

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Facultad de Ingeniería en Electricidad y Computación

Maestría en Sistemas de Información Gerencial

**“IMPLEMENTACIÓN DE UNA APLICACIÓN PARA EL
SEGUIMIENTO DE EGRESADOS Y DE INSERCIÓN LABORAL
PARA LA UNIDAD EDUCATIVA REDEMPTIO DE LA CIUDAD DE
JIPIJAPA.”**

TRABAJO DE TITULACIÓN

Previa a la obtención del grado de:

MAGISTER EN SISTEMAS DE INFORMACIÓN GERENCIAL

ANDRÉS FERNANDO GRANJA MOREIRA

GUAYAQUIL – ECUADOR

AÑO: 2018

AGRADECIMIENTO

A Dios

A mis padres: Fernando Granja Carrión y Mirian Moreira Moreira, por su apoyo y cariño incondicional, este triunfo es de ustedes.

A mi novia Ing. Katty Mishell Burgos Santiana.

A la Escuela Superior Politécnica del Litoral, por su labor, que contribuye a la construcción de profesionales íntegros y competentes.

A mi Director de Tesis: Ing. Lenin Freire Cobo, su aporte ha sido representativo para el éxito del presente trabajo.

Al personal docente, administrativo y de servicio del campus Las Peñas, por su generosa labor desempeñada.

Mi eterna gratitud.

DEDICATORIA

Dedico este proyecto de tesis a Dios y a mis padres.

A Dios que ha estado conmigo en todo momento, cuidándome y dándome fortaleza para continuar.

A mis padres, quienes a lo largo de mi vida han velado por mi educación, siendo mi apoyo en todo momento, depositando su entera confianza en cada reto que se me presentaba. Los amo con mi vida.

TRIBUNAL DE SUSTENTACIÓN

Mgs. Lenin Freire

DIRECTOR DEL MSIG/MSIA

Mgs. Lenin Freire

DIRECTOR DEL PROYECTO DE GRADUACIÓN

Mgs. Omar Maldonado

MIEMBRO DEL TRIBUNAL

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de este Trabajo de Titulación, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la **Escuela Superior Politécnica del Litoral**".

Ing. Andrés Fernando Granja Moreira

RESUMEN

El capítulo uno trata sobre las generalidades del proyecto; la descripción del problema donde se plantea la situación de la Unidad Educativa, la solución de la propuesta donde se expone y justifica las razones a realizar el proyecto.

El capítulo dos contiene el Marco Teórico, aquí se encuentra una breve descripción de las herramientas técnicas y tecnologías que se usan en la solución propuesta.

El capítulo tres contiene los Requerimientos y Metodología, en el cual se realizó el levantamiento de la información utilizando la técnica JAD y para la metodología, se formó un equipo SCRUM y se definieron los actores.

El capítulo cuatro contiene el Análisis de la Solución, donde se realizó estudios de factibilidad (técnica y operativa); costos de desarrollo y financiamiento y análisis de costo / beneficio.

El capítulo cinco contiene el Diseño y Desarrollo de la Solución, donde se realizó la construcción y verificación, de acuerdo a los requerimientos establecidos.

El capítulo seis contiene el Análisis de Resultados, en donde se realizó una recolección de datos, que nos permitió conocer si los objetivos planteados se cumplen a consecuencia de implementar la aplicación.

ÍNDICE GENERAL

RESUMEN.....	VI
ÍNDICE GENERAL.....	VIII
ABREVIATURAS.....	XV
ÍNDICE DE FIGURAS.....	XVI
ÍNDICE DE TABLAS.....	XIX
INTRODUCCIÓN.....	XXII
CAPÍTULO 1.....	1
1. GENERALIDADES.....	1
1.1. DESCRIPCIÓN DEL PROBLEMA.....	1
1.2. SOLUCIÓN DE PROPUESTA.....	3
1.3. OBJETIVO GENERAL.....	5
1.4. OBJETIVOS ESPECÍFICOS.....	5
CAPÍTULO 2.....	6
2. MARCO TEÓRICO.....	6
2.1. ANTECEDENTES DE LA INSTITUCIÓN.....	6
2.2. FUNDAMENTOS TECNOLÓGICOS.....	6
2.2.1. <i>Aplicación Web</i>	6
2.2.2. <i>REST</i>	7
2.2.2.1. Uso correcto de URLs:.....	8

2.2.2.2. Uso correcto de HTTP	9
2.2.2.3. Implementar Hypermedia.....	9
2.2.3. <i>Aplicaciones SPA(Single Page Applications)</i>	10
2.2.3.1. Arquitectura SPA	11
2.3. LENGUAJES DE PROGRAMACIÓN.....	14
2.3.1. <i>MEAN</i>	14
2.3.2. <i>Node.js</i>	15
2.3.3. <i>Express.js</i>	16
2.3.4. <i>Angular</i>	18
2.3.5. <i>MongoDB</i>	20
2.4. DEFINICIÓN DE REQUISITOS	22
2.4.1. <i>Levantamiento de requerimientos usando JAD</i>	22
2.4.1.1. Patrocinador del Proyecto.....	23
2.4.1.2. Líder del Proyecto.....	24
2.4.1.3. Clientes.....	24
2.4.2. <i>Casos de Uso</i>	25
2.4.2.1. Actor	25
2.4.2.2. Caso de uso.....	26
2.4.2.3. Asociaciones.....	26
2.5. METODOLOGÍAS	27
2.5.1. <i>SCRUM</i>	28
CAPÍTULO 3.....	30

3. REQUERIMIENTOS Y METODOLOGÍA	31
3.1. INTRODUCCIÓN	30
3.2. LEVANTAMIENTO DE REQUERIMIENTOS	30
3.3. DEFINICIÓN DE CASOS DE USO.....	32
3.3.1. <i>Registro de Usuarios</i>	33
3.3.2. <i>Ingreso al Sistema</i>	34
3.3.3. <i>Información de los Egresados</i>	35
3.3.4. <i>Información de los Empleadores</i>	37
3.3.5. <i>Encuestas Administrador</i>	39
3.3.6. <i>Encuesta Egresados</i>	40
3.3.7. <i>Ofertas Egresados</i>	41
3.3.8. <i>Ofertas Empleadores</i>	42
3.4. IMPLEMENTANDO SCRUM	42
3.4.1. <i>Selección del Equipo Scrum</i>	42
3.4.2. <i>Planeación de la Iteración (Sprint Planning)</i>	43
3.4.3. <i>Ejecución de la Iteración (Sprint)</i>	44
3.4.4. <i>Reunión Diaria de Sincronización del Equipo (Scrum Daily Meeting)</i>	44
3.4.5. <i>Demostración de requisitos completados (Sprint Demonstration)</i>	44
3.4.6. <i>Retrospectiva (Sprint Retrospective)</i>	45
3.4.7. <i>Re-planificación del proyecto</i>	45

3.5.	HISTORIAS DE USUARIOS.....	45
3.5.1.	<i>Detalle de las Historias de Usuarios.....</i>	46
3.6.	ITERACIONES	52
3.6.1.	<i>Iteración 01.....</i>	52
3.6.2.	<i>Iteración 02.....</i>	53
3.6.3.	<i>Iteración 03.....</i>	53
3.6.4.	<i>Iteración 04.....</i>	54
3.6.5.	<i>Cronograma de la Tareas.....</i>	55
3.7.	MOCKUPS.....	55
	CAPÍTULO 4.....	62
4.	ANÁLISIS DE LA SOLUCIÓN.....	63
4.1.	ESTUDIO DEL DESARROLLO DE LA PROPUESTA.....	62
4.1.1.	<i>Descripción de la Propuesta.....</i>	62
4.1.2.	<i>Alcance de la Propuesta.....</i>	63
4.2.	ESTUDIO DE FACTIBILIDAD	64
4.2.1.	<i>Factibilidad Técnica.....</i>	64
4.2.2.	<i>Factibilidad Operativa.....</i>	65
4.3.	COSTOS DE DESARROLLO Y FINANCIAMIENTO.....	66
4.3.1.	<i>Costos de Desarrollo.....</i>	66
4.3.2.	<i>Financiamiento.....</i>	67
4.4.	ANÁLISIS DE COSTOS / BENEFICIOS	67
4.4.1.	<i>Costo.....</i>	67

4.4.2.	<i>Beneficio</i>	67
4.4.3.	<i>Relación Costo / Beneficio</i>	68
4.4.4.	<i>Análisis de Retorno de Inversión</i>	69
4.4.5.	<i>Tiempo de Retorno de Inversión</i>	69
CAPÍTULO 5.....		71
5.	DISEÑO Y DESARROLLO DE LA SOLUCIÓN	72
5.1.	DISEÑO LÓGICO DE LA SOLUCIÓN	71
5.1.1.	<i>Diseño de Salida</i>	71
5.1.2.	<i>Diseño de Entrada</i>	72
5.1.3.	<i>Diagrama de Estructura de Datos</i>	72
5.1.4.	<i>Diccionario de Datos</i>	73
5.2.	DISEÑO FÍSICO DE LA SOLUCIÓN	79
5.2.1.	<i>Modelo Cliente-Servidor</i>	79
5.2.2.	<i>Ventajas y Desventajas del Modelo</i>	80
5.3.	OPCIONES DE DESARROLLO DE SOFTWARE.....	80
5.3.1.	<i>Técnicas de Desarrollo</i>	81
5.3.1.1.	<i>Programación Orientada a Objetivos (P.O.O.)</i>	81
5.3.1.2.	<i>Arquitectura de Capas</i>	81
5.3.2.	<i>Herramientas de Desarrollo</i>	82
5.3.2.1.	<i>Survey.JS</i>	82
5.4.	IMPLEMENTACIÓN	82
5.4.1.	<i>Aplicaciones Necesarias para el Funcionamiento</i>	82

5.4.2.	<i>Interfaz de la Aplicación</i>	83
5.4.2.1.	Formulario Ingreso a la Aplicación.....	83
5.4.2.2.	Formulario Dashboard de Egresados	84
5.4.2.3.	Formulario Actualización Egresado	85
5.4.2.4.	Formulario Encuesta Egresado.....	85
5.4.2.5.	Formulario Ofertas Laborales Egresados	86
5.4.2.6.	Formulario Dashboard de Empresa/Empleador.....	87
5.4.2.7.	Formulario Actualización Empresa/Empleador	87
5.4.2.8.	Formulario Ofertas Laborales Empresa/Empleador.....	88
5.4.2.9.	Formulario Dashboard Administrador	90
5.4.2.10.	Formulario Nuevos Usuarios.....	90
5.4.3.	<i>Cronograma de Actividades para la Implementación</i>	91
5.5.	DESCRIPCIÓN DEL ENTORNO DE PRUEBA	92
5.5.1.	<i>Selección de Métricas</i>	92
5.5.2.	<i>Aplicación de Métricas</i>	94
5.5.3.	<i>Análisis de Métricas</i>	95
CAPÍTULO 6	96
6. ANÁLISIS DE RESULTADOS	98
6.1.	INTRODUCCIÓN	96
6.2.	RESULTADOS OBTENIDOS EN LAS PRUEBAS DE USUARIOS	96
6.2.1.	<i>Instrumento #1 – Usabilidad</i>	97
6.2.2.	<i>Instrumento #2 – Seguridad</i>	99

6.2.3. <i>Instrumento #3 – Mantenibilidad</i>	100
6.2.4. <i>Instrumento #4</i>	102
CONCLUSIONES Y RECOMENDACIONES	105
CONCLUSIONES	105
RECOMENDACIONES	106
BIBLIOGRAFÍA.....	107
ANEXO N.1.....	110
ANEXO N 2.....	111
ANEXO N.3.....	111
ANEXO N.4.....	112
ANEXO N.5.....	113

ABREVIATURAS Y SIMBOLOGIAS

API:	Application Programming Interface
BJSON:	Binary JavaScript Object Notation
CRUD:	Create Read Update Delete
CSS:	Cascading Stylesheets
HTML:	HiperText Markup Language
HTTP:	HiperText Transfer Protocol
JAD:	Joint Application Design
JSON:	JavaScript Object Notation
MEAN:	Mongo Express Angular Node
MVC:	Model View Controller
NPM:	Node Package Manager
POO:	Programación Orientada a Objetos
REST:	Representational State Transfer
ROI:	Return On Investment
SPA:	Single Page Application
TCP/IP:	Transfer Control Protocol / Internet Protocol
URL:	Uniform Resource Locator

ÍNDICE DE FIGURAS

Figura 2.1. Esquema basado en API REST	10
Figura 2.2. Ciclo de vida de una aplicación SPA y tradicional	13
Figura 2.3. Procesamiento asíncrono de Node.js	15
Figura 2.4. Esquema de arquitectura MVC	18
Figura 2.5. Funcionamiento del flujo de datos bidireccional de Angular	19
Figura 2.6. Componentes de un diagrama de casos de uso.....	27
Figura 2.7. Ciclo de vida SCRUM	29
Figura 3.1.Caso de uso registro de usuarios.	33
Figura 3.2. Caso de uso ingreso al sistema.....	34
Figura 3.3.Caso de uso información de los egresados.....	35
Figura 3.4.Caso de uso información de los empleadores	37
Figura 3.5.Caso de uso encuestas administrador.....	39
Figura 3.6. Caso de uso encuesta egresados.	40
Figura 3.7.Caso de uso ofertas egresados.	41
Figura 3.8. Caso de uso ofertas empleadores.	42
Figura 3.9. Cronograma de las tareas por iteraciones.	55
Figura 3.10. Mockup-01 autenticación al sistema.....	56
Figura 3.11. Mockup-02 registro de usuario al sistema 1	56
Figura 3.12. Mockup-03 registro de usuario al sistema 2	57
Figura 3.13. Mockup-04 edición de usuario del sistema.....	57
Figura 3.14. Mockup-05 actualización datos egresados.....	58

Figura 3.15. Mockup-06 aplicación de encuesta.....	58
Figura 3.16. Mockup-07 creación de ofertas laborales.	59
Figura 3.17. Mockup-08 aplicación de ofertas laborales 1	59
Figura 3.18. Mockup-09 aplicación de ofertas laborales 2.....	60
Figura 3.19. Mockup-10 publicación de ofertas laborales.....	60
Figura 3.20. Mockup-11 visualización encuestas.....	61
Figura 3.21. Mockup-12 cambio de contraseña.	61
Figura 5.1. Diagrama de Estructura de Datos.....	73
Figura 5.2. Arquitectura física de la aplicación.....	79
Figura 5.3.Arquitectura de capas de la solución.....	81
Figura 5.4. Formulario ingreso a la aplicación.	84
Figura 5.5. Formulario dashboard de egresados.	84
Figura 5.6. Formulario actualización egresado.	85
Figura 5.7. Formulario encuesta egresado.	85
Figura 5.8. Formulario lista de ofertas laborales.....	86
Figura 5.9. Formulario detalle de oferta laboral	86
Figura 5.10. Formulario dashboard de empresa/empleador.	87
Figura 5.11. Formulario actualización empresa/empleador.	87
Figura 5.12. Formulario ofertas laborales empresa/empleador.....	88
Figura 5.13. Formulario nueva oferta laboral.	88
Figura 5.14. Formulario actualización de oferta laboral.	89
Figura 5.15. Formulario de publicación oferta laboral.....	89

Figura 5.16. Formulario dashboard administrador.	90
Figura 5.17. Formulario lista de usuarios de la aplicación.	90
Figura 5.18. Formulario nuevo usuario de la aplicación.....	91
Figura 5.19. Cronograma actividades de implementación.	91

ÍNDICE DE TABLAS

Tabla 1. Ventajas y desventajas de aplicaciones SPA.	12
Tabla 2. Equivalencias entre SQL y MongoDB	22
Tabla 3. Diferencias entre metodologías tradicionales y ágiles.	28
Tabla 4. CU-01 Registro de usuario del sistema.....	33
Tabla 5. CU-02 Validar información usuario del sistema.	33
Tabla 6. CU-03 Ingreso al sistema.....	34
Tabla 7. CU-04 Verificar Acceso al Sistema.	34
Tabla 8. CU-05 Verificar acceso según rol.	35
Tabla 9. CU-06 Registrar datos egresados.....	36
Tabla 10. CU-07 Actualizar datos egresado.	36
Tabla 11. CU-08 Consultar datos egresado.....	37
Tabla 12. CU-09 Registrar datos empleador	38
Tabla 13. CU-10 Actualizar datos empleador.	38
Tabla 14. CU-11 Consultar datos empleador.....	39
Tabla 15. CU-14 Ver resultados de la encuesta.	40
Tabla 16. CU-15 Responder encuesta.....	40
Tabla 17. CU-16 Ver ofertas.	41
Tabla 18. CU-17 Aplicar Ofertas.....	41
Tabla 19. CU-18 Crear ofertas.....	42
Tabla 20. Historias de usuarios.....	46
Tabla 21. HU-01 Autenticación al sistema	46

Tabla 22. HU-02 Registro de usuarios del sistema.....	47
Tabla 23. HU-03 Edición de usuarios del sistema	47
Tabla 24. HU-04 Actualización datos de egresados	48
Tabla 25. HU-05 Elaboración de encuesta.	48
Tabla 26. HU-06 Aplicación de encuesta.....	49
Tabla 27. HU-07 Actualización datos de empleadores	49
Tabla 28. HU-08 Creación de ofertas laborales.	50
Tabla 29. HU-09 Aplicación de ofertas laborales.	50
Tabla 30. HU-10 Publicación de ofertas laborales.....	51
Tabla 31. HU-11 Visualización datos de encuestas.....	51
Tabla 32. HU-12 Cambio de contraseña.....	51
Tabla 33. Detalle de iteración 01	52
Tabla 34. Detalle de iteración 02	53
Tabla 35. Detalle de iteración 03	54
Tabla 36. Detalle de iteración 04	54
Tabla 37. Requisitos de hardware para la implementación del producto.....	64
Tabla 38. Requisitos de software para la implementación del producto.	65
Tabla 39. Costo de recursos del proyecto	66
Tabla 40. Costos totales del proyecto.....	67
Tabla 41. Costo de inversión.	67
Tabla 42. Beneficios de inversión.	68
Tabla 43. Sueldo del personal vinculado al proyecto.....	69

Tabla 44. Ahorro por implementación.....	70
Tabla 45.Tabla de usuarios.	74
Tabla 46.Tabla de empresas.	74
Tabla 47.Tabla de encuestas.....	75
Tabla 48.Tabla datos de la encuesta.	75
Tabla 49.Tabla de universidades.....	76
Tabla 50.Tabla de carreras universitarias.....	76
Tabla 51. Tabla egresado.....	77
Tabla 52.Tabla de las postulaciones.....	78
Tabla 53.Tabla de las ofertas.....	78
Tabla 54.Métrica de funcionalidad – adecuación.....	93
Tabla 55. Métrica de usabilidad – comprensión.....	93
Tabla 56.Aplicando métrica de funcionalidad.	94
Tabla 57. Aplicando métrica de usabilidad.	94
Tabla 58. Resultado del instrumento #1 usabilidad - iteración 01.....	97
Tabla 59. Resultado del instrumento #1 usabilidad - iteración 02.....	98
Tabla 60. Resultado del instrumento #1 usabilidad - iteración 03.....	98
Tabla 61. Resultado del instrumento #1 usabilidad - iteración 04.....	98
Tabla 62. Resultado del instrumento #2 seguridad.....	99
Tabla 63. Resultado del instrumento #3 mantenibilidad – iteración 01	100
Tabla 64. Resultado del instrumento #3 mantenibilidad - iteración 02.....	100
Tabla 65. Resultado del instrumento #3 mantenibilidad - iteración 03.....	101

Tabla 66. Resultado del instrumento #3 mantenibilidad - iteración 04..... 101

INTRODUCCIÓN

La demanda que vive actualmente nuestro país, sus necesidades y sus lagunas reclaman fuertemente al sector educativo de nivel medio para obtener una respuesta o servicio que logre satisfacer sus necesidades. No obstante, en muchas ocasiones se hace notoria la incoherencia entre los procesos educativos, los requerimientos sociales y las exigencias del mercado laboral.

Los procesos de seguimiento de egresados no tienen relación exclusivamente en la inserción de los estudiantes egresados o graduados en el ambiente laboral o la continuación de sus estudios de nivel superior, tampoco son sólo indicadores de satisfacción del estudiante en relación con la enseñanza recibida, son también maneras muy eficaces para lograr conciencia institucional sobre sus objetivos, de manera que la implementación de un sistema informático se convierte en una herramienta que permita proveer datos valiosos con el propósito de cubrir y reajustar puntos débiles.

CAPÍTULO 1

GENERALIDADES

1.1. Descripción del Problema

La demanda que vive actualmente nuestro país, sus necesidades y sus lagunas reclaman fuertemente al sector educativo de nivel medio para obtener una respuesta o servicio que logre satisfacer sus necesidades. No obstante, en muchas ocasiones se hace notoria la incoherencia entre los procesos educativos, los requerimientos sociales y las exigencias del mercado laboral.

Es por ello, que la Unidad Educativa Particular Redemptio de la ciudad de Jipijapa requiere de evaluaciones continuas sobre sus fortalezas y debilidades, ya que ésta es la herramienta fundamental para planear y ejecutar proyectos necesarios y oportunos que permitan mejorar la formación de sus estudiantes, calificados para evaluar su calidad de vida y satisfacer los distintos requerimientos de la sociedad.

La relación que la institución mantiene con las empresas ya sean públicas o privadas y sus estudiantes egresados o graduados se encuentra de alguna forma limitada, los canales de comunicación y retroalimentación han sido débiles al momento de conseguir respuestas a temas como satisfacción profesional, demandas laborales objetivo y situación laboral, perfiles adecuados; es por ello, que la Unidad Educativa Particular Redemptio necesita implementar una solución que le permita tener acceso a toda la información indispensable sobre sus estudiantes egresados o graduados para conocer y seguir con el crecimiento profesional y/o laboral de los mismos, luego de culminar su educación secundaria.

La solución de seguimiento a graduados establecerá conexiones de doble vía entre la institución y el ambiente laboral, ya que los estudiantes graduados son los delegados en realizar la retroalimentación de las fortalezas y debilidades de la Unidad Educativa.

1.2. Solución de Propuesta

Los procesos de seguimiento de egresados no tienen relación exclusivamente en la inserción de los estudiantes egresados o graduados en el ambiente laboral o la continuación de sus estudios de nivel superior, tampoco son sólo indicadores de satisfacción del estudiante en relación con la enseñanza recibida, son también maneras muy eficaces para lograr conciencia institucional sobre sus objetivos, de manera que la implementación de un sistema informático se convierte en una herramienta que permita proveer datos valiosos con el propósito de cubrir y reajustar puntos débiles.

El sistema contemplará el registro de los candidatos de la Unidad Educativa Redemptio, la administración de la hoja de vida, privacidad del currículum, administración de contraseñas, visualización de postulaciones y encuestas a egresados. En las empresas se incluye el registro en la aplicación, publicación de ofertas laborales, administración de postulaciones, filtros de currículum, administración de contraseñas y recuperación de contraseña. La gestión y la socialización con las empresas ofertantes es responsabilidad del personal calificado asignado por la institución educativa.

La elaboración del sistema o desarrollo fullstack se basará en MEAN stack 2.0, que consta de un conjunto de subsistemas que se integran de manera exitosa, que permitirá obtener una herramienta poderosa, escalable y veloz. Dentro del conjunto de tecnologías se tiene:

- MongoDB es la base de datos NoSQL líder en el mercado que permite trabajar con documentos JSON y BSON, lo que puede ser leído por cualquier lenguaje de programación.
- Node.js es una tecnología que permite desarrollos del lado del servidor utilizando simplemente JavaScript.
- Angular 2 es un marco de trabajo para desarrollar aplicaciones modernas y escalables del lado del cliente, implementando JavaScript.

Los beneficios tales como la retroalimentación de la instrucción y empleo para el egresado se relacionan con la realización de análisis necesarios para la toma de decisiones, tomando en relación que el estudiante egresado es la referencia que tiene el establecimiento educativo.

1.3. Objetivo General

Implementar una aplicación para el seguimiento de egresados y de inserción laboral para la Unidad Educativa Particular Redemptio de la ciudad de Jipijapa.

1.4. Objetivos Específicos

- Identificar los problemas y establecer requerimientos.
- Elaborar un estudio de factibilidad de la solución propuesta.
- Desarrollar la solución propuesta según los requerimientos establecidos.
- Realizar pruebas con las personas involucradas para garantizar el correcto funcionamiento de la aplicación.
- Analizar los resultados y proponer soluciones de mejora.

CAPÍTULO 2

MARCO TEÓRICO

2.1. Antecedentes de la Institución

La Unidad Educativa Particular Redemptio de la ciudad de Jipijapa no cuenta con referencias de investigación, trabajos realizados o evidencia de utilización de sistemas informáticos acerca del tema en estudio, por lo que la información es muy limitada, soportada simplemente por formatos en físico con información básica sobre los estudiantes graduados.

Debido a esta falta de control, la implementación de una solución informática para el seguimiento de egresados se vuelve necesaria para el crecimiento de la institución y reconocer las debilidades y fortalezas, con el objetivo de plantear proyectos de mejora continua en la institución.

2.2. Fundamentos Tecnológicos

2.2.1. Aplicación Web

Se la denomina aplicación web a una solución informática donde su interfaz de usuario puede ser accedida desde un cliente web, habitualmente un navegador web o browser. En la actualidad las aplicaciones web se han vuelto muy populares debido al uso práctico que tienen los navegadores web, y así mismo la autonomía del sistema operativo.

De acuerdo a sus características más comunes podemos decir:

- Comunicación a través de HTTP con el protocolo TCP/IP.
- Acceso estable a una o varias bases de datos.
- Arquitectura de N-capas.
- Tipos de usuarios accediendo a un rol específico.

2.2.2. REST

REST [1] es una clase de arquitecta que se utiliza en el desarrollo de aplicaciones web bajo el protocolo HTTP para su respectivo funcionamiento, de tal manera que no se necesita de la utilización de otro protocolo extra para el poder usar REST, a diferencia de otras arquitecturas. Dentro del desarrollo de aplicaciones o servicios web y especialmente en una API REST, deben cumplirse 3 niveles de calidad que se detallan en el modelo de madurez de Richardson [2] que se muestra a continuación:

2.2.2.1. Uso correcto de URLs

Las URLs permiten reconocer y localizar los recursos de forma única. Cuando nos referimos a REST, todo lo que contiene una página ya sea información en secciones se los denomina recursos, por tanto son los datos a los que se pueden acceder, borrar, modificar sin importar el formato que utilice. Hay varias reglas que se tienen que cumplir para establecer una URL a un recurso:

- Las URLs no deben hacer referencia a una acción, por lo tanto se tiene que evitar usar verbos en ellos.
- Deben referirse a un único recurso.
- Establecer una jerarquía lógica.
- El formato que se utilice debe ser independiente.

2.2.2.2. Uso correcto de HTTP

Cuando se está desarrollando una API REST hay que priorizar ciertos aspectos claves como los métodos HTTP y códigos de estado. Los métodos HTTP permiten manipular los recursos, en los cuales se encuentran los más usados como el GET que permite leer recursos, el POST que permite crear recursos, el PUT que ayuda a editar recursos y el DELETE que permite eliminar recursos. Durante el desarrollo se debe obviar la creación de herramientas y métodos propios y utilizar lo que ya está establecido y testeado, para evitar caer en errores de tipo código de error y estado.

2.2.2.3. Implementar Hypermedia

El propósito de Hypermedia es conectar las APIs con las aplicaciones clientes y agregar información extra al recurso sobre otros recursos relacionados con el. La Hypermedia es tan importante porque evita realizar mantenimientos concurrentes a los distintas URLs de los recursos en futuras modificaciones.

En la Figura 2.1 se puede observar un esquema basado en API REST.



Figura 2.1. Esquema basado en API REST [3].

2.2.3. Aplicaciones SPA(Single Page Applications)

El desarrollo de las aplicaciones web está evidenciando un cambio importante hacia las aplicaciones llamadas SPA (Single Page Applications), donde su objetivo principal es mejorar el performance y satisfacción de usuario, ya que se logra una disminución en los tiempos de carga gracias a las llamadas asincrónicas, evitando esperas y latencias entre vistas.

Comúnmente la lógica de negocio de las aplicaciones en la web se la construye en el back-end(lado del servidor) y un sistema de URLs que muestran las vistas. Tomemos como ejemplo, en una aplicación de comercio electrónico el detalle de un producto se habilitará en una URL única, mientras que la vista de registro estará en otra URL diferente.

Para el navegador o browser, cada una de las URLs son totalmente autónomas del resto, no obstante posean el mismo estilo o plantilla, esto lleva a que el browser deba volver a cargar y procesar desde cero. Lo anterior provoca que la interacción entre las vistas genere problemas de latencia en la Web [4].

2.2.3.1. Arquitectura SPA

En conclusión, una aplicación SPA es la interfaz web hecha casi en su totalidad en el navegador, no obstante tienen una base de codificación de HTML y CSS donde se puede utilizar frameworks bajo javascript que facilitan el trabajo del mismo. En la Tabla 1 se muestran las ventajas y desventajas.

Tabla 1. Ventajas y desventajas de aplicaciones SPA.

Ventajas	Desventajas
Interfaz de usuario más rápida.	Son aplicaciones más difíciles de construir.
Interactivas.	Necesita conocimientos sólidos de JavaScript.
Trabajan de forma offline.	Difícil elección por basarse en JavaScript.
Son perfectas para aplicaciones híbridas.	

Fuente: autor del proyecto.

Como todas las vistas se encuentran integradas en una SPA, al momento que se realiza la primera llamada al servidor esta devuelve todas la vistas de la interfaz, simplemente dejando en segundo plano los recursos más pesados, ya sean imágenes, videos, cantidades grandes de datos, entre otros. En la Figura 2.2 se podrá observar este proceso.

Esta clase de aplicaciones tienen un tiempo de respuesta mucho más veloz que las web tradicionales, ya que al cargar una página no se devuelve todo el HTML completo, sino exclusivamente la información de un contenido específico. Esto permite eliminar los cuellos de botella que se han convertido en un problema muy habitual.

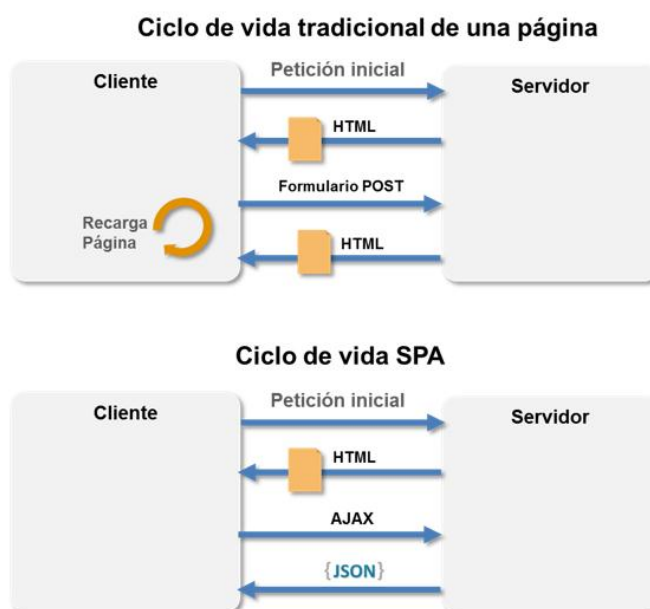


Figura 2.2. Ciclo de vida de una aplicación SPA y tradicional [5].

2.3. Lenguajes de Programación

2.3.1. MEAN

Es el conjunto de soluciones para el desarrollo que cubren las diferentes caras de una aplicación, ya sea el lado del servidor o el lado del cliente, con una característica en común, el uso exclusivo de JavaScript que es un lenguaje orientado a objetos, liviano, interpretado y muy versátil. Dentro de las tecnologías de conforman MEAN tenemos:

- Node.js
- Express.js
- Angular 2
- MongoDB

Las ventajas en la implementación de estas tecnologías son muchas, a continuación las más notables:

- El uso de JavaScript como lenguaje único en toda la solución (servidor y cliente).
- Una aplicación desarrollada bajo MEAN soporta el uso de la arquitectura MVC.

- Utiliza generalmente objetos JSON para la comunicación.

2.3.2. Node.js

Node.js [6] es un entorno multiplataforma que ejecuta código JavaScript del lado del servidor, pero su principal característica que es la gestión de peticiones HTTP por sí mismo es muy compleja, por lo que es necesaria la inclusión de Express.js que ayuda enormemente en dicho caso.

Node.js trabaja con un hilo de ejecución y gestiona sus tareas de forma asíncrona logrando que su ejecución sea mucho mas eficiente.

A continuación en la Figura 2.3 se representa el procesamiento asíncrono que realiza Node.js:

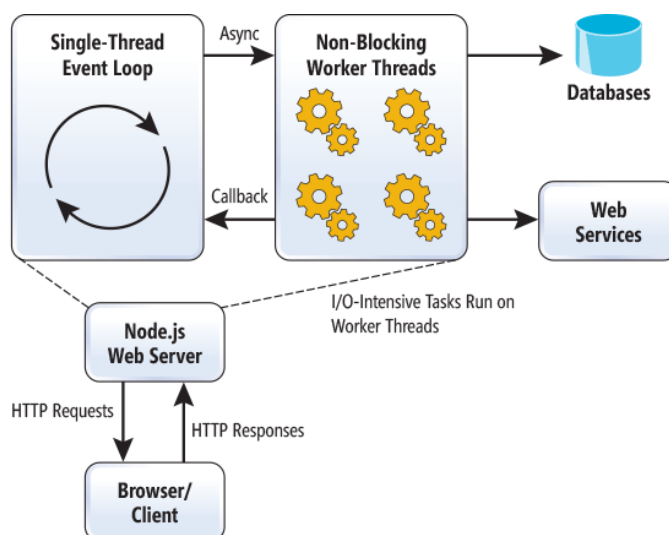


Figura 2.3. Procesamiento asíncrono de Node.js [7].

La mayoría de los desarrolladores concuerdan que las aplicaciones escritas en Node.js tienen un parecido común con PHP o Ruby, haciendo más fácil la migración o cambio a Node.js al contrario de los desarrollares que vienen de .NET o JAVA.

2.3.3. Express.js

El componente encargado de la gestión de las conexiones HTTP en Node.js es Express.js, lo que facilita un perfil de servidor web a Node.js. De acuerdo a la página oficial de Express.js lo denomina como un framework web flexible que otorga una serie de características para soluciones web y móviles.

Uno de los aspectos más interesantes que tiene Express.js es que es minimalista, que va de lo general a lo esencial y desecha lo que no usa o está de más, una característica perfecta para los desarrolladores con la idea de “menos es más”, esto no significa que no sea robusto o que no conste de características útiles, sino que la ideología de Express.js es brindar una brecha mínima entre el desarrollador y el servidor, lo cual permite plasmar mejores ideas.

Otra característica importante dentro de su ideología es su extensibilidad, donde se le pueden adicionar diferentes funcionalidades según la necesidad y/o cambiar lo que no se necesite. Esto genera alivio a diferencia de otros frameworks que traen muchas funcionalidades por defecto que en muchas de las ocasiones no son utilizadas, lo que produce pérdida de tiempo quitando funcionalidades no necesarias o reemplazando otras que no cumplan su objetivo.

Express.js posee varios comandos para poder gestionar las conexiones HTTP, a continuación se detallarán las más básicas que componen las operaciones elementales como escritura, lectura, actualización y eliminación, que forman una interfaz CRUD (create, read, update, delete).

- **Método Get:** se dispara cuando se recibe una solicitud de tipo GET, se utiliza en peticiones de consultas.
- **Método Post:** se dispara cuando se recibe una solicitud de tipo POST, se utiliza en peticiones de inserción o creación de registros.

- **Método Put:** se dispara cuando se recibe una solicitud de tipo PUT, se utiliza para peticiones de actualización de registros.
- **Método Delete:** se dispara cuando se recibe una solicitud de tipo DELETE, se utiliza para eliminar registros.

2.3.4. Angular

Angular [8] es un framework muy potente basado o concebido en JavaScript, que se utiliza para manejar el front-end(lado del cliente) de una aplicación. Angular utiliza la estructura MVC, que vio sus primeras implementaciones en los años 80, y desde entonces fue adoptada por las tecnologías más importantes. La Figura 2.4 nos enseña el esquema de esta estructura.

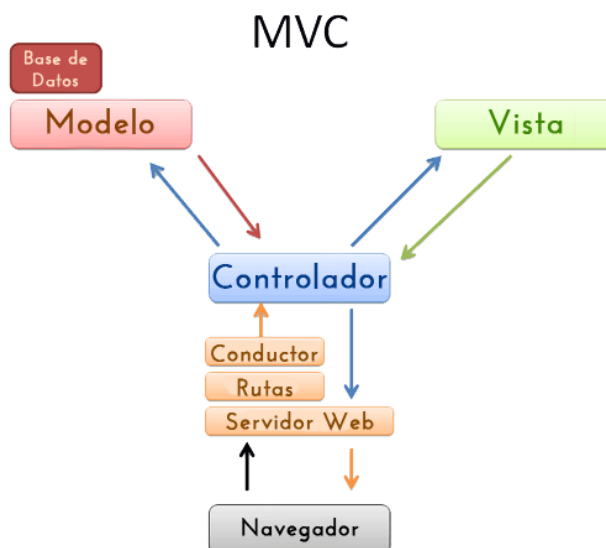


Figura 2.4. Esquema de arquitectura MVC [9].

Conocer cómo implementar la estructura MVC es muy ventajoso, ya que brinda una idea de dónde ubicar cada elemento, lo que ayuda a las buenas prácticas de desarrollo. Otra razón para utilizar esta arquitectura es que ayuda a comprender el código a terceras personas que quieran conocer las funcionalidades de cada elemento.

Angular.js posee un gestor de paquetes denominado BOWER, que funciona de forma similar al gestor de paquetes NPM de Node.js, donde consta un archivo llamado bower.json en el que se incluyen las dependencias necesarias del proyecto, que mediante comandos se pueden agregar nuevos módulos o actualizar los existentes. El flujo de datos en 2 vías o direccional es otra característica interesante de Angular.js, que ayuda a la aplicación mantenerse sincronizada tanto la vista con los datos. En la Figura 2.5 se muestra el flujo de datos bidireccional.

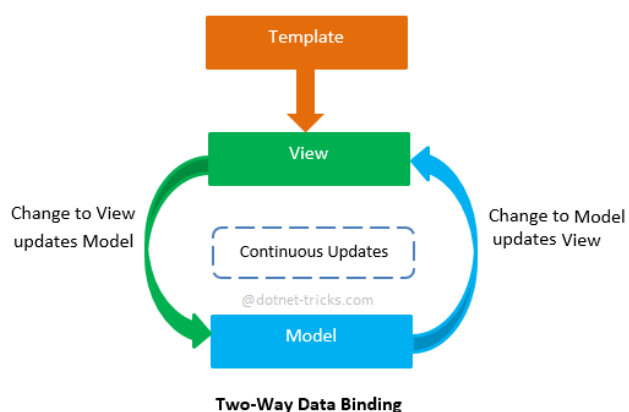


Figura 2.5. Funcionamiento del flujo de datos bidireccional de Angular [10].

2.3.5. MongoDB

MongoDB [11] es una base de datos NoSQL que maneja un nuevo concepto en relación a las tradicionales, Nació por la necesidad de quebrar el esquema fijo tradicional de tablas que brinda SQL e implantar movilidad lo que permita tener registros de información no estructurada.

Esto es un avance muy significativo, debido a que la gran cantidad de datos que se almacenan desde hace varios años hasta la actualidad, han tenido un crecimiento exponencial. Hoy en día se generan cerca de 2.3 exabytes de datos, de los cuales alrededor del 80% de los mismos son estructurados provenientes de mensajería instantánea, redes sociales entre otros [12].

El concepto base de datos no estructuradas abre las puertas a nuevas oportunidades de negocio mediante la explotación de los datos, ya que una vez los tengamos almacenados, con el uso de técnicas de minería de datos o Big Data podríamos extraer información precisa que nos ayuda a la predicción de datos.

A continuación se detallan las características principales de MongoDB:

- Puede actualizarse en segundo plano sin dejar de dar servicio.
- No produce cuellos de botella.
- Utiliza el estándar web JSON para todas sus operaciones de información.
- Permite procesar información de funciones JavaScript mediante Map-Reduce.
- MongoDB Management Service ayuda a monitorear las bases de datos.

En MongoDB la sintaxis cambia en relación a SQL, ya que no necesita espacios entre palabras, paréntesis, comas o palabras entrecomillas, utiliza la estructura operador – operando. En la Tabla 2 se muestra la equivalencia de sentencias entre SQL y MongoDB en las operaciones CRUD(insertar, consultar, actualizar y eliminar).

Tabla 2. Equivalencias entre SQL y MongoDB

Sentencias SQL	Sentencias MongoDB
Consulta	
SELECT * FROM users	db.users.find()
Insertar	
INSERT INTO users (user_id, age, status) VALUES ("bcscw",30,"B")	db.users.insert({ user_id: "bcscw", age: 30, status: "B" })
Actualizar	
UPDATE users SET status = "C" WHERE age > 25	db.users.update({ age: { \$gt: 25 } }, { \$set: { status: "C" } }, { multi: true })
Eliminar	
DELETE FROM users WHERE status = "D"	db.users.remove({ status: "D" })

Fuente: autor del proyecto.

2.4. Definición de Requisitos

2.4.1. Levantamiento de requerimientos usando JAD

Una manera de diseñar sistemas informáticos donde colaboran grupos de discusión que participan en el proyecto es mediante el uso de JAD [13], que fue creado para que los usuarios y desarrolladores con diferentes opiniones y orígenes, puedan estar en un ambiente creativo, colaborativo y productivo. La mejor forma de lograr requisitos de calidad son las reuniones.

JAD utiliza una lluvia de ideas de los distintos grupos del proyecto para determinar de forma ágil los requerimientos de un sistema informático. Para lo cual se forma un grupo con los involucrados más significativos ya sean estos administradores, usuarios y analistas. Luego se realizan reuniones con el objetivo de debatir ideas, analizar necesidades y lograr soluciones alternativas.

JAD es utilizado para elaborar distintos análisis como: análisis de viabilidad, análisis de costos, análisis de riesgos, análisis de implementación, análisis de factibilidad, entre otros. Para el desarrollo de las reuniones JAD hay que tener como mínimo tres grupos que son:

2.4.1.1. Patrocinador del Proyecto

Es el encargado de realizar el presupuesto del proyecto y el dueño del mismo. Consta con el nivel más alto en la organización, de tal manera que puede proporcionar recursos y tomar decisiones. Las responsabilidades del patrocinador son:

- Asegurar los clientes idóneos.
- Asegurar personal técnico.
- Definir la funcionalidad y alcance.

- Evaluar resultados del sistema.

2.4.1.2. Líder del Proyecto

Deben estar muy comprometidos e involucrados con el proyecto, tener conocimientos de sistemas de información y del área comercial. El líder necesita ser objetivo y no permitir que otro miembro trate de dominar el grupo. Sus responsabilidades son:

- Verificar que los roles estén ocupados.
- Planificar las reuniones.
- Asegurar que las tareas se cumplan según el cronograma establecido.

2.4.1.3. Clientes

Son los dueños de los procesos, conocen como funcionan y su uso. Los clientes del sistema ayudarán a entender las tareas manejadas por el sistema. Sus responsabilidades son:

- Definir la información que maneja el proceso.
- Examinar las dificultades actuales.

2.4.2. Casos de Uso

Los casos de uso [14] permiten realizar una descripción de cada uno de los pasos o actividades que se ejecutan a lo largo de un proceso. A las entidades que participan se las llaman actores. La ingeniería de software define a los caso de uso como una secuencia de intercambios que mantendrá un sistema y sus actores como resultado de un evento. Podríamos decir de estos diagramas se los utiliza para describir la comunicación y comportamiento de un sistema por medio de la interacción que exista entre los usuarios y otros sistemas.

Por consiguiente, los casos de uso ayudan a definir los requisitos funcionales de la solución, en otros términos, constituyen las funciones de un sistema. En la Figura 2.6 se muestran componentes de un diagrama de casos de uso.

2.4.2.1. Actor

Los actores personifican un tipo de usuario, que representa un rol que se comunica con el sistema, disparando eventos sobre el mismo con el objetivo de conseguir una respuesta dentro de un proceso.

2.4.2.2. Caso de uso

Representa un proceso o tarea que se realiza luego de la creación de un evento ejecutado por un actor o un caso de uso. Los casos de uso deben ser detallados y fundamentados mediante una descripción textual e indique su flujo.

2.4.2.3. Asociaciones

Los casos de uso constan de tres principales relaciones, soportadas por el estándar UML. La relación de inclusión es utilizada cuando dos o más casos tiene un comportamiento común. La relación de extensión especifica varios cambios de un mismo caso de uso. La relación de generalización no es más que la herencia de actores o casos de uso.

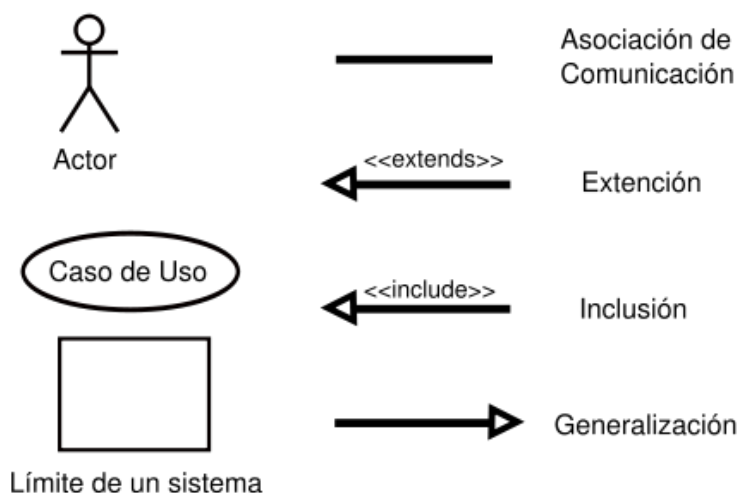


Figura 2.6. Componentes de un diagrama de casos de uso [15].

2.5. Metodologías

Entre el desarrollo del software, la alta necesidad de que los proyectos tengan el éxito esperado y alcanzar una solución que agregue valor a los clientes, crean enormes cambios al momento de adoptar una metodología que logre cumplir sus objetivos, debido que, unas se adaptan mejor en relación a la otra, dependiendo mucho el tipo de proyecto que nos pueda brindar mejores ventajas. El éxito o fracaso de un producto final está muy ligado a la metodología que se escoja, sea esta tradicional o ágil, donde el equipo saque el máximo de su capacidad, magnifiquen la calidad del producto con los recursos y tiempos determinados.

En base a la experiencia personal y la tabla comparativa de metodologías que se muestra en la Tabla 3, se ha seleccionado una metodología ágil para el desarrollo del producto.

Tabla 3. Diferencias entre metodologías tradicionales y ágiles.

Metodologías Tradicionales	Metodologías Ágiles
Basadas en estándares acogidas por el ambiente de desarrollo	Basadas en producción de código
Mucha resistencia a cambios	Ideales para cambios durante el proyecto
El cliente se comunica con el equipo por medio de reuniones	El cliente se encuentra en el equipo
Mayores recursos	Pocos recursos
Grupos de trabajo grandes y muchos roles	Grupos de trabajo pequeños menores a 10 y pocos roles
Arquitectura de software es importante	Flexible en la arquitectura de software

Fuente: autor del proyecto.

2.5.1. SCRUM

La metodología ágil Scrum es un proceso liviano que ayuda a controlar el desarrollo de software. Se ejecuta de forma iterativa, donde cada ciclo termina con una parte del producto ejecutable que agrega nuevas funciones. Los ciclos o iteraciones tienen un tiempo entre 2 y 4 semanas. Scrum se enfoca en darle valor al negocio, extendiendo al máximo la utilidad de lo que se elabora, en especial a la flexibilidad a los cambios en los requerimientos. Esto logra que efectivamente se concreten las necesidades y crezca la satisfacción del cliente [16]

Scrum está basado en el concepto de inspección continua, innovación y adaptación, que forman reglas pequeñas y simples. Esto permite que el cliente vea crecer su producto en cada ciclo. En la siguiente Figura 2.7 se muestra el ciclo completo de esta metodología.

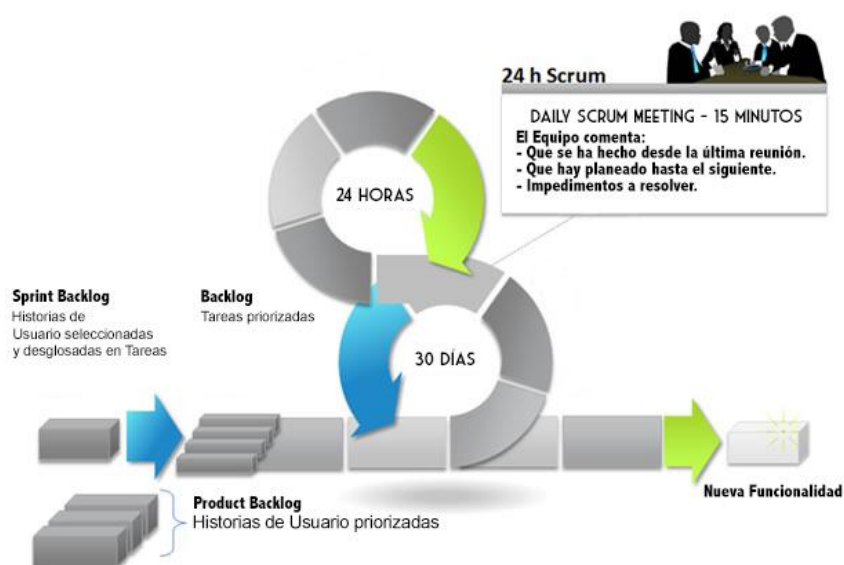


Figura 2.7. Ciclo de vida SCRUM [17].

CAPÍTULO 3

REQUERIMIENTOS Y METODOLOGÍA

3.1. Introducción

Para la metodología de desarrollo se hizo el levantamiento de información utilizando las técnica JAD, se conformó el equipo Scrum en los cuales se definieron los actores, tales como equipo de desarrollo y el dueño del producto con los cuales se desarrollaron reuniones y se concretaron las iteraciones. Además, se realizó el Scrum diario entre el equipo y las revisiones de iteración, lo que se detalla más adelante.

3.2. Levantamiento de Requerimientos

En el levantamiento de los requerimientos iniciales se realizaron reuniones con el dueño del producto y el patrocinador, donde se pudieron exponer ideas y lograr definir las. También se formó el grupo de desarrolladores donde los mismos hicieron sugerencias y propusieron ideas.

Luego de varias de reuniones se procedió a realizar un informe acerca de los temas discutidos durante este proceso, lo que se detalla en el anexo nº1.

La Unidad Educativa Redemptio desea administrar la aplicación web, para eso contará con una sección preferencial donde podrá gestionar algunas características del sistema como: Egresados, Empleadores, Encuesta de seguimientos a graduados y Reportes del sistema.

En base a todo lo realizado se han definido las siguientes funcionalidades que se desarrollaron en la API de la solución. A continuación se detallan:

- Creación, Edición, Borrado y Consulta de Administradores.
- Creación, Edición, Borrado y Consulta de Egresados.
- Creación, Edición, Borrado y Consulta de Empleadores.
- Responder, Reportes y Consulta de Encuestas.

- Creación, Edición, Borrado, Postulación, Aceptación y Consulta de ofertas de trabajo.

La sección donde tendrán acceso los Egresados son:

- Autenticación a la aplicación.
- Registro en la aplicación.
- Recuperación de contraseña.
- Visualización pantalla de inicio.
- Resolución de encuestas de seguimientos a graduados.
- Visualización y aplicación a ofertas de trabajo.

La sección donde tendrán acceso los Empleadores son:

- Autenticación a la aplicación.
- Registro en la aplicación.
- Visualización pantalla de inicio.
- Creación y postulación de ofertas de trabajo.

3.3. Definición de Casos de Uso

Para el modelado de los casos de uso se han definido los actores que interactúan con el sistema: Egresados, Empleadores, Administradores.

3.3.1. Registro de Usuarios



Figura 3.1. Caso de uso registro de usuarios.
Fuente: autor del proyecto.

Tabla 4. CU-01 Registro de usuario del sistema.

CU-01	Registro de Usuarios del Sistema
Actores	Administrador
Propósito	Registrar información de los usuarios del sistema para poder otorgar permisos de acuerdo a su rol.
Tipo	Primario esencial.
Flujo típico de eventos	<ol style="list-style-type: none"> 1. Administrador inicia sesión. 2. Observará el formulario de ingreso nuevo usuario: egresado o empleador. 3. Valida los datos proporcionados. 4. Se muestra un mensaje de registro realizado con éxito.
Flujo alternativo	<ol style="list-style-type: none"> 1. Administrador ingresa su usuario y contraseña de forma incorrecta, se termina el caso. 2. El formulario no encontrado, se termina el caso.

Fuente: autor del proyecto.

Tabla 5. CU-02 Validar información usuario del sistema.

CU-02	Validar Información Usuario del Sistema
Actores	Administrador
Propósito	Validar la información registrada.
Tipo	Primario esencial.
Flujo típico de eventos	La cédula ingresada es válida.
Flujo alternativo	La cédula ingresada no es válida, termina el caso.

Fuente: autor del proyecto.

3.3.2. Ingreso al Sistema

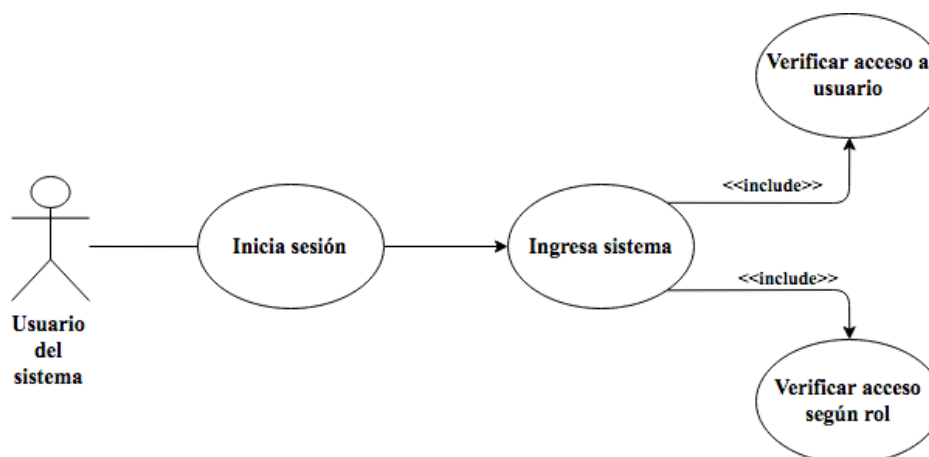


Figura 3.2. Caso de uso ingreso al sistema.
Fuente: autor del proyecto.

Tabla 6. CU-03 Ingreso al sistema.

CU-03	Ingreso al Sistema
Actores	Egresados, Empleadores y Administradores
Propósito	Ingresar al sistema con los permisos otorgados según el rol.
Tipo	Primario esencial.
Flujo típico de eventos	<ol style="list-style-type: none"> 1. El usuario inicia su sesión. 2. El sistema valida la información según su rol. 3. Se muestra la información.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa su usuario y contraseña incorrectos, termina el caso. 2. Formulario no se encuentra, termina el caso de uso.

Fuente: autor del proyecto.

Tabla 7. CU-04 Verificar Acceso al Sistema.

CU-04	Verificar Acceso al Sistema
Actores	Egresados, Empleadores y Administradores.
Propósito	Validar la información de usuario y contraseña.
Tipo	Primario esencial.
Visión General	Se ingresan datos al sistema como: usuario y contraseña, para su autenticación.

Fuente: autor del proyecto.

Tabla 8.CU-05 Verificar acceso según rol.

CU-05	Verificar Acceso según Rol
Actores	Egresados, Empleadores y Administradores.
Propósito	Permitir ingresar al sistema y otorgar permisos según su rol.
Tipo	Primario esencial.
Visión General	Se ingresan datos al sistema como: usuario y contraseña, que le permita tener acceso.

Fuente: autor del proyecto.

3.3.3. Información de los Egresados

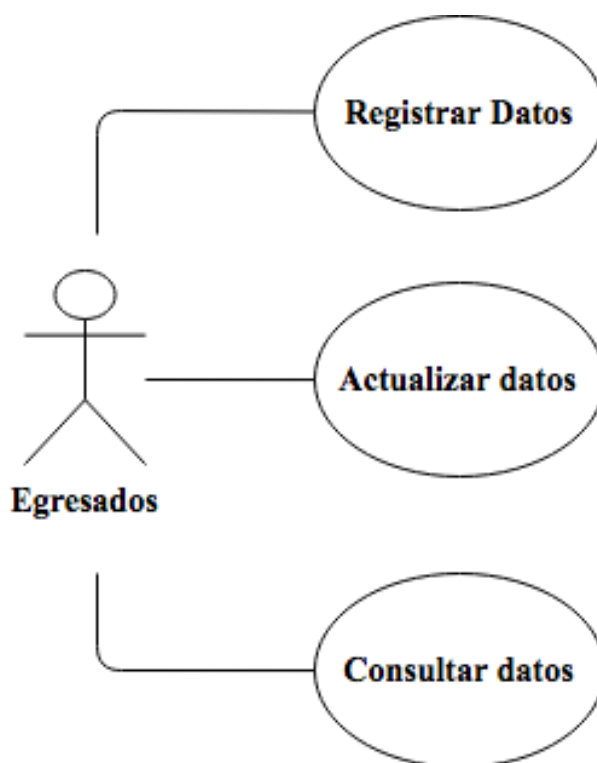


Figura 3.3. Caso de uso información de los egresados.

Fuente: autor del proyecto.

Tabla 9. CU-06 Registrar datos egresados.

CU-06	Registrar Datos Egresado
Actores	Egresados y Administradores
Propósito	Permitir a los egresados y administradores ingresar información de los egresados.
Tipo	Primario esencial
Flujo típico de eventos	<ol style="list-style-type: none"> 1. Usuario ingresa al sistema. 2. Se muestra el formulario con la respectiva información, según el rol. 3. Se muestran las alternativas de registro, edición y consulta de datos. 4. Se registra los datos con éxito.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa sus credenciales incorrectas, termina el caso. 2. Formulario no se encuentra según su rol, termina el caso. 3. Formulario no muestra las opciones de registrar, editar y consultar los datos, termina el caso. 4. La información registrada no puede ser validada, termina el caso.

Fuente: autor del proyecto.

Tabla 10. CU-07 Actualizar datos egresado.

CU-07	Actualizar Datos Egresado
Actores	Egresados
Propósito	Permitir a los egresados actualizar su información.
Tipo	Primario esencial.
Flujo típico de eventos	<ol style="list-style-type: none"> 1. Usuario ingresa al sistema. 2. Se muestra el formulario con la respectiva información, según el rol. 3. Se muestran las alternativas de registro, edición y consulta de datos. 4. Se actualizan los datos con éxito.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa sus credenciales incorrectas, termina el caso. 2. Formulario no se encuentra según su rol, termina el caso. 3. Formulario no muestra las opciones de registrar, editar y consultar los datos, termina el caso. 4. La información registrada no puede ser validada, termina el caso.

Fuente: autor del proyecto.

Tabla 11.CU-08 Consultar datos egresado.

CU-08	Consultar Datos Egresado
Actores	Egresados
Propósito	Permitir a los egresados actualizar su información.
Tipo	Primario esencial.
Flujo típico de eventos	<ol style="list-style-type: none"> 1. Usuario ingresa al sistema. 2. Se muestra el formulario con la respectiva información, según el rol. 3. Se muestran las alternativas de registro, edición y consulta de datos. 4. Se muestra información solicitada.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa sus credenciales incorrectas, termina el caso. 2. Formulario no se encuentra según su rol, termina el caso. 3. Formulario no muestra las opciones de registrar, editar y consultar los datos, termina el caso. 4. La información registrada no puede ser validada, termina el caso.

Fuente: autor del proyecto.

3.3.4. Información de los Empleadores

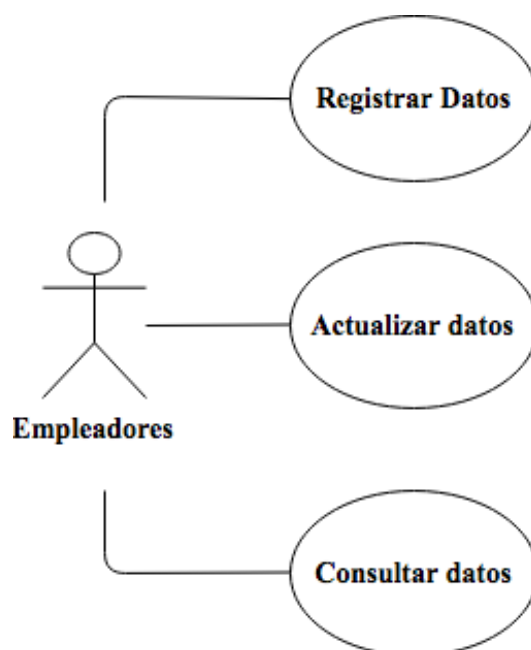


Figura 3.4.Caso de uso información de los empleadores
Fuente: autor del proyecto.

Tabla 12.CU-09 Registrar datos empleador

CU-09	Registrar Datos Empleador
Actores	Empleadores y Administradores
Propósito	Permitir a los empleadores actualizar su información.
Tipo	Primario esencial.
Flujo típico de eventos	<ol style="list-style-type: none"> 1. El empleador ingresa al sistema. 2. Se muestra el formulario con la respectiva información, según el rol. 3. Se muestran las alternativas de registro, edición y consulta de datos. 4. Se registran los datos con éxito.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa sus credenciales incorrectas, termina el caso. 2. Formulario no se encuentra según su rol, termina el caso. 3. Formulario no muestra las opciones de registrar, editar y consultar los datos, termina el caso. 4. La información registrada no puede ser validada, termina el caso.

Fuente: autor del proyecto.

Tabla 13.CU-10 Actualizar datos empleador.

CU-10	Actualizar Datos Empleador
Actores	Empleadores
Propósito	Permitir a los empleadores actualizar su información.
Tipo	Primario esencial
Flujo típico de eventos	<ol style="list-style-type: none"> 1. El empleador ingresa al sistema. 2. Se muestra el formulario con la respectiva información, según el rol. 3. Se muestran las alternativas de registro, edición y consulta de datos. 4. Se actualizan los datos con éxito.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa sus credenciales incorrectas, termina el caso. 2. Formulario no se encuentra según su rol, termina el caso. 3. Formulario no muestra las opciones de registrar, editar y consultar los datos, termina el caso. 4. La información registrada no puede ser validada, termina el caso.

Fuente: autor del proyecto.

Tabla 14.CU-11 Consultar datos empleador.

CU-11	Consultar Datos Empleador
Actores	Empleadores
Propósito	Permitir a los empleadores actualizar su información.
Tipo	Primario esencial
Flujo típico de eventos	<ol style="list-style-type: none"> 1. El empleador ingresa al sistema. 2. Se muestra el formulario con la respectiva información, según el rol. 3. Se muestran las alternativas de registro, edición y consulta de datos. 4. Se muestran los datos solicitados.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa sus credenciales incorrectas, termina el caso. 2. Formulario no se encuentra según su rol, termina el caso. 3. Formulario no muestra las opciones de registrar, editar y consultar los datos, termina el caso. 4. La información registrada no puede ser validada, termina el caso.

Fuente: autor del proyecto.

3.3.5. Encuestas Administrador

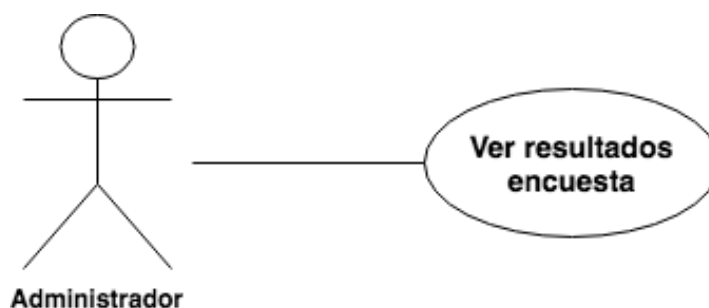


Figura 3.5.Caso de uso encuestas administrador.

Fuente: autor del proyecto.

Tabla 15. CU-14 Ver resultados de la encuesta.

CU-14	Ver Resultados de la Encuesta
Actores	Administradores
Propósito	Permitir visualizar la encuesta que van a realizar los egresados.
Tipo	Primario esencial.
Flujo típico de eventos	1. El administrador escoge la información que desea revisar.
Flujo alternativo	1. Botón de estadísticas no existe. 2. Vista de reportes no se encuentra, termina el caso.

Fuente: autor del proyecto.

3.3.6. Encuesta Egresados

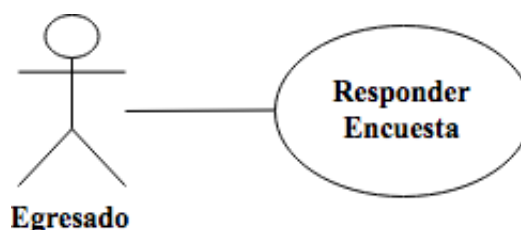


Figura 3.6. Caso de uso encuesta egresados.

Fuente: autor del proyecto.

Tabla 16. CU-15 Responder encuesta.

CU-15	Responder Encuesta
Actores	Egresados
Propósito	Permitir al egresado resolver una encuesta elaborada por el administrador.
Tipo	Primario esencial.
Flujo típico de eventos	1. El administrador responde todas las preguntas de la encuesta.
Flujo alternativo	1. Vista de encuesta no se encuentra, termina el caso.

Fuente: autor del proyecto.

3.3.7. Ofertas Egresados

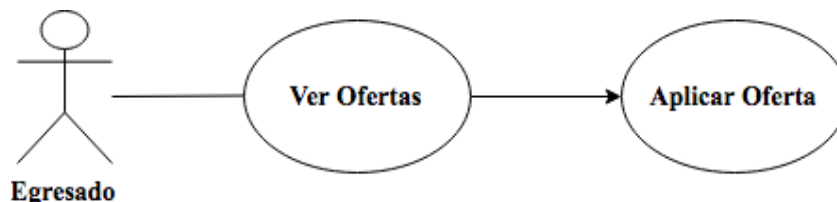


Figura 3.7. Caso de uso ofertas egresados.

Fuente: autor del proyecto.

Tabla 17. CU-16 Ver ofertas.

CU-16	Ver Ofertas
Actores	Egresados
Propósito	Permitir al egresado ver las ofertas laborales ingresadas por los empleadores.
Tipo	Primario esencial.
Flujo típico de eventos	1. El egresado podrá ver las ofertas laborales.
Flujo alternativo	2. Vista de ofertas laborales no se encuentra, termina el caso.

Fuente: autor del proyecto.

Tabla 18. CU-17 Aplicar Ofertas.

CU-17	Aplicar Ofertas
Actores	Egresados
Propósito	Permitir al egresado ver las ofertas laborales ingresadas por los empleadores y aplicar a ellas.
Tipo	Primario esencial.
Flujo típico de eventos	1. El egresado previamente encuentra una oferta. 2. Aplicar la oferta con éxito.
Flujo alternativo	1. Vista de ofertas laborales no se encuentra, termina el caso.

Fuente: autor del proyecto.

3.3.8. Ofertas Empleadores

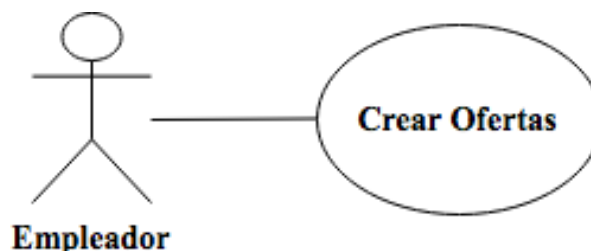


Figura 3.8. Caso de uso ofertas empleadores.
Fuente: autor del proyecto.

Tabla 19. CU-18 Crear ofertas.

CU-18	Crear Ofertas
Actores	Empleadores
Propósito	Permitir al empleador registrar ofertas laborales para que los egresados puedan aplicar en ellas.
Tipo	Primario esencial.
Flujo típico de eventos	<ol style="list-style-type: none"> 1. El empleador llena el formulario con todos los requisitos que el está solicitando. 2. Se publica la oferta con éxito.
Flujo alternativo	<ol style="list-style-type: none"> 1. Formulario de ofertas laborales no se encuentra, termina el caso.

Fuente: autor del proyecto.

3.4. Implementando Scrum

3.4.1. Selección del Equipo Scrum

El equipo de Scrum se conformará de la siguiente manera:

- Dueño del Producto: Dra. Luisa Ponce.
- Equipo de Desarrollo: Ing. Andrés Granja Moreira.
- Scrum Master: Lcdo. Fulco Pincay.

Cada uno de los miembros del equipo Scrum desempeñarán sus funciones designadas y aportarán al desarrollo del proyecto.

3.4.2. Planeación de la Iteración (Sprint Planning)

En este proceso se podrá realizar una visión muy aproximada y definir un análisis del negocio, estimaciones y alcance. El tiempo estimado 2 días. En esta reunión participan el dueño del producto (Product Owner) y el equipo de desarrollo (Scrum Team), y se la denomina reunión de planeación de la iteración (Sprint Planning Meeting).

El dueño del producto presentará un detalle de los requerimientos prioritarios del proyecto al equipo de desarrollo, quienes resolverán sus dudas y se comprometen a terminar la iteración con éxito, de forma que logren entregarla si el cliente lo requiere.

El tiempo estimado es 4 horas máximo.

3.4.3. Ejecución de la Iteración (Sprint)

El proceso se elaborará en el lapso de 4 semanas, donde cada sprint debe entregar un resultado completo, en base a la lista lista de requerimientos o product backlog creada, donde se definieron tareas con una lista de responsabilidades o sprint backlog. Cuando la iteración presente obstáculos se deben tratar de eliminarlos, lo que permitirá avanzar a la siguiente iteración.

3.4.4. Reunión Diaria de Sincronización del Equipo (Scrum Daily Meeting)

Este proceso se lo elabora diariamente, donde el objetivo es proporcionar información y proponer cambios que permitan incrementar la producción.

3.4.5. Demostración de requisitos completados (Sprint Demonstration)

Si las tareas se han completado con éxito, tendremos una versión estable con las funcionalidades establecidas en el product backlog. A esta altura ya ha transcurrido aproximadamente un mes desde el inicio del proyecto.

3.4.6. Retrospectiva (Sprint Retrospective)

Una práctica común para mantener una alta producción, es que el scrum master se ocupe de identificar obstáculos e ir eliminándolos, en el caso que el equipo de desarrollo no pueda resolverlos. Tiempo máximo: 4 horas.

3.4.7. Re-planificación del proyecto

Durante una iteración, el dueño del producto podrá realizar cambios de acuerdo a la prioridad del proyecto, añadiendo o modificando requisitos. En esta parte se puntualizan los nuevos cambios pero no se podrán realizar durante el sprint. Tiempo de duración: 2 horas.

3.5. Historias de Usuarios

En esta parte del proyecto y luego de haber realizado el levantamiento de requerimientos en conjunto con el cliente, procedemos a realizar una descripción clara y simple de las características que la solución debe presentar. Para esto se han utilizado historias de usuarios tal y como lo recomienda la metodología SCRUM. En la Tabla 20 se muestra las historias y las tareas que se tuvieron que realizar.

Tabla 20. Historias de usuarios.

#	Historia de Usuario	Tareas
1	Autenticación al Sistema	- Diseño e implementación de la BBDD y desarrollo del módulo de egresados.
2	Registro de Usuarios al Sistema	
3	Edición de Usuarios del Sistema	
4	Actualización Datos Egresados	
5	Elaboración de Encuesta	Desarrollo complemento del módulo egresados.
6	Aplicación de Encuesta	
7	Actualización Datos Empleadores	Desarrollo del módulo empleadores y aplicación de ofertas de egresados.
8	Creación de Ofertas Laborales	
9	Aplicación de Ofertas Laborales	
10	Publicación de Ofertas Laborales	
11	Visualización Datos de Encuestas	Creación de gráficos estadísticos.
12	Cambio de Contraseñas	

Fuente: autor del proyecto.

3.5.1. Detalle de las Historias de Usuarios

Tabla 21. HU-01 Autenticación al sistema

HU-01	
Número: 1	Usuario: Egresados/Empleadores/Administradores
Nombre: Autenticación al Sistema	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario registrado en el sistema quiero ingresar a la aplicación, ingresando mi correo electrónico y contraseña para tener acceso a las funcionalidades de acuerdo a mi rol de usuario.	
Observaciones: Existen tres tipos de usuarios: Egresados, Empleadores y Administradores, con diferentes roles y acceso a las distintas funcionalidades.	

Fuente: autor del proyecto.

Tabla 22. HU-02 Registro de usuarios del sistema.

HU-02	
Número: 2	Usuario: Administradores
Nombre: Registro de Usuarios del Sistema	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como Administrador del sistema quiero registrar usuarios nuevos en la aplicación, para que los usuarios nuevos puedan tener acceso a las funcionalidades de acuerdo a mi rol de usuario.	
Observaciones: Las funciones son asignadas de acuerdo al rol de usuario ya sea Egresados o Empleadores.	

Fuente: autor del proyecto.

Tabla 23. HU-03 Edición de usuarios del sistema

HU-03	
Número: 3	Usuario: Administradores
Nombre: Edición de Usuarios del Sistema	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como Administrador del sistema quiero editar usuarios registrados en la aplicación,	
Observaciones: El Administrador no puede eliminar usuarios del sistema.	

Fuente: autor del proyecto.

Tabla 24. HU-04 Actualización datos de egresados

HU-04	
Número: 4	Usuario: Egresados
Nombre: Actualización Datos de Egresados	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario Egresado del sistema quiero actualizar los datos correspondientes.	
Observaciones: El usuario Administrador crea un usuario egresado para poder acceder a la aplicación.	

Fuente: autor del proyecto.

Tabla 25. HU-05 Elaboración de encuesta.

HU-05	
Número: 5	Usuario: Administradores/Programador
Nombre: Elaboración de Encuesta	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Medio
Descripción: Como Administrador del sistema quiero elaborar una encuesta nueva, para poder iniciar un proceso nuevo, sin eliminar la anterior.	
Observaciones: La elaboración de la encuesta se la realiza con una herramienta externa que genera un documento JSON, que luego podemos ingresar en la base datos. Este paso lo realiza el programador del sistema.	

Fuente: autor del proyecto.

Tabla 26. HU-06 Aplicación de encuesta.

HU-06	
Número: 6	Usuario: Egresados
Nombre: Aplicación de Encuesta	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Medio
Descripción: Como usuario Egresado quiero resolver la encuesta para poder proporcionar datos importantes al proceso de seguimiento a egresados.	
Observaciones: El usuario Egresado debe haber actualizado sus datos primero antes de aplicar la encuesta.	

Fuente: autor del proyecto.

Tabla 27. HU-07 Actualización datos de empleadores

HU-07	
Número: 7	Usuario: Empleadores
Nombre: Actualización Datos de Empleadores	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario Empleador del sistema quiero actualizar mis datos	
Observaciones: El usuario Administrador crea un usuario empleador para poder acceder a la aplicación.	

Fuente: autor del proyecto.

Tabla 28. HU-08 Creación de ofertas laborales.

HU-08	
Número: 8	Usuario: Empleadores
Nombre: Creación de Ofertas Laborales	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario Empleador del sistema quiero crear ofertas laborales de acuerdo a mis requerimientos, para que los egresados puedan aplicar a las mismas.	
Observaciones: El usuario Empleador debe haber actualizado sus datos correspondientes para poder crear ofertas laborales.	

Fuente: autor del proyecto.

Tabla 29. HU-09 Aplicación de ofertas laborales.

HU-09	
Número: 9	Usuario: Egresados
Nombre: Aplicación de Ofertas Laborales	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario Egresado quiero aplicar a las ofertas laborales creadas por usuarios Empleador.	
Observaciones: El usuario Empleador debe publicar la oferta después de haberla creado respectivamente.	

Fuente: autor del proyecto.

Tabla 30. HU-10 Publicación de ofertas laborales.

HU-10	
Número: 10	Usuario: Empleador
Nombre: Publicación de Ofertas Laborales	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario Empleador quiero publicar las ofertas laborales, luego de haberlas creado para que los usuarios Egresado puedan aplicar a las mismas.	
Observaciones: Las ofertas laborales no pueden ser eliminadas por el usuario Empleador.	

Fuente: autor del proyecto.

Tabla 31. HU-11 Visualización datos de encuestas

HU-11	
Número: 11	Usuario: Administrador
Nombre: Visualización datos de encuestas	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: Como usuario Administrador quiero ver los indicadores de cada pregunta, por periodo, por promoción, por género, de tal manera que me permita tomar decisiones y tomar correctivos en mis procesos educativos y/o administrativos.	
Observaciones:	

Fuente: autor del proyecto.

Tabla 32. HU-12 Cambio de contraseña.

HU-12	
Número: 12	Usuario: Administrador/ Egresado/ Empleador
Nombre: Cambio de contraseña	
Prioridad en Negocio: Medio	Riesgo en Desarrollo: Baja
Descripción: Como usuario Administrador, Egresado o Empleador quiero cambiar la clave que se proporciona en el momento de la creación del mismo.	
Observaciones:	

Fuente: autor del proyecto.

3.6. Iteraciones

3.6.1. Iteración 01

Se comenzó con la construcción de la base de datos y el desarrollo, el cual fue sustentado con los requerimientos del cliente, quien informó sobre los datos importantes en el proceso. En esta iteración se pretende hacer la primera entrega funcional sobre los egresados, quienes podrán actualizar sus datos y los administradores crear usuarios del sistema. En la Tabla 33 se visualiza el desarrollo de las respectivas historias de usuarios donde el esfuerzo total fue de 2 semanas.

Tabla 33. Detalle de iteración 01

#	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración
1	Autenticación al Sistema	Alta	Baja	1	1
2	Registro de Usuarios al Sistema	Alta	Baja		1
3	Edición de Usuarios del Sistema	Alta	Baja	1	1
4	Actualización Datos Egresados	Alta	Baja		1

Fuente: autor del proyecto.

3.6.2. Iteración 02

En esta iteración se pretende hacer la segunda entrega funcional relacionada con el instrumento de evaluación, donde los egresados podrán llenar la encuesta. Previo a esto la encuesta es diseñada en una herramienta externa e ingresada en la base de datos para que pueda ser visualizada desde la aplicación. En la Tabla 34 se visualiza el desarrollo de las respectivas historias de usuarios donde el esfuerzo total fue 2 semanas.

Tabla 34. Detalle de iteración 02

#	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración
5	Elaboración de Encuesta	Alta	Medio	1	2
6	Aplicación de Encuesta	Alta	Medio	1	2

Fuente: autor del proyecto.

3.6.3. Iteración 03

En esta iteración se pretende hacer la tercera entrega funcional relacionada al módulo empleador y un complemento a la iteración 01 que tiene relación con los egresados, donde podrán aplicar a las ofertas laborales previamente creadas por los empleadores. En la Tabla 35 se visualiza el desarrollo de las respectivas historias de usuarios donde el esfuerzo total fue de 4 semanas.

Tabla 35. Detalle de iteración 03

#	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración
7	Actualización de Empleadores	Alta	Baja	2	3
8	Creación de Ofertas Laborales	Alta	Baja		3
9	Aplicación de Ofertas Laborales	Alta	Baja	2	3
10	Publicación de Ofertas Laborales	Alta	Baja		3

Fuente: autor del proyecto.

3.6.4. Iteración 04

En esta iteración se pretende hacer la cuarta y última entrega funcional relacionada al módulo de reportes y visualización datos de encuestas. Adicional a eso, cada tipo de usuario ya sea administrador, egresado o empleador pueden cambiar su clave de acceso respectivamente. En la Tabla 36 se visualiza el desarrollo de las distintas historias de usuarios donde el esfuerzo total fue de 3 semanas.

Tabla 36. Detalle de iteración 04

#	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración
11	Visualización Datos de Encuesta	Alta	Baja	2	4
12	Cambio de Contraseña	Medio	Baja	1	4

Fuente: autor del proyecto.

3.6.5. Cronograma de la Tareas

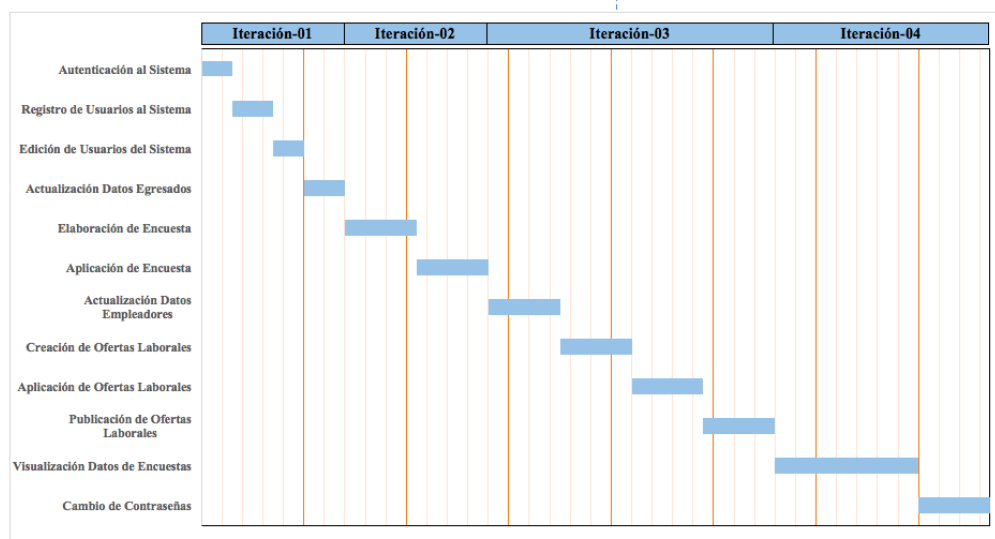


Figura 3.9. Cronograma de las tareas por iteraciones.
Fuente: autor del proyecto.

3.7. Mockups

El término “maquetado” se hace cada vez más común dentro de los proyectos ágiles, ya que ayudan a dar un soporte y complemento gráfico de manera muy general a los requisitos o historias de usuarios. Generalmente cuando se complementan las historias de usuarios con mockups suelen otorgar resultados favorables, siempre y cuando se consideren estos 2 aspectos:

- Que el soporte gráfico no haga perder la visión funcional y que las historias de usuarios sigan siendo el eje principal.
- Que al momento de generar la documentación, esta sea útil y simple.

A continuación se detallan las maquetas de las historias de usuarios de este proyecto:

The mockup shows a browser window with a single tab labeled 'Page 1'. The address bar contains 'https://www.default.com'. The main content area features a centered login form with the following elements:

- Title: Inicio de Sesión
- Input field: Email
- Input field: Contraseña (password)
- Button: Entrar

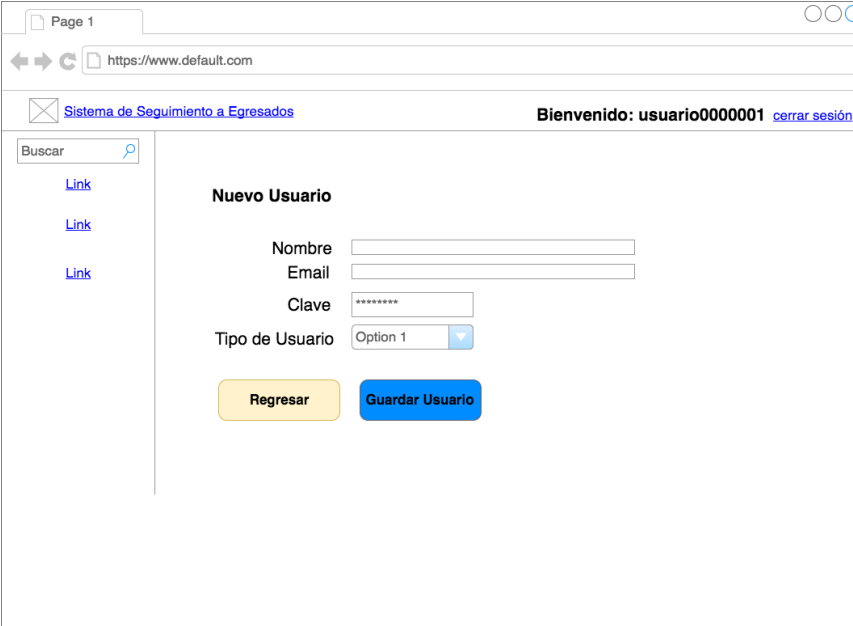
Figura 3.10. Mockup-01 autenticación al sistema
Fuente: autor del proyecto.

The mockup shows a browser window with a single tab labeled 'Page 1'. The address bar contains 'https://www.default.com'. The page content includes:

- Page Title: Sistema de Seguimiento a Egresados
- User Greeting: Bienvenido: usuario0000001 [cerrar sesión](#)
- Search Bar: Buscar
- Sidebar: Three [Link](#) items.
- Main Content:

Nombre Personal	Tipo de usuario	Nuevo Usuario
nombre 1	administrador	Editar, Eliminar
nombre 2	egresado	Editar, Eliminar

Figura 3.11. Mockup-02 registro de usuario al sistema 1
Fuente: autor del proyecto.



Page 1

https://www.default.com

Sistema de Seguimiento a Egresados Bienvenido: usuario0000001 [cerrar sesión](#)

Buscar

[Link](#)

[Link](#)

[Link](#)

Nuevo Usuario

Nombre

Email

Clave

Tipo de Usuario

[Regresar](#) [Guardar Usuario](#)

Figura 3.12. Mockup-03 registro de usuario al sistema 2
Fuente: autor del proyecto.



Page 1

https://www.default.com

Sistema de Seguimiento a Egresados Bienvenido: usuario0000001 [cerrar sesión](#)

Buscar

[Link](#)

[Link](#)

[Link](#)

Edición Usuario

Nombre

Email

Clave

Tipo de Usuario

[Regresar](#) [Actualizar Usuario](#)

Figura 3.13. Mockup-04 edición de usuario del sistema
Fuente: autor del proyecto.

Page 1 ○ ○ ○

← → ↻

✉ Sistema de Seguimiento a Egresados **Bienvenido: usuario000001** [cerrar sesión](#)

Buscar

[Link](#)
[Link](#)
[Link](#)

Actualización Empleador

Empresa

Actividad

Email

Ciudad

Dirección

Fijo

Celular

Figura 3.14. Mockup-05 actualización datos egresados
Fuente: autor del proyecto.

Page 1 ○ ○ ○

← → ↻

✉ Sistema de Seguimiento a Egresados **Bienvenido: usuario000001** [cerrar sesión](#)

Buscar

[Link](#)
[Link](#)
[Link](#)

Pagina 1 Pagina 2 **Pagina 3** Pagina 4

Encuesta a Egresados

Pregunta 1 Option 1 Option 2 Option 3 Option 4

Pregunta 2 Option 1 Option 2 Option 3 Option 4

Pregunta 3 Option 1 Option 2 Option 3 Option 4

Pregunta 4 Option 1 Option 2 Option 3 Option 4

Figura 3.15. Mockup-06 aplicación de encuesta
Fuente: autor del proyecto.

Page 1
https://www.default.com

Sistema de Seguimiento a Egresados Bienvenido: usuario0000001 cerrar sesión

Buscar

Link
Link
Link

Creación de Oferta

Oferta

Jornada

Tipo contrato

Salario

Lugar

Descripción

Educación

Experiencia

Regresar Guardar Oferta

Figura 3.16. Mockup-07 creación de ofertas laborales.
Fuente: autor del proyecto.

Page 1
https://www.default.com

Sistema de Seguimiento a Egresados Bienvenido: usuario0000001 cerrar sesión

Buscar

Link
Link
Link

Lista de Ofertas

Programador C#
MANEJO EN C#, FOXPRO, VB.NET CRYSTAL REPORTS
Ver Oferta

Programador VB.NET
MANEJO EN VB.NET CRYSTAL REPORTS
Ver Oferta

Figura 3.17. Mockup-08 aplicación de ofertas laborales 1
Fuente: autor del proyecto.

Page 1 ○ ○ ○

← → ↻

✉ [Sistema de Seguimiento a Egresados](#) Bienvenido: usuario0000001 [cerrar sesión](#)

Buscar

[Link](#)

[Link](#)

[Link](#)

Oferta Laboral

[Regresar](#) [Aplicar Oferta](#)

Oferta

Jornada

Tipo contrato

Salario

Lugar

Descripción

Educación

Experiencia

Figura 3.18. Mockup-09 aplicación de ofertas laborales 2
Fuente: autor del proyecto.

Page 1 ○ ○ ○

← → ↻

✉ [Sistema de Seguimiento a Egresados](#) Bienvenido: usuario0000001 [cerrar sesión](#)

Buscar

[Link](#)

[Link](#)

[Link](#)

Lista de Ofertas Laborales

Oferta	Lugar	Nuevo Oferta
Oferta 1	JIPIJAPA	Info, Editar, Publicar , Quitar
Oferta 2	MANTA	Info, Editar, Publicar , Quitar

Figura 3.19. Mockup-10 publicación de ofertas laborales.
Fuente: autor del proyecto.

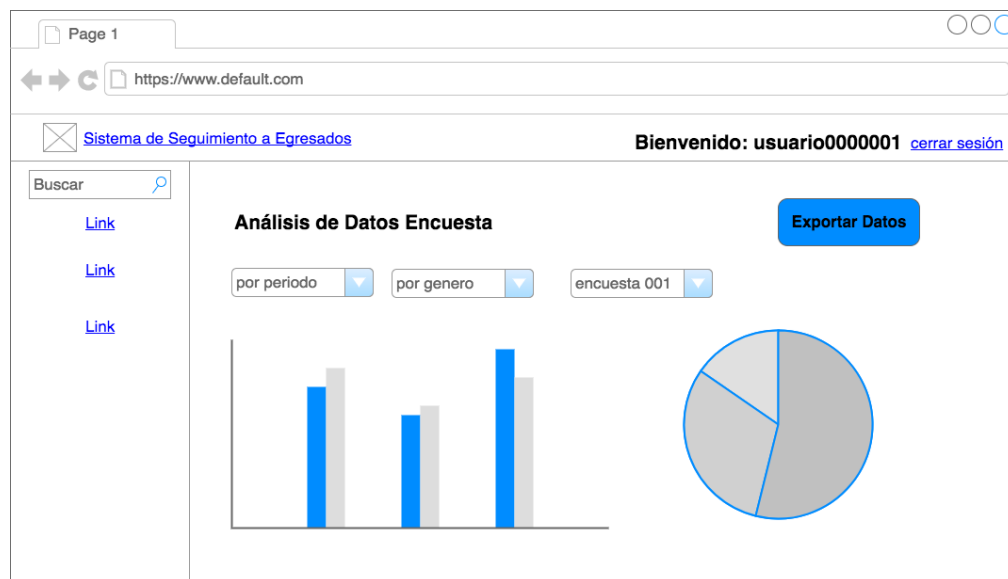


Figura 3.20. Mockup-11 visualización encuestas.
Fuente: autor del proyecto.

Figura 3.21. Mockup-12 cambio de contraseña.
Fuente: autor del proyecto.

CAPÍTULO 4

ANÁLISIS DE LA SOLUCIÓN

4.1. Estudio del Desarrollo de la Propuesta

4.1.1. Descripción de la Propuesta

La aplicación tiene como objetivo administrar el proceso de seguimiento a egresados que actualmente se realiza de forma parcial en la Unidad Educativa Redemptio de la ciudad de Jipijapa, donde la institución ha perdido vínculos importantes con sus ex dirigidos. La solución permitirá a los graduados involucrarse en la mejora y crecimiento del colegio, mediante instrumentos que permitan recopilar datos (encuestas) que muestran las fortalezas y debilidades. Adicionalmente beneficiarse del interés de empresas o instituciones que logren convertirse en oportunidades de empleo.

La información almacenada en base de datos nos dará un acceso oportuno a la misma y que la manipulación de los datos se haga con mucha rapidez, incrementando la productividad del proceso. Es importante decir que la aplicación fue diseñada para beneficiarse de los recursos tecnológicos existentes de la unidad educativa.

La solución está parametrizada para administrar y gestionar los egresados, encuestas, empleadores y administradores del sistema.

4.1.2. Alcance de la Propuesta

El sistema gestionará todo lo referente al egresado, actualización datos, hoja de vida, responder encuestas, participar en ofertas de trabajo. El empleador podrá actualizar los datos de su institución y promocionar oportunidades laborales. Adicionalmente la institución estableció utilizar el registro de egresados de los últimos 8 años, debido a que es la única información consistente que se maneja.

4.2. Estudio de Factibilidad

Una vez realizado un análisis de los requerimientos tecnológicos tanto de hardware como de software en la Unidad Educativa Redemptio de la ciudad de Jipijapa, se evidencia que estos dos elementos fundamentales cumplen con las exigencias necesarias para una correcta implementación de la solución, lo que permitió comprobar la viabilidad del proyecto que se detalla a continuación en estos 3 aspectos; Técnico, Operativo y Económico.

4.2.1. Factibilidad Técnica

De acuerdo al recurso tecnológico otorgado por la institución, se desarrollaron las Tablas 37 y 38 donde se detallan los requisitos de hardware y software necesarios para la implementación del producto.

Tabla 37. Requisitos de hardware para la implementación del producto.

Descripción	Mínimos	Existentes
Procesador	2 núcleos	Intel® Xeon® E5-2403, 10MB Cache
Velocidad del Procesador	1.2 GHz	1.80GHz
Memoria RAM	512 Mb.	8GB RDIMM
Espacio en Disco Duro	100 Gb.	1 TB
Precio	\$350	\$3900

Fuente: autor del proyecto.

Tabla 38. Requisitos de software para la implementación del producto.

Software	Descripción	Precio
Sistema Operativo	Linux	0,00
Base Datos	MongoDB	0,00
Lenguaje de Programación	Node.js, Angular 2	0,00
Total		0,00

Fuente: autor del proyecto.

4.2.2. Factibilidad Operativa

Actualmente la información de seguimientos a egresados se la maneja con hojas de cálculo, no siempre tomando conciencia al momento de llenar de forma correcta los datos, por no contar con una aplicación que facilite el manejo de este proceso. Para esto, la solución mediante una interfaz amigable al usuario, permitirá almacenar todos los datos necesarios.

4.3. Costos de Desarrollo y Financiamiento

4.3.1. Costos de Desarrollo

Tabla 39. Costo de recursos del proyecto

Recursos Humanos			
Recurso	Horas	Costo(\$)	Total
Asesora de Tesis	40	5,00 USD	200,00 USD
Elaboración de Documento	200	3,00 USD	600,00 USD
Programación del Producto	420	8,00 USD	3.360,00 USD
Subtotal			4.160,00 USD
Recursos Materiales			
Recurso	Cantidad	Valor Unitario	Total
Cartuchos de Tinta	3	15,00 USD	45,00 USD
Pen drivers	1	5,00 USD	5,00 USD
Suministros de Oficina	1	250,00 USD	250,00 USD
Subtotal			300,00 USD
Recursos Tecnológicos			
Costo de Hardware			
Depreciación de Equipo	1	90,00 USD	90,00 USD
Impresora	1	150,00 USD	150,00 USD
Costos de Software			
MongoDB	1	0,00 USD	0,00 USD
Angular 2, Node.js	2	0,00 USD	0,00 USD
Subtotal			240,00 USD
Otros			
Recurso	Horas	Valor Unitario	Total
Viáticos	1	50,00 USD	50,00 USD
Internet	500	0,20 USD	100,00 USD
Energía Eléctrica	500	0,05 USD	25,00 USD
Subtotal			175,00 USD

Fuente: autor del proyecto.

Tabla 40. Costos totales del proyecto

Total de Costo del Proyecto	
Recursos Humanos	4.160,00 USD
Recursos Materiales	300,00 USD
Recursos Tecnológicos	240,00 USD
Otros	175,00 USD
Total Proyecto	4.875,00 USD

Fuente: autor del proyecto.

4.3.2. Financiamiento

Los costos detallados en las Tablas 39 y 40 fueron financiados en su totalidad por el autor de este proyecto.

4.4. Análisis de Costos / Beneficios

4.4.1. Costo

Tabla 41. Costo de inversión.

Costos de Inversión		Valores
Infraestructura	Configuración de un servidor para el procesamiento de datos.	\$ 1400,00
Institución	Consultoría para la instalación y ejecución del proyecto.	\$ 0
Entrenamiento	Costo de entrenamiento para los usuarios.	\$ 0
Costo Total de la Inversión (C.T.I.)		\$1400,00

Fuente: autor del proyecto.

4.4.2. Beneficio

Se han cuantificado 420 horas de trabajo de programación durante el proceso de desarrollo de la aplicación, bajo un precio de \$8.00 la hora.

Tabla 42. Beneficios de inversión.

Beneficios de Inversión		Valores
Costo del desarrollo de aplicación a implementar	El uso de tecnología open source nos permite ahorrar en pagos de licencias y futuras actualizaciones.	\$ 3360,00
Mantenimiento	Se implementó la metodología SCRUM lo que admite escalabilidad del mismo	\$ 1500,00
Beneficio Total de la Solución (B.T.S.)		\$ 4860,00

Fuente: autor del proyecto.

4.4.3. Relación Costo / Beneficio

Una vez obtenido los valores de costos (véase Tabla 41) y los beneficios (véase Tabla 42), se procedió a realizar la respectiva relación:

CT = BENEFICIOS sobre COSTOS

CT = \$4860 / \$1400 = 3.47

CT = 3.47

CT > 1 es factible económicamente

El resultado de esta relación nos muestra que los beneficios son 3 veces más a los costos envueltos en el desarrollo del proyecto con su respectiva implementación. Teniendo esto, se resuelve que el proyecto será económicamente rentable.

4.4.4. Análisis de Retorno de Inversión

ROI es uno de los ratios más representativos del mundo financiero, para calcular análisis de retorno de inversión.

$$\text{ROI} = (\text{BENEFICIOS} - \text{INVERSIÓN}) / \text{INVERSIÓN}$$

$$\text{ROI} = (\$4860 - \$1400) / \$1400$$

$$\text{ROI} = 2.47$$

De acuerdo al ROI el beneficio de la inversión es del 247%, lo que denota que por cada dólar invertido estamos ganando \$2.47, siempre cuando hagamos la resta del costo de de inversión.

4.4.5. Tiempo de Retorno de Inversión

Analizando la Tabla 41 de este capítulo, donde la inversión es de \$1400, se concluye que el tiempo de retorno de la inversión para este proyecto será de 4 meses aproximadamente, véase en Tabla 43.

Tabla 43. Sueldo del personal vinculado al proyecto.

	Sueldo en dólares (por mes)	Total Horas de Trabajo (por mes)	Precio por Hora
Empleado A	\$640	160	\$4
Total Mes	\$640		

Fuente: autor del proyecto.

Tabla 44. Ahorro por implementación.

Horas de Ahorro (aprox.)	Costo de Ahorro (por día)	Costo de Ahorro (por semana)	Costo de Ahorro (por mes)
5	\$20	\$100	\$400
Total			\$400

Fuente: autor del proyecto.

CAPÍTULO 5

DISEÑO Y DESARROLLO DE LA SOLUCIÓN

5.1. Diseño Lógico de la Solución

Durante esta etapa se realizó la construcción y verificación del modelo de datos lógico de acuerdo a los requerimientos establecidos, en los cuales se guardará todos los datos relacionados a los estudiantes, empresas empleadoras e instrumentos de evaluación.

5.1.1. Diseño de Salida

La aplicación de seguimiento a graduados posee 2 tipos de salidas, ya sea por pantalla y medio físico (impresión), los mismos que no se ejecutan de forma automática, solo mediante solicitud del usuario.

5.1.2. Diseño de Entrada

Se realiza la captura por pantalla de los datos actualizados de los estudiantes y empresas empleadoras. Una vez estos usuarios se encuentren actualizados en el sistema podrán realizar lo siguiente:

- Los estudiantes podrán hacer uso del instrumento de evaluación (encuesta).
- Los empleadores podrán publicar ofertas de trabajo para los estudiantes.

5.1.3. Diagrama de Estructura de Datos

La aplicación de seguimientos a graduados utilizó el modelado de datos que se presenta a continuación, para representar un conjunto de datos y relaciones entre si, describiendo de forma colectiva los componentes del sistema. Este esquema ayuda a representar las asociaciones entre las distintas entidades.

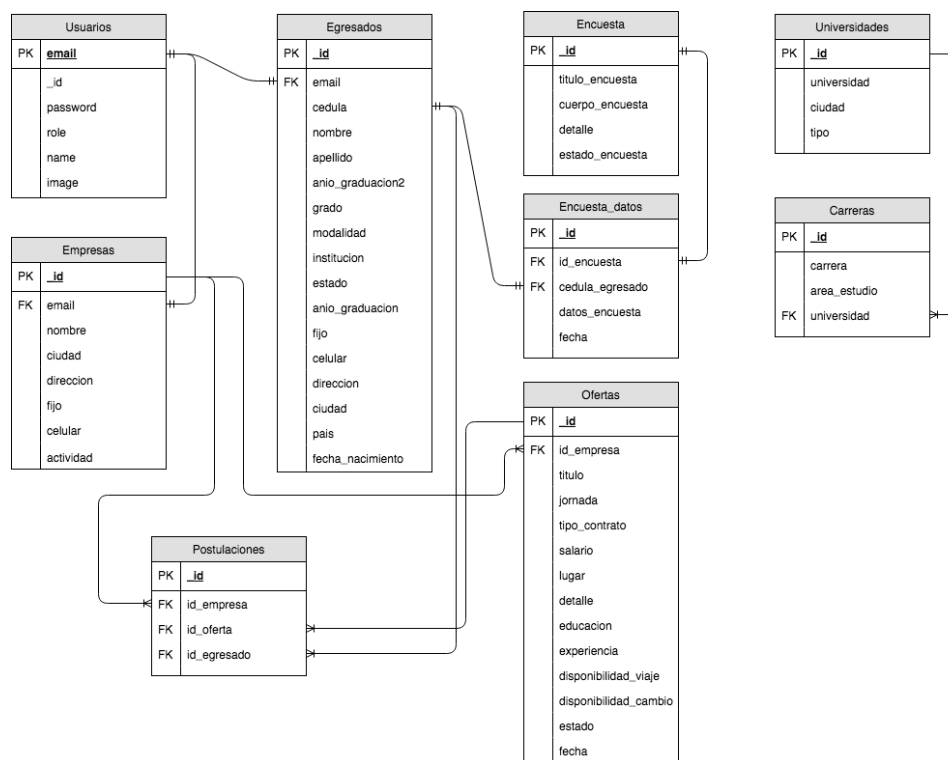


Figura 5.1. Diagrama de Estructura de Datos
Fuente: autor del proyecto.

5.1.4. Diccionario de Datos

El diccionario de datos tiene como objetivo precisar información que se manipula en la aplicación de seguimiento a egresados, evitando así ambigüedades, para ello utilizaremos todos los datos de las distintas entidades. El detalle de los datos es un buen complemento al diagrama de estructura de datos.

Tabla 45. Tabla de usuarios.

Nombre de la tabla	usuarios	
Descripción de la tabla	Contiene datos de los usuarios (egresados, empresas, administradores)	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
email	String	Correo electrónico del usuario
password	String	Contraseña del usuario
name	String	Nombre personal del usuario
role	String	Identificado tipo de usuario
image	String	Ruta de la imagen del usuario

Fuente: autor del proyecto.

Tabla 46. Tabla de empresas.

Nombre de la tabla	empresas	
Descripción de la tabla	Contiene datos de las empresas o empleadores	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
nombre	String	Nombre de empresa o empleador independiente
email	String	Correo electrónico del empleador
ciudad	String	Ciudad de residencia
direccion	String	Dirección actual del empleador
fijo	String	Teléfono de oficinas
celular	String	Celular del empleador
actividad	String	Actividad económica del empleador

Fuente: autor del proyecto.

Tabla 47. Tabla de encuestas.

Nombre de la tabla	encuestas	
Descripción de la tabla	Contiene la estructura del instrumento de evaluación	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
titulo_encuesta	String	Título de encuesta
detalle	String	Descripción del contenido de la encuesta
cuerpo_encuesta	Object	Estructura de datos de encuesta
estado_encuesta	String	Activación de encuesta actual de proceso

Fuente: autor del proyecto.

Tabla 48. Tabla datos de la encuesta.

Nombre de la tabla	encuesta_datos	
Descripción de la tabla	Contiene los datos del instrumento de evaluación	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
id_encuesta	String	Clave foránea de la estructura de la encuesta
cedula_egresado	String	Clave foránea del estudiante que resuelve la encuesta
datos_encuesta	String	Datos de la encuesta resuelta
fecha	Date	Fecha que se realiza la transacción

Fuente: autor del proyecto.

Tabla 49. Tabla de universidades

Nombre de la tabla	universidades	
Descripción de la tabla	Contiene los datos de las universidades	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerado
universidad	String	Nombre de universidad
ciudad	String	Ciudad de la universidad
tipo	String	Tipo de universidad (pública, privada)

Fuente: autor del proyecto.

Tabla 50. Tabla de carreras universitarias.

Nombre de la tabla	carreras	
Descripción de la tabla	Contiene los datos de las carreras por universidad	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerado
universidad	String	Clave de universidad
area_estudio	String	Área de estudio de la carrera
carrera	String	Nombre de la carrera

Fuente: autor del proyecto.

Tabla 51. Tabla egresado.

Nombre de la tabla	egresados	
Descripción de la tabla	Contiene datos de los egresados y su situación actual	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
nombre	String	Nombres del egresado
apellido	String	Apellidos del egresado
cedula	String	Cédula del egresado
email	String	Correo electrónico del egresado
fecha_nacimiento	String	Fecha de nacimiento del egresado
pais	String	País de residencia del egresado
ciudad	String	Ciudad de residencia del egresado
direccion	String	Dirección actual del egresado
celular	String	Numero celular del egresado
fijo	String	Teléfono del domicilio
anio_graduacion	String	Año de graduación del egresado
estado	String	Situación académica actual
institucion	String	Institución académica donde estudia o estudió
modalidad	String	Modalidad de estudio (presencial, distancia, otros)
grado	String	Título académico alcanzado

Fuente: autor del proyecto.

Tabla 52. Tabla de las postulaciones

Nombre de la tabla	postulaciones	
Descripción de la tabla	Datos de postulaciones de los egresados	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
id_empresa	String	Clave de empleador
id_oferta	String	Clave de la oferta
id_egresado	String	Clave del egresado

Fuente: autor del proyecto.

Tabla 53. Tabla de las ofertas

Nombre de la tabla	ofertas	
Descripción de la tabla	Datos de las ofertas publicadas por el empleador	
Campo	Tipo de dato	Descripción
_id	ObjectId	Identificador único autogenerated
titulo	String	Título de la oferta
jornada	String	Jornada laboral de oferta
tipo_contrato	String	Tipo de contrato de oferta
salario	String	Detalle del salario de oferta
lugar	String	Ciudad específica de oferta
detalle	String	Descripción de oferta
educacion	String	Nivel de educación de oferta
experiencia	String	Años de experiencia de oferta
disponibilidad_viaje	String	Posibilidad de viajar
disponibilidad_cambio	String	Cambio de domicilio
id_empresa	String	Clave de empleador
estado	String	Estado activo/inactivo de oferta
fecha	Date	Fecha de publicación de oferta

Fuente: autor del proyecto.

5.2. Diseño Físico de la Solución

5.2.1. Modelo Cliente-Servidor

En la Figura 5.2 de la arquitectura física del sistema de seguimiento a egresados, se muestra que la Unidad Educativa Redemptio de la ciudad de Jipijapa posee un servidor con salida a internet, el cual permite a los usuarios de la aplicación lograr acceder desde cualquier dispositivo con acceso a la web mediante un navegador o browser.

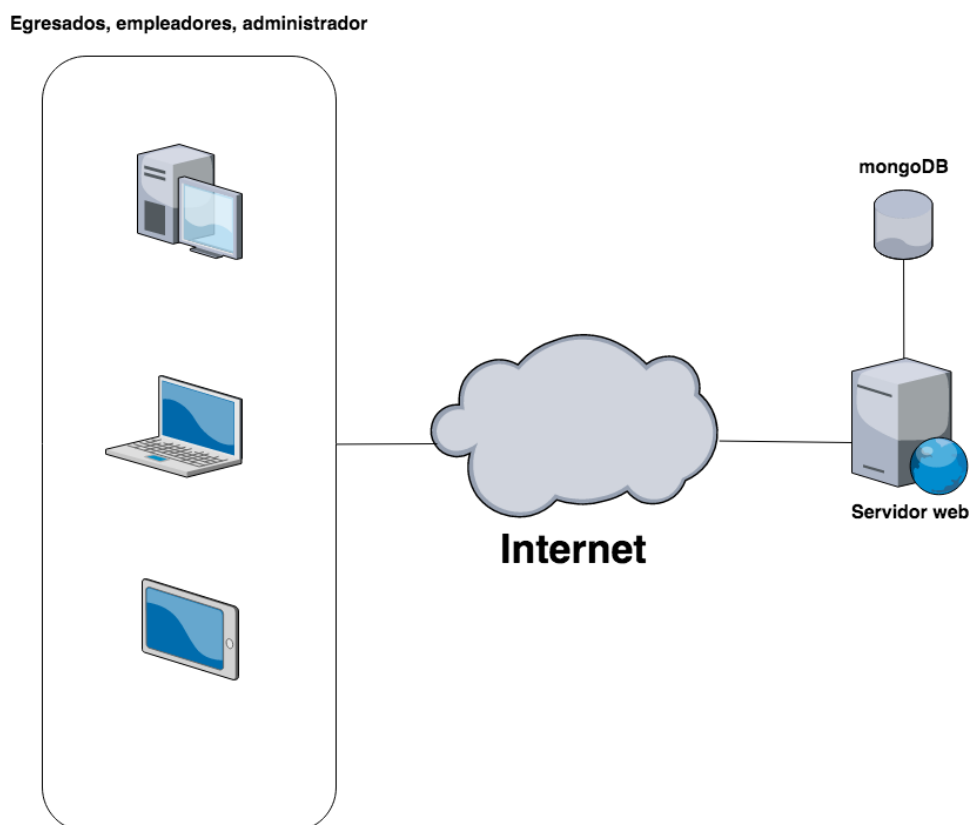


Figura 5.2. Arquitectura física de la aplicación.
Fuente: autor del proyecto.

5.2.2. Ventajas y Desventajas del Modelo

La principal ventaja del modelo es la centralización de los recursos y datos, la cual permite añadir nuevos servidores e integrarlos de manera fácil al sistema. Sin embargo, si la cantidad de clientes conectados de manera simultánea es elevada, el servidor podría saturarse y provocar fallas que paralicen los servicios de lado del cliente. En este modelo se pueden realizar la carga de tareas de procesamiento de datos en el lado del servidor como de lado del cliente.

Para la solución de seguimiento a egresados hemos implementado el modelo de tarea *cliente liviano o ligero*, ya que el servidor realizará la mayor carga de trabajo y los clientes accederán mediante un navegador sin necesidad de instalar la aplicación en cada terminal.

5.3. Opciones de Desarrollo de Software

Dentro de las opciones de desarrollo para la aplicación de seguimiento a egresados, se tomaron todas las alternativas que logren satisfacer los requerimientos y objetivos, de tal forma que brinde una solución de administración e inteligencia de la información.

5.3.1. Técnicas de Desarrollo

5.3.1.1. Programación Orientada a Objetivos (P.O.O.)

Este paradigma se basa en 4 pilares fundamentales tales como la herencia, la abstracción, el polimorfismo y el encapsulamiento, que brindan un toque sofisticado y profesional en el trascurso del desarrollo de software.

5.3.1.2. Arquitectura de Capas

Se ha considerado esta técnica de programación ya que permite construir la solución por niveles, en donde cada capa de trabajo se encuentra abstraída de una de la otra. En el caso de la aplicación web de seguimiento a egresados, permitió tener organizados todos los archivos de acuerdo a su capa respectivamente. En la Figura 5.3 se muestra los niveles de nuestra aplicación.

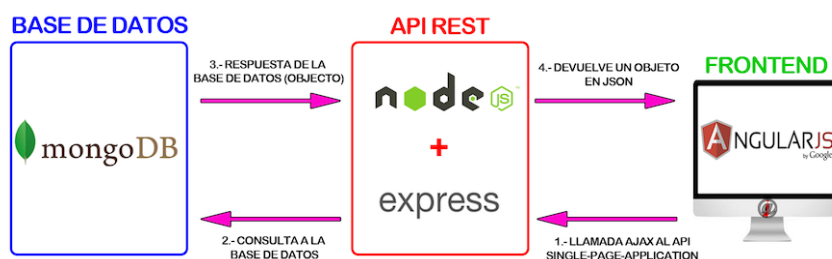


Figura 5.3.Arquitectura de capas de la solución [18].

5.3.2. Herramientas de Desarrollo

5.3.2.1. Survey.JS

Es una librería construida en JavaScript y el manejo de datos basado en objetos JSON, permite crear y resolver encuestas de manera sencilla y elegante. Actualmente es un proyecto de código abierto que también consta de otro componente SurveyEditor.JS que permite crear las encuestas de manera más rápida. Actualmente es un producto licenciado [19].

5.4. Implementación

Para la implementación de la aplicación se tuvieron 2 opciones, alojarla en un servidor físico o contratar un servicio en la nube. La Unidad Educativa Redemptio de la ciudad de Jipijapa dentro de su infraestructura tecnológica cuenta con varios servidores utilizados para sus distintos procesos académicos, en el cual se destinó una maquina equipada con un sistema operativo basado en Linux (Debian), con una IP pública como salida hacia internet.

5.4.1. Aplicaciones Necesarias para el Funcionamiento

El servidor web para la aplicación debe tener los siguientes programas o tecnologías instaladas para su funcionamiento:

- Node.js v8.1.3

- Angular CLI v1.4.7
- Servidor SSH para Debian
- GIT v2.14.1
- MongoDB v3.4.5

5.4.2. Interfaz de la Aplicación

Las interfaces de la aplicación contendrán un menú principal que permitirá administrar opciones de actualización de egresados y empresas empleadoras, resolución de encuesta, aplicación a ofertas laborales y creación de ofertas. Se ha considerado poner énfasis en un diseño simple y amigable de manera que el usuario ya sea este un egresado, un empleador o el administrador no encuentre barreras al momento de interactuar con la solución. A continuación se muestra las principales imágenes de la aplicación web.

5.4.2.1. Formulario Ingreso a la Aplicación

La página de ingreso presenta una forma de entrada y solicita ingresar el correo electrónico y la clave, para otorgar acceso a su cuenta.

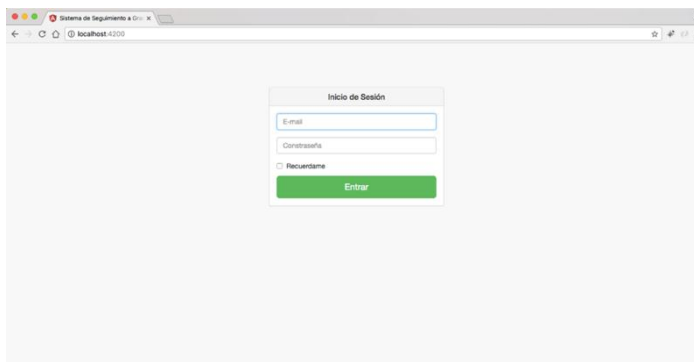


Figura 5.4. Formulario ingreso a la aplicación.
Fuente: autor del proyecto.

5.4.2.2. Formulario Dashboard de Egresados

Este formulario muestra todas las opciones a las que tienen acceso los egresados.

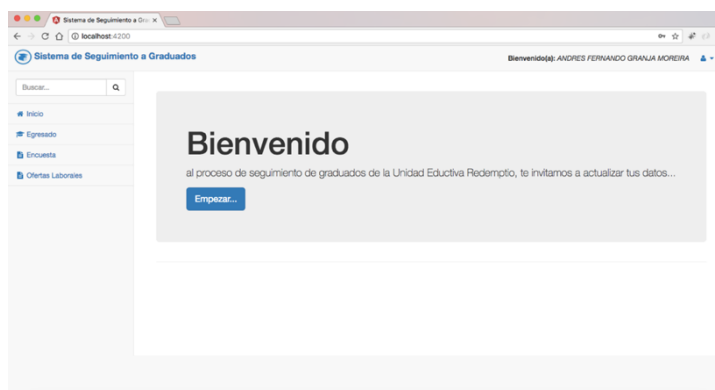


Figura 5.5. Formulario dashboard de egresados.
Fuente: autor del proyecto.

5.4.2.3. Formulario Actualización Egresado

Este formulario permite actualizar los datos del egresado.

The screenshot shows a web browser window with the URL 'localhost:4200/egresado/andresgranja001@gmail.com'. The page title is 'Sistema de Seguimiento a Graduados' and the user is logged in as 'ANDRES FERNANDO GRANJA MOREIRA'. A sidebar on the left contains navigation links: 'Inicio', 'Egresado', 'Encuesta', and 'Ofertas Laborales'. The main content area is titled 'Actualizar Datos' and contains the following fields:

- Nombres: ANDRES FERNANDO
- Apellidos: GRANJA MOREIRA
- Cédula: 1310185143
- Email: andresgranja001@gmail.com
- Ciudad Actual: JPUJANA
- Dirección Actual: BGSRTBT
- Celular: 0435465
- Fijo: 43254
- Año de Graduación: 2016
- Actividades Post-Graduación: (empty field)
- Situación: ESTUDIA

Figura 5.6. Formulario actualización egresado.
Fuente: autor del proyecto.

5.4.2.4. Formulario Encuesta Egresado

Este formulario permite resolver la encuesta de seguimiento a egresados, previamente debe actualizar los datos para poder acceder a esta opción.

The screenshot shows the same web browser window as Figure 5.6, but the URL is 'localhost:4200/egresado/encuesta'. The sidebar navigation is the same. The main content area is titled 'Encuesta' and contains the following text:

La encuesta de Seguimiento a Graduados tiene como fin analizar el desarrollo profesional y personal de los graduados. Para la Unidad Educativa Redemptio representa una excelente oportunidad para recolectar información valiosa que permita:

- A los graduados, visualizar nuevas perspectivas para complementar su proceso de formación y el acceso al mercado laboral.
- A los bachilleres, padres de familia y orientadores mejorar la toma de decisiones en procesos de selección de programas académicos.
- A la comunidad local, tener un referente de gran valor estratégico para analizar los avances en la calidad de la educación, la pertinencia de los programas y la equidad social.

At the bottom of the text area is a blue button labeled 'Empezar...'.

Figura 5.7. Formulario encuesta egresado.
Fuente: autor del proyecto.

5.4.2.5. Formulario Ofertas Laborales Egresados

Este formulario permite visualizar una lista de ofertas laborales de todas las empresas o empleadores registrados, una vez elegida una oferta se podrá aplicar a la misma.

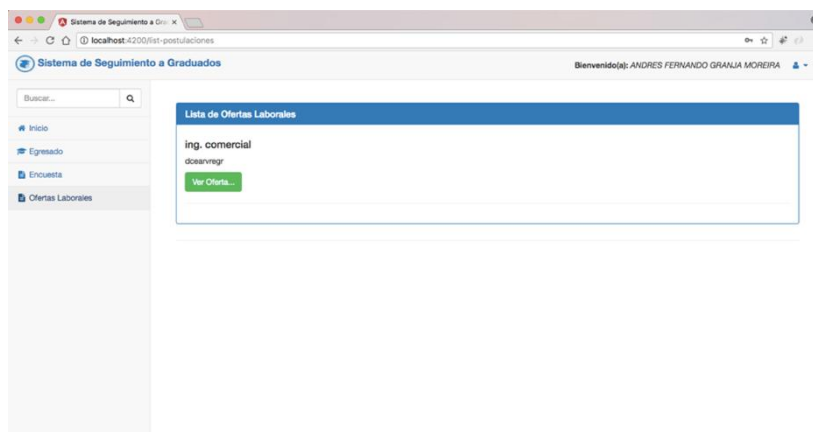


Figura 5.8. Formulario lista de ofertas laborales.
Fuente: autor del proyecto.

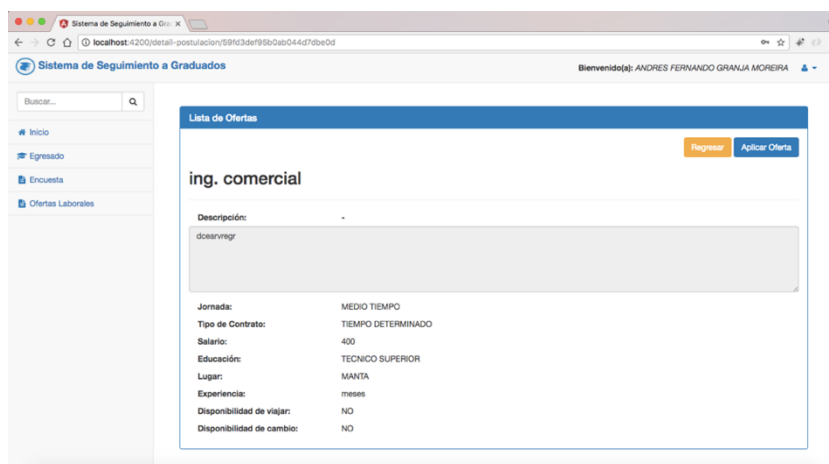


Figura 5.9. Formulario detalle de oferta laboral
Fuente: autor del proyecto.

5.4.2.6. Formulario Dashboard de Empresa/Empleador

Este formulario permite visualizar todas las opciones a las que tiene acceso la empresa o empleador.

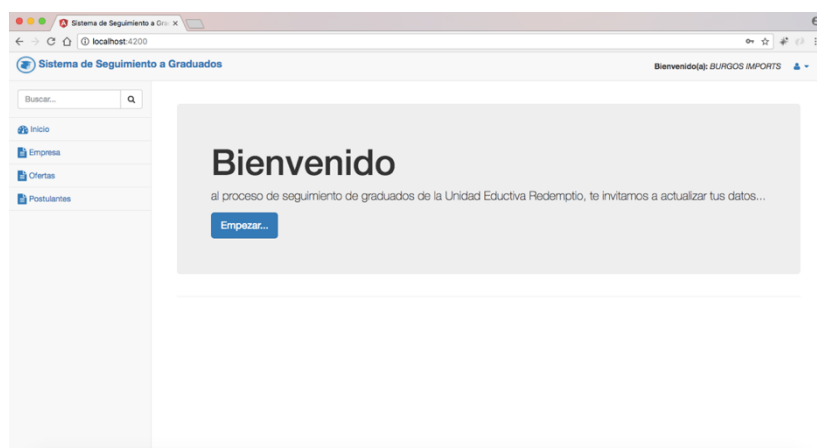


Figura 5.10. Formulario dashboard de empresa/empleador.
Fuente: autor del proyecto.

5.4.2.7. Formulario Actualización Empresa/Empleador

Este formulario permite actualizar los datos de la empresa o empleador.

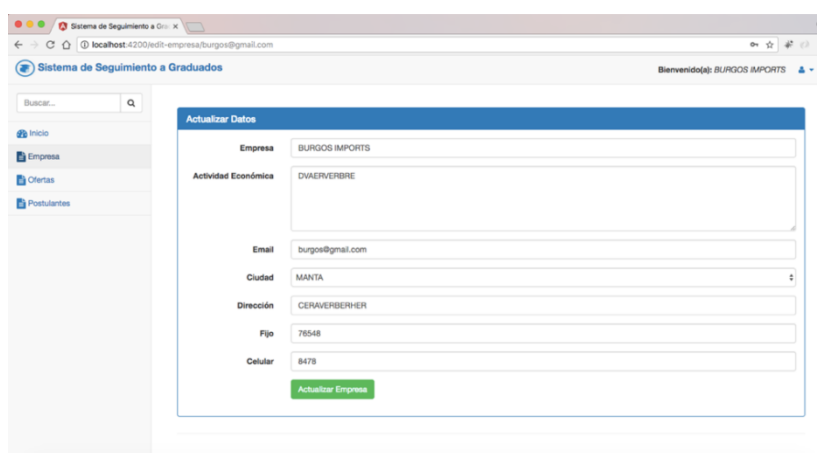


Figura 5.11. Formulario actualización empresa/empleador.
Fuente: autor del proyecto.

5.4.2.8. Formulario Ofertas Laborales Empresa/Empleador

Este formulario permite ingresar nuevas ofertas para los egresados, se podrá visualizar el detalle de las ofertas y publicar o dar de baja a la oferta.

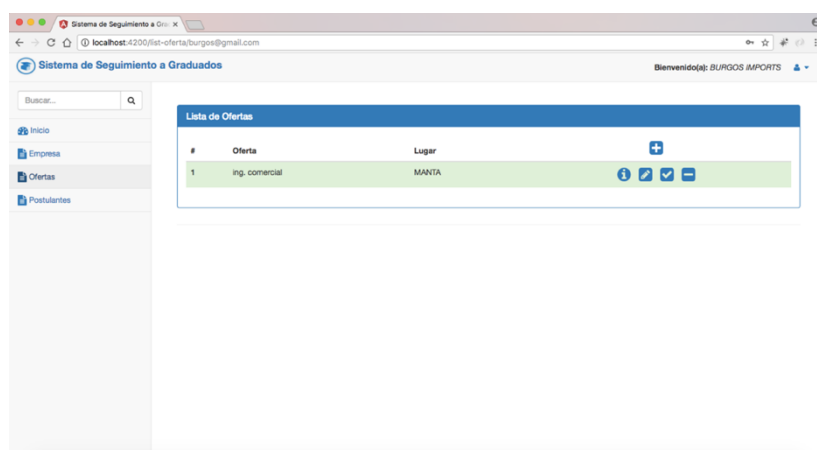


Figura 5.12. Formulario ofertas laborales empresa/empleador.
Fuente: autor del proyecto.

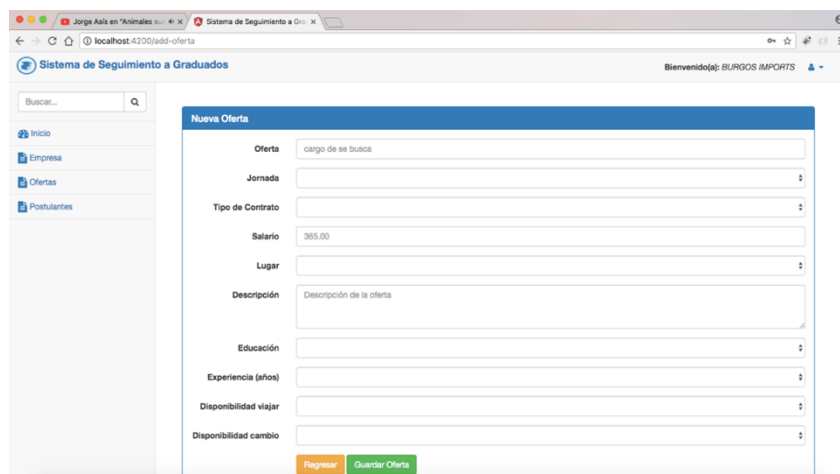


Figura 5.13. Formulario nueva oferta laboral.
Fuente: autor del proyecto.

Buscar...

Inicio
Empresa
Ofertas
Postulantes

Nueva Oferta

Oferta: ing. comercial

Jornada: MEDIO TIEMPO

Tipo de Contrato: TIEMPO DETERMINADO

Salario: 400

Lugar: MANTA

Descripción: docenreg

Educación: TÉCNICO SUPERIOR

Experiencia (años): 6 meses

Disponibilidad viajar: NO

Disponibilidad cambio: NO

Regresar Actualizar Oferta

Figura 5.14. Formulario actualización de oferta laboral.
Fuente: autor del proyecto.

Buscar...

Inicio
Empresa
Ofertas
Postulantes

Sistema de Seguimiento a Graduados Bienvenido(a): BURGOS IMPORTS

Lista de Ofertas

#	Oferta	Lugar	
1	ing. comercial	MANTA	+

Figura 5.15. Formulario de publicación oferta laboral
Fuente: autor del proyecto.

5.4.2.9. Formulario Dashboard Administrador

Este formulario permite acceder a todas las opciones a los usuarios administradores.

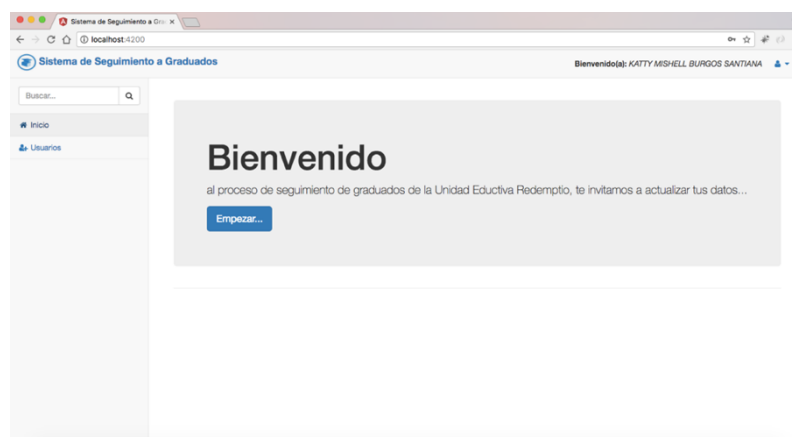


Figura 5.16. Formulario dashboard administrador.
Fuente: autor del proyecto.

5.4.2.10. Formulario Nuevos Usuarios

Este formulario permite visualizar todos los usuarios registrados, ya sean egresados o empresas/empleadores.

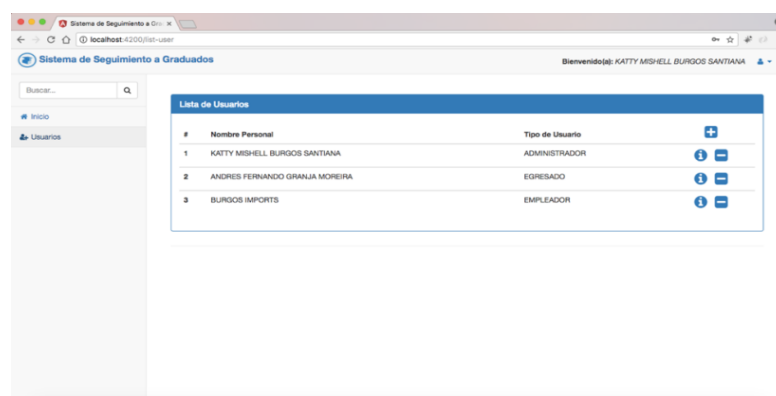


Figura 5.17. Formulario lista de usuarios de la aplicación.
Fuente: autor del proyecto.

Sistema de Seguimiento a Graduados

Bienvenido(a): KATY MISHELL BURGOS SANTANA

Buscar...

Inicio

Usuarios

Crear Usuarios

Nombre

Email

Password

Rol

Regresar Guardar Usuario

Figura 5.18. Formulario nuevo usuario de la aplicación.
Fuente: autor del proyecto.

5.4.3. Cronograma de Actividades para la Implementación

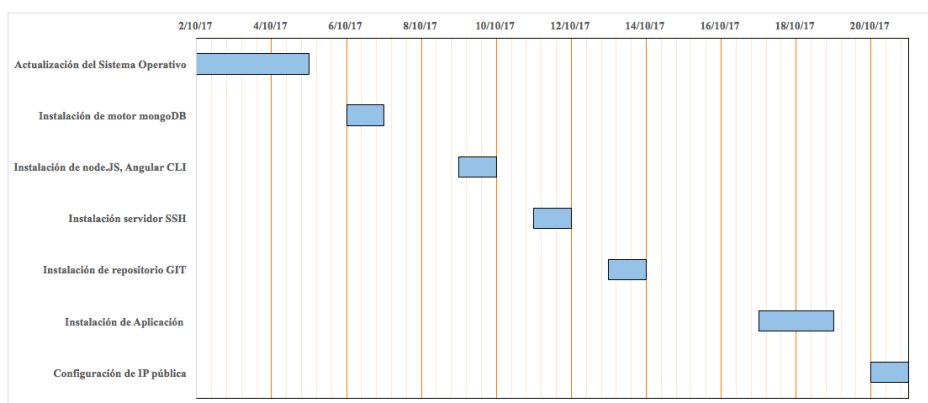


Figura 5.19. Cronograma actividades de implementación.
Fuente: autor del proyecto.

5.5. Descripción del Entorno de Prueba

Es indispensable medir el proceso de desarrollo de software para lograr obtener una solución de calidad, poder cuantificar todo lo que se ha hecho y lo que faltaría por hacer, costos, tamaño, tiempo en el desarrollo, entre otros [20]. La medición de esta solución se la realizó mediante métricas, que permitieron cuantificar los diferentes aspectos del desarrollo de la aplicación. Existen 3 parámetros básicos para medir la calidad:

- Estadísticas basadas en la experiencia.
- Avances en el producto.
- Cumplimiento de requerimientos.

5.5.1. Selección de Métricas

Para este producto seleccionamos 2 grupos a medir, la funcionalidad y la usabilidad. La información recolectada nos permitió tomar acciones en concreto y planificar mejoras en las mismas. A continuación se muestran las tablas de las métricas.

Tabla 54. Métrica de funcionalidad – adecuación [21].

Nombre:	Complejidad de implementación funcional
Propósito:	Qué tan completa está la implementación funcional.
Método de aplicación:	Contar las funciones faltantes detectadas y comparar con el número de funciones descritas en la especificación de requisitos.
Medición, fórmula:	$X = 1 - A/B$ A = número de funciones faltantes B = número de funciones descritas en la especificación de requisitos
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más completa.
Fuente de medición:	Especificación de requisitos Diseño

Tabla 55. Métrica de usabilidad – comprensión [21].

Nombre:	Funciones evidentes
Propósito:	Qué proporción de las funciones del sistema son evidentes al usuario.
Método de aplicación:	Contar las funciones evidentes al usuario y comparar con el número total de funciones.
Medición, fórmula:	$X = B/A$ A = número de funciones evidentes al usuario B = total de funciones
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mejor.
Fuente de medición:	Especificación de requisitos Diseño

5.5.2. Aplicación de Métricas

Tabla 56. Aplicando métrica de funcionalidad.

Funciones	Requisitos específicos	Funciones faltantes
Egresados	6	1
Empleadores	7	2
Administradores	12	2
Seguridad	3	1
TOTAL	28	6

Fuente: autor del proyecto.

Tabla 57. Aplicando métrica de usabilidad.

Funciones evidentes al usuario		Funciones en la aplicación
Egresados	Actualización	X
	Eliminación	-
	Resolución de encuesta	X
	Consulta de encuesta realizada	-
	Visualizar ofertas laborales	X
	Aplicar ofertas laborales	X
	Cambio de clave	X
Empleadores	Actualización	X
	Eliminación	-
	Creación de ofertas laborales	X
	Publicación de ofertas laborales	X
	Registro de egresados	X
Administradores	Registro de empleadores	X
	Consulta de encuestas	X
	Análisis estadístico por fecha	X
	Análisis estadístico por pregunta	X
	Análisis estadístico por periodos	X

Fuente: autor del proyecto.

5.5.3. Análisis de Métricas

Se pudo observar que un 79% de la aplicación de seguimiento a egresados es funcional, debido a que surgieron nuevos requerimientos a nivel de administrador durante el desarrollo, acerca de nuevos formularios de análisis de datos. A nivel de uso el producto cuenta con un 82% de usabilidad.

CAPÍTULO 6

ANÁLISIS DE RESULTADOS

6.1. Introducción

Durante este capítulo, luego del desarrollo y la implementación de la solución final en la Unidad Educativa Redemptio de la ciudad de Jipijapa, se procederá a efectuar la respectiva recolección de los datos, que nos permita conocer si los objetivos planteados se cumplen, a consecuencia de implementar la aplicación.

6.2. Resultados Obtenidos en las Pruebas de Usuarios

Para las pruebas de usuarios se utilizaron 2 instrumentos de evaluación que se aplicaron durante y al final del proyecto, donde se tomó a consideración lo siguiente:

- La evaluación 1 se la realizó al dueño del producto, ver el anexo 2.

- La evaluación 2 se la realizó al scrum master del proyecto, ver el anexo 3.
- La evaluación 3 se la realizó a los egresados y al administrador del aplicativo, ver el anexo 4.

6.2.1. Instrumento #1 – Usabilidad

Durante la finalización de cada iteración, se le solicitó realizar la presente encuesta al dueño del proyecto la Dra. Luisa Ponce, quien por iniciativa propia decidió participar en cada fase del proyecto. Obteniendo los siguientes datos:

Tabla 58. Resultado del instrumento #1 usabilidad - iteración 01

Instrumento #1 – Usabilidad	Iteración 01
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente	
Preguntas	Resultado
¿Cómo califica la manera de manipular la aplicación?	4
¿Cómo califica la interfaz de la aplicación?	3
¿Cómo califica la forma de aprender a usar la aplicación?	3
¿Cómo califica la capacidad de entender el sistema?	4

Fuente: autor del proyecto.

Tabla 59. Resultado del instrumento #1 usabilidad - iteración 02

Instrumento #1 – Usabilidad		Iteración 02
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente		
Preguntas		Resultado
¿Cómo califica la manera de manipular la aplicación?		4
¿Cómo califica la interfaz de la aplicación?		5
¿Cómo califica la forma de aprender a usar la aplicación?		4
¿Cómo califica la capacidad de entender el sistema?		4

Fuente: autor del proyecto

Tabla 60. Resultado del instrumento #1 usabilidad - iteración 03

Instrumento #1 – Usabilidad		Iteración 03
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente		
Preguntas		Resultado
¿Cómo califica la manera de manipular la aplicación?		4
¿Cómo califica la interfaz de la aplicación?		5
¿Cómo califica la forma de aprender a usar la aplicación?		4
¿Cómo califica la capacidad de entender el sistema?		4

Fuente: autor del proyecto.

Tabla 61. Resultado del instrumento #1 usabilidad - iteración 04

Instrumento #1 – Usabilidad		Iteración 04
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente		
Preguntas		Resultado
¿Cómo califica la manera de manipular la aplicación?		5
¿Cómo califica la interfaz de la aplicación?		5
¿Cómo califica la forma de aprender a usar la aplicación?		5
¿Cómo califica la capacidad de entender el sistema?		4

Fuente: autor del proyecto.

Luego de la recolección de un total de 16 preguntas del instrumento #1, se aspira obtener en la medición un resultado igual o mayor que 4 en cada pregunta. A continuación se muestra el resultado.

Resultado del instrumento #1; $X = (14 / 16) * 100 = 87,5 \%$

6.2.2. Instrumento #2 – Seguridad

Tabla 62. Resultado del instrumento #2 seguridad

Instrumento #2 – Seguridad	
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente	
Preguntas	Resultado
¿Cómo califica la seguridad de la aplicación?	4
¿Cómo califica la forma de ingresar al sistema?	5
¿Cómo califica la manera como se visualiza la información personal del usuario en la aplicación?	4

Fuente: autor del proyecto.

En la entrega final de la solución se le realizó la siguiente encuesta al scrum master del proyecto Lcdo. Fulco Pincay, donde se aspira obtener en la medición un resultado igual o mayor que 4 en cada pregunta. A continuación se muestra el resultado.

Resultado del instrumento #2; $X = (3 / 3) * 100 = 100 \%$

6.2.3. Instrumento #3 – Mantenibilidad

Durante la finalización de cada iteración, se le solicitó realizar la presente encuesta al scrum master del proyecto el Lcdo. Fulco Pincay, obteniendo los siguientes datos:

Tabla 63. Resultado del instrumento #3 mantenibilidad – iteración 01

Instrumento #3 – Mantenibilidad		Iteración 01
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente		
Preguntas	Resultado	
¿Cómo califica la capacidad de hacer cambios en la aplicación?	4	
¿Cómo califica la estabilidad de la aplicación?	3	
¿Cómo califica la forma de probar las funciones de la aplicación?	4	

Fuente: autor del proyecto.

Tabla 64. Resultado del instrumento #3 mantenibilidad - iteración 02

Instrumento #3 – Mantenibilidad		Iteración 02
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente		
Preguntas	Resultado	
¿Cómo califica la capacidad de hacer cambios en la aplicación?	4	
¿Cómo califica la estabilidad de la aplicación?	3	
¿Cómo califica la forma de probar las funciones de la aplicación?	4	

Fuente: autor del proyecto.

Tabla 65. Resultado del instrumento #3 mantenibilidad - iteración 03

Instrumento #3 – Mantenibilidad	Iteración 03
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente	
Preguntas	Resultado
¿Cómo califica la capacidad de hacer cambios en la aplicación?	4
¿Cómo califica la estabilidad de la aplicación?	3
¿Cómo califica la forma de probar las funciones de la aplicación?	4

Fuente: autor del proyecto.

Tabla 66. Resultado del instrumento #3 mantenibilidad - iteración 04

Instrumento #3 – Mantenibilidad	Iteración 04
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente	
Preguntas	Resultado
¿Cómo califica la capacidad de hacer cambios en la aplicación?	4
¿Cómo califica la estabilidad de la aplicación?	4
¿Cómo califica la forma de probar las funciones de la aplicación?	4

Fuente: autor del proyecto.

Luego de la recolección de un total de 12 preguntas del instrumento #3, se aspira obtener en la medición un resultado igual o mayor que 4 en cada pregunta. A continuación se muestra el resultado.

Resultado del instrumento #3; $X = (9 / 12) * 100 = 75 \%$

6.2.4. Instrumento #4

Dentro del análisis estadístico realizado a las 6 preguntas del anexo 5, donde participaron 33 estudiantes egresados de la Unidad Educativa Redemptio de la ciudad de Jipijapa, del periodo 2015 – 2016, obtuvimos los siguientes datos:

- **¿Cómo le resulta la navegación o interacción con la aplicación de seguimiento a egresados?**

Nos muestra que el 48,5% de egresados les parece relativamente sencilla la navegación en el aplicativo, el 30,3% normal y al 21,2% le resultó muy sencillo.

- **¿Cómo le resulta la actualización de su información en la aplicación de seguimiento a egresados?**

Nos muestra que el 75,8% de egresados les parece muy fácil actualizar los datos, el 15,2% relativamente fácil, y el 9,1% les pareció normal.

- **¿Cómo calificas el aspecto de nuestra aplicación web?**

Nos muestra que el 81,8% de egresados califican la aplicación como muy buena, mientras que el resto con 18,2% les resultó normal.

- **¿Cómo le resulta la búsqueda de ofertas laborales a través de nuestra aplicación web?**

Nos muestra que el 60,6% de egresados les resulta normal la búsqueda de ofertas, el 24,2% les resulta muy sencilla y el 15,2 les pareció algo complicada.

- **¿Se siente satisfecho(a) con nuestra aplicación web?**

Nos muestra que el 72,7% se sienten satisfechos(as) con la aplicación web, el 21,2% está muy satisfechos(as), y el 6,1% medianamente satisfechos(as).

- **¿Recomendarías nuestra aplicación de seguimiento a egresados a otras personas?**

Nos muestra que el 36,4% probablemente si recomendarían la aplicación de seguimiento a egresados a otras personas, el 30,3% si, definitivamente, el 21,2% probablemente no, y el 12,1% no lo sabe.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Luego del análisis de las distintas herramientas que se aplicaron para el desarrollo de la solución se ha llegado a las siguientes conclusiones:

- 1) Se ha desarrollado una aplicación para la Unidad Educativa Redemptio de la ciudad de Jipijapa, la misma que dará acceso al seguimiento de los estudiantes que han logrado terminar con satisfacción sus estudios en esta institución.
- 2) Se empleó la metodología SCRUM para el desarrollo de la solución, que permitió darle valor al negocio, aprovechando al máximo el resultado de lo que se elabora, especialmente la flexibilidad que tiene al momento de realizar cambios en los requerimientos.

- 3) Se implementó tecnologías open source robustas, con comunidades muy activas, que permitió obtener una solución estable y funcional.
- 4) Se puede concluir que la aplicación es un medio oportuno para la obtención de información que permita verificar la realidad de los egresados y la toma de decisiones en beneficio de la institución.

Recomendaciones

- 1) Mantener la información actualizada según la necesidad.
- 2) Hacer seguimiento de las herramientas tecnológicas open source utilizadas y sus actualizaciones, lo que permitirá obtener mayor provecho de las mismas.
- 3) Desarrollar un módulo de análisis de datos más robusto.
- 4) Coordinar con las instituciones responsables, la socialización de la aplicación, con el propósito de crear relaciones laborales con posibles empresas candidatas.

BIBLIOGRAFÍA

- [1] A. Marqués, Conceptos sobre APIs REST, <http://asiermarques.com/2013/conceptos-sobre-apis-rest/>, fecha de consulta julio 2017.
- [2] I. Salvadori y F. Siqueira, A Maturity Model for Semantic RESTful Web APIs, *IEEE International Conference on Web Services*, 2015, pp. 703-710.
- [3] M. Alvarez, Ventajas e inconvenientes de API REST para el desarrollo, <https://desarrolloweb.com/articulos/ventajas-inconvenientes-apirest-desarrollo.html>, fecha de consulta julio 2017.
- [4] J. Tejero, SPA un paradigma de Arquitectura de Aplicaciones Web en auge, <https://cink.es/blog/2013/10/07/spa-un-paradigma-de-arquitectura-de-aplicaciones-Web-en-auge>, fecha de consulta julio 2017.
- [5] V. Blanco Rodriguez, Super héroes en la web con AngularJS, <http://blog.gfi.es/super-heroes-en-la-web-con-angularjs>, fecha de consulta julio 2017.
- [6] B. A. Syed, *Beginning Node.js*. Berkeley, CA: Apress, 2014.
- [7] R. García, Construyendo un servicio Web de MongoDB y Node.js, <https://msdn.microsoft.com/es-es/magazine/dn754378.aspx>, fecha de consulta julio 2017.
- [8] B. Green y S. Seshadri, *AngularJS*, O'Reilly Media, 2013.

- [9] A. Romero, Crear un framework en PHP5, <http://www.suenyos.com/2011/07/20/crear-un-framework-en-php5-desde-cero-1^a-parte>, fecha de consulta julio 2017.
- [10] S. Chauhan, Understanding Data Binding in AngularJS, <http://www.dotnettricks.com/learn/angularjs/understanding-data-binding-in-angularjs>, fecha de consulta julio 2017.
- [11] D. Hows, P. Membrey, y E. Plugge, *MongoDB Basics*. Berkeley, CA: Apress, 2014.
- [12] M. Wall, Estamos listos para la revolución de los grandes datos, http://www.bbc.com/mundo/noticias/2014/03/140304_big_data_grandes_datos_rg, fecha de consulta julio 2017.
- [13] M. Martinez, Joint Application Design (JAD), <https://prezi.com/twuzzym3th5p/joint-application-design-jad>, fecha de consulta julio 2017.
- [14] S. Ramírez Rojas, CASOS DE USO: Definición, <http://exposicioncasosdeusouml.blogspot.com/2009/03/definicion.html>, fecha de consulta julio 2017.
- [15] A. Lopez, C. Quijada, L. Medina, L. Monsalve, y R. Milano, ANALISIS Y DISEÑO DE SISTEMAS, <http://www.oocities.org/es/monsalvelaura/fase2/analisis.html>, fecha de consulta noviembre 2017.
- [16] R. G. Figueroa, C. J. Solís, y A. A. Cabrera, Metodologías tradicionales

vs. Metodologías ágiles, *Universidad Técnica Particular de Loja*, 2008.

- [17] F. Pulido, Scrum y Scrum de Scrums (I), <http://www.franciscojavierpulido.com/2014/01/scrum-scrum-de-scrums-i.html>, fecha de consulta noviembre 2017.
- [18] R. Moya, MEAN (Mongo-Express-Angular-Node) Ejemplo de Aplicación Web (Parte II), <https://jarroba.com/mean-mongo-express-angular-node-ejemplo-de-aplicacion-web-parte-ii>, fecha de consulta noviembre 2017.
- [19] Hackernoon, Meet SurveyJS, <https://hackernoon.com/meet-surveyjs-d8ac6a61db62>, fecha de consulta noviembre 2017.
- [20] R. Pressman, *Ingeniería del software. Un enfoque práctico*. 2010.
- [21] C. De Freitas, J. Benavidez, y C. Irumbe, Calidad, Métricas del Producto y Proceso de Pruebas de Software, <http://uptaprocesodepruebasycalidadymetricas.blogspot.com/2012/12/ejemplos-de-metricas.html>, fecha de consulta noviembre 2017.

ANEXO N°1

Informe JAD

Jipijapa, 7 de julio del 2017

El presente informe trata sobre la reunión realizada entre el Lcdo. Fulco Pincay y el equipo de desarrollo. Para la toma de requerimientos donde se va a realizar la aplicación para la Unidad Educativa Redemptio.

La institución desea una aplicación web para el proceso de seguimiento a graduados, para lo cual se definió lo siguiente:

1	Autenticación al Sistema
2	Registro de Usuarios al Sistema
3	Edición de Usuarios del Sistema
4	Actualización Datos Egresados
5	Elaboración de Encuesta
6	Aplicación de Encuesta
7	Actualización Datos Empleadores
8	Creación de Ofertas Laborales
9	Aplicación de Ofertas Laborales
10	Publicación de Ofertas Laborales
11	Visualización Datos de Encuestas

La institución dentro de su infraestructura tecnológica cuenta con varios servidores equipados con sistemas operativos Linux(Debian), por lo tanto desea que la solución esté basada en tecnologías open source. Para la finalización de este documento constan las firmas del representante de la unidad educativa y el equipo de desarrollo.

Unidad Educativa Redemptio
Lcdo. Fulco Pincay

Equipo de Desarrollo
Ing. Andrés Granja

ANEXO N°2

Instrumento #1 – Usabilidad		Iteración:
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente		
Preguntas		Resultado
¿Cómo califica la manera de manipular la aplicación?		
¿Cómo califica la interfaz de la aplicación?		
¿Cómo califica la forma de aprender a usar la aplicación?		
¿Cómo califica la capacidad de entender el sistema?		

ANEXO N°3

Instrumento #2 – Seguridad	
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente	
Preguntas	Resultado
¿Cómo califica la seguridad de la aplicación?	
¿Cómo califica la forma de ingresar al sistema?	
¿Cómo califica la manera como se visualiza la información personal del usuario en la aplicación?	

ANEXO Nº4

Instrumento #3 – Mantenibilidad	Iteración:
Colocar un valor en cada pregunta, utilizando la siguiente escala: 1-Terrible, 2-Malo, 3-Normal, 4-Bueno, 5-Excelente	
Preguntas	Resultado
¿Cómo califica la capacidad de hacer cambios en la aplicación?	
¿Cómo califica la estabilidad de la aplicación?	
¿Cómo califica la forma de probar las funciones de la aplicación?	

ANEXO N°5

Encuesta a Estudiantes egresados de la Unidad Educativa

Redemptio de la ciudad de Jipijapa.

1 ¿Cómo le resulta la navegación o interacción con la aplicación de seguimiento a egresados?

- Muy sencilla
- Relativamente sencilla
- Normal
- Algo complejo
- Muy compleja

2 ¿Cómo le resulta la actualización de su información en aplicación de seguimiento a egresados?

- Muy fácil
- Relativamente fácil
- Normal
- Relativamente difícil
- Muy difícil

3 ¿Cómo calificas el aspecto de nuestra aplicación web?

- Muy bueno
- Normal
- Peor que la media
- No me gusta nada

4 ¿Cómo te resultó la búsqueda de ofertas laborales a través de nuestra aplicación web?

- Muy sencilla
- Sencilla
- Normal
- Algo complicado
- Muy difícil

5 ¿Se siente satisfecho(a) con nuestra aplicación web?

- Muy satisfecho(a)
- Satisfecho(a)
- Medianamente satisfecho(a)
- Insatisfecho(a)
- Muy insatisfecho(a)

6 ¿Recomendarías nuestra aplicación de seguimiento a egresados a otras personas?

Sí, definitivamente

Probablemente sí

No lo sé

Probablemente no

No, para nada