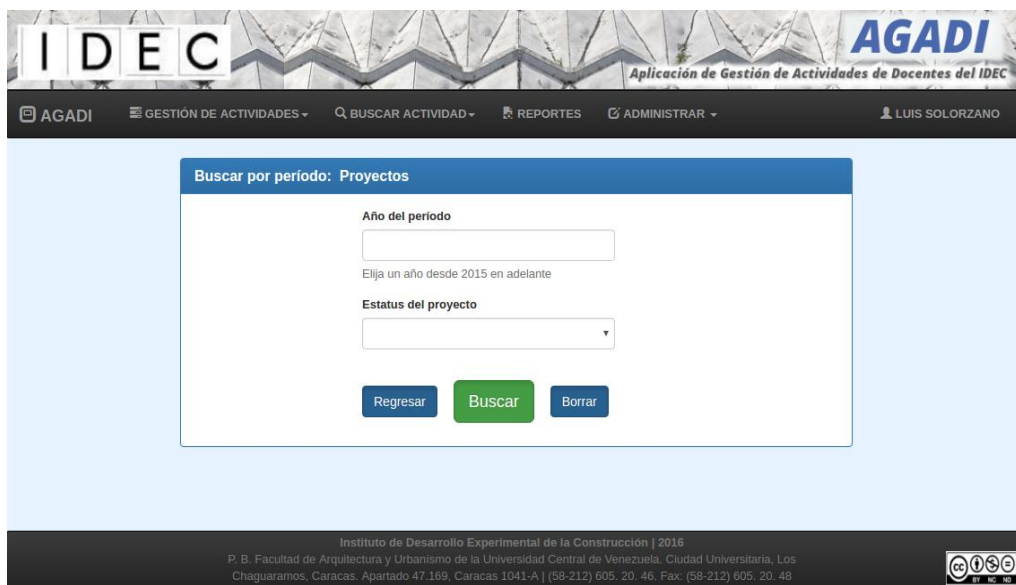


Figura 39. Filtro de búsqueda por nombre de actividad

El **filtro por período** tiene como objetivo que el investigador pueda consultar las actividades por períodos, es decir los usuarios con rol de jefe de departamento o administrador podrán visualizar todas las actividades de todos los investigadores del Instituto del período que seleccionó, en cambio los usuarios de rol investigador como en los filtros anteriores solo podrán visualizar las actividades del período que seleccionó en donde él sea el creador o este asociado a dichas actividades.

En la figura 40 se muestra la funcionalidad de búsqueda de actividades aplicando el filtro

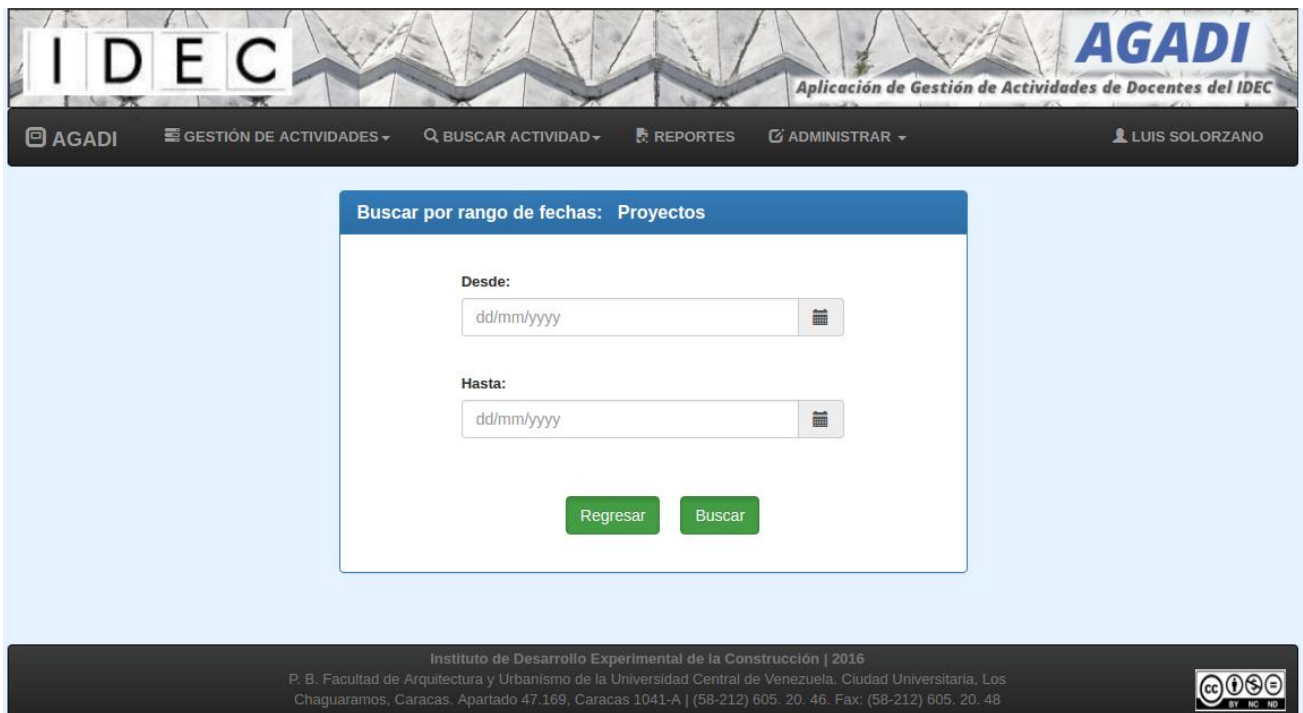


por períodos.

Figura 40. Filtro de búsqueda por período

El **filtro por rango de fecha** se basa en permitir a los usuarios consultar las actividades especificando un rango de fecha, la cual tendrá como resultado las actividades con fecha de creación que estén dentro del rango de fecha. Como en los otros filtros los usuarios con rol investigador no podrán visualizar las actividades de otros, la consulta solo tendrá como resultado actividades donde ellos son los creadores o estén asociados a dicha actividad.

En la figura 41 se presenta la funcionalidad de búsqueda de actividad aplicando el filtro por rango de fecha.



The image shows a screenshot of the AGADI web application interface. At the top, there is a header with the IDEC logo on the left and the AGADI logo on the right, with the text 'Aplicación de Gestión de Actividades de Docentes del IDEC' below it. Below the header is a navigation bar with the following items: 'AGADI', 'GESTIÓN DE ACTIVIDADES', 'BUSCAR ACTIVIDAD', 'REPORTES', 'ADMINISTRAR', and a user profile for 'LUIS SOLORZANO'. The main content area features a search form titled 'Buscar por rango de fechas: Proyectos'. The form has two input fields: 'Desde:' and 'Hasta:', both containing the placeholder text 'dd/mm/yyyy' and a calendar icon. Below the input fields are two green buttons: 'Regresar' and 'Buscar'. At the bottom of the page, there is a footer with the text: 'Instituto de Desarrollo Experimental de la Construcción | 2016', 'P. B. Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela, Ciudad Universitaria, Los Chaguaramos, Caracas. Apartado 47.169, Caracas 1041-A | (58-212) 605. 20. 46. Fax: (58-212) 605. 20. 48', and a Creative Commons license logo (CC BY-NC-ND).

Figura 41. Filtro de búsqueda por rango de fecha

5.2.3 Funcionalidad generar reportes

Esta funcionalidad solo podrá ser visualizada en el menú principal por los usuarios con rol jefes de departamento y administradores. La funcionalidad de generar reporte tiene como objetivo presentarle al usuario un menú de selección con el departamento de investigación, docencia y

extensión, cada selección brindará la posibilidad de generar un reporte específico del departamento seleccionado. Además esta funcionalidad cuenta con una cuarta selección que permite generar otros reportes, como la lista de investigadores que son jefes de departamento así como también permitirá generar una lista con la información de todos los investigadores que son miembros del I.D.E.C. especificando su estatus actual, el cual se puede definir como activo, jubilado o en permiso no remunerado.

En la siguiente figura 42 se presenta el menú de la funcionalidad de generar reporte.

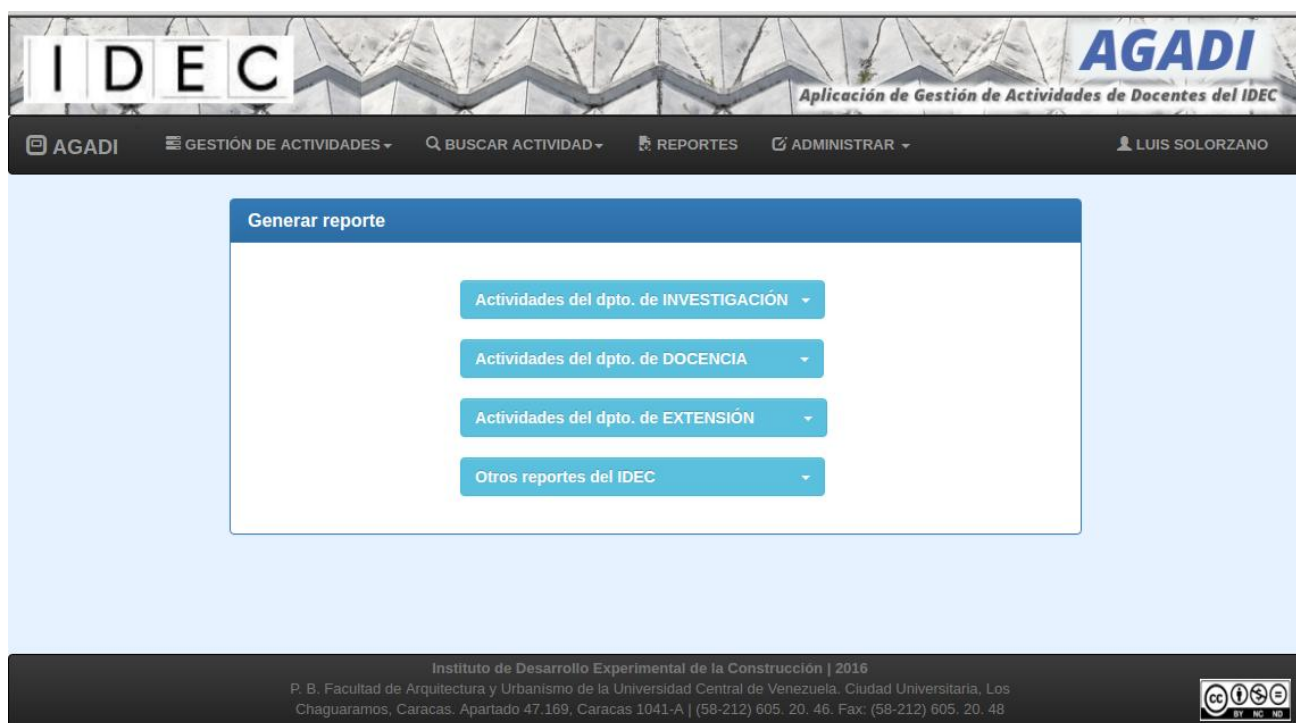


Figura 42. Funcionalidad generar reportes

Esta funcionalidad proporciona al jefe de departamento, herramientas que faciliten el trabajo realizado por este, brindándole la información de manera fácil y eficaz, acortando en gran medida el tiempo que el jefe de departamento utilizaba para esta tarea, además que permite reunir, analizar una gran volumen de actividades para presentar la información de manera clara y precisa, para así cumplir con la realización del informe de gestión.

A continuación en las figuras 43, 44 y 45 se presenta un ejemplo de generación de reporte del departamento de investigación, en este caso el resumen totalizador de proyectos de investigación. Presentando una tabla donde se contabiliza las cantidades de proyectos en grupo e individuales, complementando el reporte con una gráfica de barra, donde se observa la cantidad de proyectos en grupo e individual y por último la aplicación genera otra grafica en forma de torta donde proporciona al jefe de departamento, el porcentaje de las cantidades de proyectos antes mencionados.

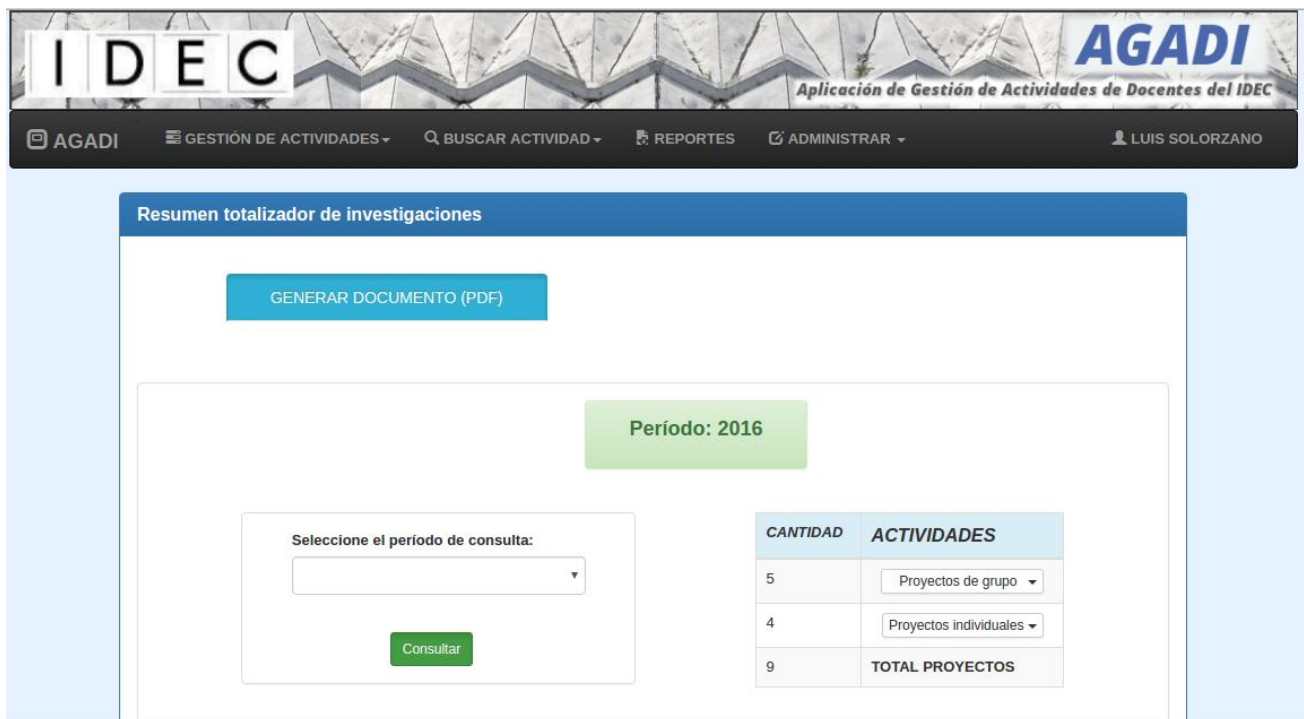


Figura 43. Ejemplo de reporte de resumen totalizador de proyectos



Figura 44. Ejemplo de reporte de gráfico de barra para proyectos de investigación



Figura 45. Ejemplo de reporte de gráfico de tota para proyectos de investigación

En la figura anterior 43, se muestra el recuadro donde el jefe de departamento podrá seleccionar el período para el cual desea generar la tabla de la figura 43, el gráfico de barra de la figura 44 y el gráfico de torta de la figura 45. Permitiendo así al jefe de departamento generar reportes históricos en cualquier momento de manera sencilla.

5.2.4 Funcionalidad administrar

Esta funcionalidad será visible en el menú principal solo por los usuarios con rol administrador, el objetivo de esta funcionalidad es proporcionar al administrador, la posibilidad de gestionar a los usuarios de la aplicación, proporcionando una lista completa de los usuarios registrados en AGADI y brindando unas serie de acciones como registrar usuario, modificar información del usuario, consultar en detalle la información de algún investigador, modificar la contraseña de ingreso de algún investigador, por último permitir que el administrador elimine alguno de los investigadores registrados. Esta función también permite generar un reporte en formato PDF con la lista de todos los usuarios registrados.

A continuación se presenta la figura 46 donde se observa la funcionalidad de administrar usuarios.

The screenshot shows the AGADI web application interface. At the top, there is a header with the IDEC logo on the left and the AGADI logo on the right, with the text "Aplicación de Gestión de Actividades de Docentes del IDEC" below it. Below the header is a navigation bar with the following items: AGADI, GESTIÓN DE ACTIVIDADES, BUSCAR ACTIVIDAD, REPORTES, ADMINISTRAR, and a user profile for LUIS SOLORIZANO. Below the navigation bar is a blue bar with the text "Menu de gestión de usuarios". Underneath this bar are two buttons: "CREAR INVESTIGADOR" and "GENERAR DOCUMENTO (PDF)". In the center, there is a button labeled "LISTA DE INVESTIGADORES". Below this button is a table with the following columns: Cedula, Primer Nombre, Primer Apellido, Rol, Cargo, and a set of action buttons (Mostrar, Editar, cambiar clave, Eliminar). The table contains four rows of data:

Cedula	Primer Nombre	Primer Apellido	Rol	Cargo	Mostrar	Editar	cambiar clave	Eliminar
15831533	daniel	trias	Jefe de Departamento	eeee	Mostrar	Editar	cambiar clave	Eliminar
16852963	dario	gonzalez	Jefe de Departamento	sfrdgtjdfgh	Mostrar	Editar	cambiar clave	Eliminar
15831534	jose	perez	Jefe de Departamento	tttt	Mostrar	Editar	cambiar clave	Eliminar
15831532	luis	solorzano	administrador	programador	Mostrar	Editar	cambiar clave	Eliminar

Figura 46. Funcionalidad administrar, menú gestión de usuarios

En la siguiente figura 47 se muestra el documento PDF generado de la figura anterior con la lista de todos los investigadores del I.D.E.C.



LISTA DE INVESTIGADORES

Fecha de emisión: 15/05/2016

CEDULA	PRIMER NOMBRE	PRIMER APELLIDO	EMAIL	ROL
15831533	daniel	trias	dansol0@gmail.com	Jefe de Departamento
16852963	dario	gonzalez	eeer@gmail.com	Jefe de Departamento
15831534	jose	perez	eee@gmail.com	Jefe de Departamento
15831532	luis	solorzano	unionucv@gmail.com	administrador
1000000	qq	ww	rrr@gmail.com	administrador
1234567	qqqqqqqq	wwwwww	epa@gmail.com	investigador
2000000	rrrrr	ttttttt	eeeee@gmail.com	investigador
74158963	rrrrr	wwwwww	uik@gmail.com	investigador

Figura 47. Reporte generado con lista de investigadores

En la figura 48 se muestra el formulario utilizado para registrar a un investigador en la aplicación AGADI.

Registro de investigadores

Los campos con (*) son obligatorios.

Cedula: (*)

Correo: (*)

Primer nombre: (*)

Segundo nombre:

Primer apellido: (*)

Segundo apellido:

Teléfono: (*)

Rol: (*)

Contraseña: (*)

Confirme la contraseña: (*)

Dirección:

Figura 48. Formulario de registro de investigador

Por último la funcionalidad de administrar, contará con la opción de gestionar períodos. Debido a que el informe de gestión de un período puede estar conformado por dos (2) años, se vio la necesidad al desarrollar la aplicación de asignarle a cada actividad un período, de manera que permita agrupar las actividades que pertenecen a un informe de gestión u otro, lo cual facilitará en gran medida al momento de realizar búsquedas históricas. Otros de los problemas que se buscó solucionar con la opción de gestionar los períodos, es que se podría dar el caso que se terminara un período y se iniciara el siguiente pero podrían existir investigadores que quisieran registrar actividades del período pasado.

Con esta opción en la funcionalidad de administrar, el administrador podrá conceder permisos para esta acción, debido a que este irá registrando los períodos y asignarle el estatus activo o desactivado, es decir si se le asigna a un período, el estatus activo se le permitirá a los investigadores que quieran registrar una actividad, que la registren con ese año y por el contrario si el administrador le asigna al período el estatus desactivado, el investigador al registrar cualquier actividad no podrá visualizar ese año. Lo normal es que el administrador vaya creando un nuevo período y le asigne el estatus de activo y al período pasado le asigne el estatus de inactivo para evitar que los investigadores puedan seguir registrando actividades en el período anterior.

A continuación se presenta la figura 49 en la cual se muestra el menú de gestión de períodos.



Figura 49. Interfaz administrar, opción gestionar períodos

Una vez presentada la aplicación AGADI y todas sus funcionalidades, como cierre del TEG, a continuación se presentan las conclusiones.

Conclusiones

Se puede concluir que la aplicación del método AgilUS en el desarrollo del proyecto, permitió implementar funcionalidades apegadas a los requerimientos del usuario debido a la constante retroalimentación. Los diferentes artefactos de AgilUS permitieron el diseño de una interfaz usable, tomando en cuenta determinados lineamientos y estándares. Por tratarse de un método iterativo e incremental, se desarrollaron las funcionalidades de manera progresiva, considerando al usuario en cada una de las etapas para el desarrollo de las mismas.

A su vez el empleo de herramientas de software libre, como el sistema de gestión de base de datos PostgreSQL y la utilización de Ruby on Rails como tecnología del lado del servidor, contribuyó a un desarrollo económico y respaldado debido a la gran cantidad de fuentes de información disponible acerca de estas tecnologías. Ruby on Rails agilizó considerablemente el proceso de desarrollo, ya que se aprovecharon las diversas estructuras y métodos predefinidos que posee esta tecnología, ya que, si se hubiese desarrollado la aplicación con otros lenguajes, hubiera requerido de un mayor esfuerzo y tiempo de codificación

La Aplicación AGADI permitirá mejorar en gran medida y en forma eficaz el proceso de registro de actividades llevado a cabo por cada investigador del Instituto. Cabe resaltar que el desarrollo es de gran importancia para el I.D.E.C. ya que permitirá contar con un repositorio dedicado a registrar las actividades realizadas en el departamento de investigación, docencia y extensión, dando así la posibilidad de realizar consultas históricas, lo cual antes era inexistente en el Instituto. Además, que beneficiará en gran medida a los jefes de departamento, ya que contarán con una herramienta que les permitirá acceder a la información de una manera más fácil y sencilla, contribuyendo a un gran ahorro de tiempo y recurso humano que antes era necesario para procesar el gran volumen de información generado por los investigadores, dando la posibilidad de generar reportes y presentar la información en tablas, así como reflejar las estadísticas en gráficos para su mejor comprensión, lo cual contribuirá enormemente a facilitar el informe de gestión del departamento del cual están a cargo.

Finalmente se puede afirmar que los resultados obtenidos fueron altamente satisfactorios debido a que se logró desarrollar una aplicación usable con un diseño claro, a su vez que se

diseñó una base de datos que permite almacenar las actividades registradas por los investigadores, pudiendo así implementar las diferentes funcionalidades antes descritas, las cuales forman parte de los objetivos principales de este TEG.

Recomendaciones

Para que la aplicación AGADI cumpla por completo con su objetivo se recomienda que todos los investigadores del I.D.E.C. registren a lo largo del año todas sus actividades a fin de que se nutra el repositorio de la aplicación y a su vez emerjan posibles propuestas de mejoras para trabajos futuros, que hagan de AGADI una aplicación cada vez más completa.

Todo software de calidad requiere mantenimiento adaptativo, correctivo y preventivo para garantizar su calidad en el tiempo. A continuación se listan las posibles mejoras y recomendaciones que se pueden hacer en función a este trabajo especial de grado:

- Realizar la debida promoción de AGADI que propicie su uso masivo por la comunidad del Instituto en virtud de evaluar su correcto funcionamiento una vez puesta la aplicación en producción.
- Instalar y configurar la aplicación AGADI en los servidores de la Facultad, lo cual permita que la aplicación sea usada fuera de la intranet por los investigadores del Instituto.
- Definir otros requerimientos de reporte en base al informe de gestión, que permita implementar tales requerimientos en trabajos futuros que complementen a la aplicación de este TEG.

Referencias

- Acosta, A., (2011). *AgilUs: Construcción ágil de la Usabilidad. Centro de Ingeniería de Software y Sistemas (ISYS)*. Escuela de Computación - Facultad de Ciencias - Universidad Central de Venezuela, Venezuela. Recuperado de: www.ciens.U.C.V..ve/escueladecomputacion/documentos/archivo/121.
- Álvarez, M.A (2008). *Introducción a CSS 3*. Recuperado de: <http://www.desarrolloweb.com/articulos/introduccion-css3.html>.
- Álvarez M., Monteiro J., & Méndez N., (2003). *Introducción a los lenguajes del web. Desarrollo Web*. Recuperado de: <http://www.ipereda.com/descargas/manuales/php/1.-Manual%20de%20Introducci%C3%B3n%20Lenguajes%20Web%20-%202022%20pag.pdf>
- Álvarez S. (2007). *Arquitectura cliente-servidor. Características principales de este tipo de arquitectura de cara a base de datos. Desarrollo Web*. Recuperado de: <http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>
- Arias A., y Portela F. (1997). *Sistema de información y sistema de calidad: relación y dependencia en las organizaciones empresariales. E.U.B.D.* Madrid: Universidad Complutense.
- Arreaza A., y Manyoma W. (2012). *Desarrollo de un módulo de gestión y planificación para los cursos de Extensión de la Escuela de Idiomas Modernos de la Facultad de Humanidades y Educación de la U.C.V.* (Tesis de Pregrado). U.C.V. Caracas.
- Ayala, J. (2013). *Arquitectura de aplicaciones web – Capa de presentación. Blog sobre arquitectura, multitarea, programación y tecnología web y mobile*. Recuperado de: <http://jmaw.blogspot.com/2013/01/arquitectura-de-aplicaciones-web-capa.html>
- Ayala, J. (2013). *Arquitectura de aplicaciones web – Capa de acceso a datos. Blog sobre arquitectura, multitarea, programación y tecnología web y mobile*. Recuperado de: http://jmaw.blogspot.com/2013/01/arquitectura-de-aplicaciones-web-capa_6972.html
- Ayala, J. (2013). *Arquitectura de aplicaciones web – Capa de negocio. Blog sobre arquitectura, multitarea, programación y tecnología web y mobile*. Recuperado

de: http://jmaw.blogspot.com/2013/01/arquitectura-de-aplicaciones-web-capa_5.html.

Beck, K. et al. (2001). *Manifiesto por el Desarrollo Ágil de Software*. Recuperado de: <http://www.agilemanifesto.org/iso/es/>

Beck, K. (1999). *Extreme Programming Explained. Embrace Change*. Pearson Education.

Berzal, F., Cortijo, F. y Cubero, J. C. (2005) Desarrollo profesional de aplicaciones web con asp.net. ISBN 84-609-4245-7.

Carneiro, C. y Al Barazi, R. (2010). *Beginning Rails 3*. Estados Unidos: Editorial APress

Canós, J., Patricio, L., y M. Carmen, P. (2012). *Metodologías Ágiles en el Desarrollo de Software*. Universidad Politécnica de Valencia. Recuperado de: http://noqualityinside.com/nqi/nqifiles/XP_Agil.pdf

Chin, G. (2004). *Agile Project Management: How to Succeed in the face of Changing Project Requirements*. AMACOM.

Creative Commons. (2015). *Creative Commons Venezuela*. Recuperado de <http://creativecommonsvenezuela.org/ve/>

Corabel, S. (2011). *Manejadores de Bases de Datos - SQL, ORACLE. Ilustrados*. Recuperado de: <http://www.ilustrados.com/tema/1285/Manejadores-Bases-Datos-ORACLE-INFORMIX.html>

Diez, T., Domínguez, M., Martínez J., & Sáenz, J. (2012). *Creación de páginas Web accesibles con HTML5*. Ecuador. Actas del IV Congreso Internacional ATICA 2012 – Loja.

Flanagan, D. (2011). *JavaScript: The Definitive Guide*, (6th Edition). Estados Unidos: Editorial O'Reilly Media

Fowler, M., (2005). *The New Methodology*. Recuperado de: <http://martinfowler.com/articles/newMethodology.html>

Gabardini, J., & Campos, L. (2004). *Balanceo de Metodologías Orientadas al Plan y Ágiles*.

Herramientas para la Selección y Adaptación. Argentina. PMI Global Congress Proceedings

Galan, M. (2009). *La entrevista en investigación. Metodología de la investigación.* Recuperado de: <http://manuelgalan.blogspot.com/2009/05/la-entrevista-en-investigacion.html>.

Garrett, J. (2005). *Ajax: A New Approach to Web Applications. Adaptive path.* Recuperado de: <http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications/>

Geschwinde, E., & Schönig, H.-J. (2002). *PostgreSQL Developer's Handbook.* Sams Publishing.

Ginestà M., & Mora O., (2007). *Bases de datos en PostgreSQL.* Universidad Oberta de Catalunya. Recuperado de: http://informatica.gonzalonazareno.org/plataforma/file.php/3/Apuntes/UOC_Postgres.pdf

Gómez A., et al, (2005). *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web.* España: Ediciones Díaz de Santos

Jeffries, R., Anderson, A., & Hendrickson, C. (2001). *Extreme Programming Installed.* Boston, USA: Addison-Wesley.

Larman, C. (2003). *UML y Patrones.* 2ª Edición. Prentice Hall.

Letelier P. (2006). *Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP).* Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. Recuperado de: http://www.cyta.com.ar/ta0502/b_v5n2a1.htm.

Márquez, B. (2004). *Implementación de un reconocedor de voz gratuito al sistema de ayuda a invidentes Dos-Vox en español.* Catarina. Recuperado de: http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf.

Mora, L. (2002). *Programación de aplicaciones web: historia, principios básicos y cliente web.* Alicante, España: Editorial Club Universitario.

- Ortega, J. (2013). *Metodología de desarrollo de software. Paradigmas de programación*. Recuperado de: <http://paradigmasiut.blogspot.com/2013/04/metodologia-de-desarrollo-de-software.html>.
- Paré, R. C. (2002). *Introducción a las bases de datos*. UOC.
- PHP. (s.f.). *¿Qué puede hacer PHP?* Recuperado de: <http://www.php.net/manual/es/intro-whatcando.php>
- Ralph Schulz. (2009). *Diseño web con CSS*. Barcelona: Marcombo ediciones técnicas
- Serrano, J. (2007). *Explicando Scrum a mi Abuela*. Recuperado de: <http://geeks.ms/blogs/jorge/archive/2007/05/09/explicando-scrum-a-mi-abuela.aspx>
- Silberschatz, A., (2002). *Fundamentos de bases de datos*. (Cuarta edición). Aravaca, Madrid: Mc Graw Hill.
- Sommerville, I. (2005). *Ingeniería del software* (Séptima Edición). España: Pearson Educación.
- Teixidor, S (2010). *Modelo Vista Controlador y algunas variantes. Neleste*. Recuperado de: <http://www.neleste.com/modelo-vista-controlador-y-algunas-variantes/>
- Thibaud, C. (2006). *MySQL 5, Instalación, implementación, administración, programación*. Cornell de Llobregat. Barcelona: Ediciones ENI.
- Valarezo, E. & Luján, S. (2014). *Aplicaciones Web - Diseño para Aplicaciones Web*. Universidad de Alicante. Departamento de Lenguajes y Sistemas Informáticos.
- Garcia, J., & Yanez, F. (2014). *Estudio teórico para la construcción de un generador de objetos de aprendizaje de contenido abierto, basado en estrategias de razonamiento para la resolución de problemas de principios de química* (Tesis de pregrado). Universidad Central de Venezuela, Caracas.
- Zapata J., Carlos M.; Tamayo O., Paula A.; Arango I., & Fernando (2007). Conversión de esquemas preconceptuales a diagrama de casos de uso empleando atom3. *Dyna*. 74 (153), 237-251. Universidad Nacional de Colombia Medellín, Colombia.

Anexos A

Descripción de casos de uso

Tabla 2. Descripción caso de uso de nivel 0 – actor administrador

Actor	ACT. 1 Administrador
Descripción	Este actor tiene una visión general de la aplicación web. Puede crear/eliminar jefes de departamento y profesores. Puede ver información completa creada por los jefes de departamento y profesores.

Tabla 3. Descripción caso de uso de nivel 0 – actor jefe de departamento

Actor	ACT. 1 Jefe de departamento
Descripción	Tiene acceso a los reportes realizados por los investigadores dependiendo de sudepartamento. Debe generar reportes de los investigadores correspondientes a su departamento.

Tabla 4. Descripción caso de uso de nivel 0 – actor investigador

Actor	ACT. 1 Investigador
Descripción	Deben generar reportes por cada actividad realizada, ya sea: Investigación, Docencia, Extensión. Además pueden crear reportes de información personal, como: Mejoramiento y capacitación, Actividades académicas

Tabla 5. Descripción opción 1 - caso de uso de nivel 1 – actor investigador – jefe de departamento – administrador

Caso de Uso	1.Autenticar
Actor	profesor – jefe de departamento - administrador
Descripción	Acción que se encarga de autenticar en la aplicación web al actor.
Precondiciones	ninguna
Postcondiciones	Se autentica o deniega el acceso al usuario

Tabla 6. Descripción opción 2 - caso de uso de nivel 1 – actor profesor – jefe de departamento

Caso de Uso	2. Gestionar Actividad
Actor	profesor y jefe de departamento
Descripción	Acción que se encarga de gestionar actividades (consultar, eliminar, modificar, crear), luego de señalar el departamento a la cual pertenece la actividad a gestionar
Precondiciones	Estar autenticado
Postcondiciones	Visualiza, crea, modifica o elimina la actividad, dependiendo del caso

Tabla 7. Descripción opción 3 - caso de uso de nivel 1 – actor jefe de departamento /administrador

Caso de Uso	3.Generar Reportes
Actor	Profesor / jefe de departamento / Administrador
Descripción	Acción que se encarga de consultar en la aplicación web los registros de actividades realizados.
Precondiciones	Estar autenticado en la aplicación web
Postcondiciones	Se muestra información de los reportes realizados por los profesores.

Tabla 8. Descripción opción 4 - caso de uso de nivel 1 – actor jefe de departamento /administrado

Caso de Uso	4.Buscar Actividad
--------------------	--------------------

Actor	jefe de departamento /Administrador
Descripción	Acción que se encarga de realizar consultas de actividades o formularios.
Precondiciones	Ingresar a la aplicación web como jefe de departamento o Administrador
Postcondiciones	Se visualiza la información.

Tabla 9. Descripción opción 5 - caso de uso de nivel 1 – actor administrador

Caso de Uso	5.Gestionar Usuario
Actor	Administrador
Descripción	Acción que se encarga de gestionar un usuario en la aplicación web (crear, modificar, eliminar, consultar).
Precondiciones	Ingresar a la aplicación web como Administrador
Postcondiciones	Se creará, eliminará, modificará o consultará la información de un usuario en la aplicación web.

Tabla 10. Descripción opción 2.1 - caso de uso de nivel 1 – actores investigador / jefe de departamento

Caso de Uso	2.1. Registrar Actividad
Actor	Profesor /jefe de departamento
Descripción	Acción que se encarga de registrar en la aplicación web la actividad realizada por el actor.
Precondiciones	Haber seleccionado alguna de las cinco clasificaciones de actividades.
Postcondiciones	Se registra en la aplicación web.

Caso de Uso	2.2.Modificar Actividad
Actor	profesor / jefe de departamento
Descripción	Acción que se encarga de modificar en el sistema la actividad realizada por el actor.
Precondiciones	Haber seleccionado alguna de las cinco clasificaciones de actividades. Debe estar registrada en el sistema la actividad a modificar
Postcondiciones	Se actualizara la nueva información en el sistema de la actividad.

Tabla 8. Descripción opción 2.2 - caso de uso de nivel 2 – actores investigador / jefe de departamento

Caso de Uso	2.3.Eliminar Actividad
Actor	Profesor / jefe de departamento
Descripción	Acción que se encarga de eliminar del sistema la actividad realizada por el actor.
Precondiciones	Haber seleccionado alguna de las cinco clasificaciones de actividades. Debe estar creada en el sistema la actividad
Postcondiciones	Se elimina del sistema la actividad.

Tabla 12. Descripción opción 2.3 - caso de uso de nivel 2 – actores profesor / jefe de departamento

Tabla 13. Descripción opción 2.4 - caso de uso de nivel 2 – actores profesor / jefe de departamento

Caso de Uso	2.4.Buscar Actividad
Actor	profesor / jefe de departamento
Descripción	Acción que se encarga de buscar en la aplicación web la actividad realizada por el actor.
Precondiciones	Haber seleccionado alguna de las cinco clasificaciones de actividades. Debe estar creada en la aplicación web la actividad
Postcondiciones	Se muestra la actividad.