

# **ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA DE SISTEMAS**

**ANÁLISIS COMPARATIVO DE LOS FRAMEWORKS ADOBE FLEX,  
JAVA RICH FACES Y EXTJS PARA EL DESARROLLO DE  
APLICACIONES ENRIQUECIDAS EN INTERNET (RIA)**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

**PABLO RAMIRO CORREA LOZANO  
pabloc\_mh@hotmail.com**

**DIRECTOR: ING. CARLOS XAVIER CEVALLOS TERÁN  
xavicevallos@hotmail.com**

**Quito, marzo de 2010**

## **DECLARACIÓN**

Yo Pablo Ramiro Correa Lozano, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido en la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad constitucional vigente.

---

Pablo Ramiro Correa Lozano

## **CERTIFICACIÓN**

Certifico que el presente trabajo fue desarrollado por Pablo Ramiro Correa Lozano bajo mi supervisión.

---

Ing. Carlos Cevallos  
**DIRECTOR DEL PROYECTO**

## **AGRADECIMIENTOS**

*A Dios por mantenerme en esta vida llenándome de fortaleza y bendiciones cada día.*

*A mis padres y mi hermana quienes en cada paso han estado junto a mí con su amor y apoyo incondicional.*

*Y a todos aquellos que de una y otra manera me brindaron su apoyo y ayuda para la culminación de mi tesis.*

*PABLO*

## DEDICATORIA

*A mi familia por su apoyo incondicional, cada logro mío es suyo también.*

*A todas aquellas personas que son parte de mi vida, que han compartido su tiempo, paciencia y dedicación durante el desarrollo de mis estudios.*

*“E assurdo pensare che a volte le cose non vadano bene e vadano non rese”*

*Tiziano Ferro*

## Contenido

CAPITULO 1. FRAMEWORKS DE DESARROLLO PARA APLICACIONES RIA.	1
1.1 CARÁCTERÍSTICAS DE LAS APLICACIONES RIA.....	1
1.2 FRAMEWORKS DE DESARROLLO.....	4
1.2.1 ADOBE FLEX.....	4
1.2.2 JAVA RICH FACES.....	11
1.2.3 Ext Js .....	16
CAPITULO 2. ANÁLISIS DE LOS FRAMEWORKS DE DESARROLLO PARA APLICACIONES RIA.....	23
2.1 DEFINICIÓN DE CRITERIOS DE EVALUACIÓN DE LOS FRAMEWORKS .....	23
2.1.1 CRITERIOS DE EVALUACIÓN CONSIDERADOS .....	23
2.1.2 PONDERACIÓN DE CRITERIOS .....	33
2.1.3 DEFINICIÓN DEL CASO DE APLICACIÓN.....	39
2.1.4 MATRIZ DE PRIORIZACIÓN.....	45
2.2 ANÁLISIS DE LAS CARACTERÍSTICAS DE ADOBE FLEX.....	46
2.2.1 APARIENCIA EN LOS DIFERENTES NAVEGADORES .....	46
2.2.2 INDEPENDENCIA DE LA RESOLUCIÓN.....	47
2.2.3 REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA EL DESARROLLO.....	47
2.2.4 COSTOS.....	48
2.2.5 PATRONES DE DISEÑO.....	48
2.2.6 ESTANDARIZACIÓN DEL LENGUAJE .....	48
2.2.7 SOPORTE DEL PRODUCTO .....	49
2.2.8 ROAD MAP TECNOLÓGICO.....	49

2.2.9 TIPO DE LICENCIAMIENTO .....	50
2.2.10 CURVA DE APRENDIZAJE .....	50
2.2.11 HERRAMIENTAS DE DESARROLLO .....	51
2.2.12 FACILIDAD DE INTERACCIÓN CON OTRAS TECNOLOGÍAS .....	51
2.2.13 FACILIDAD DE PUESTA EN PRODUCCIÓN .....	52
2.3 ANÁLISIS DE LAS CARACTERÍSTICAS DE JAVA RICH FACES.....	52
2.3.1 INDEPENDENCIA DE LA RESOLUCIÓN.....	52
2.3.2 APARIENCIA EN LOS DIFERENTES NAVEGADORES .....	52
2.3.3 REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA EL DESARROLLO.....	53
2.3.4 COSTOS .....	54
2.3.5 PATRONES DE DISEÑO .....	54
2.3.6 ESTANDARIZACIÓN DEL LENGUAJE .....	54
2.3.7 SOPORTE DEL PRODUCTO .....	55
2.3.8 ROAD MAP TECNOLÓGICO.....	55
2.3.9 TIPO DE LICENCIAMIENTO .....	55
2.3.10 CURVA DE APRENDIZAJE .....	55
2.3.11 HERRAMIENTAS PARA EL DESARROLLO .....	56
2.3.12 FACILIDAD DE INTERACCIÓN CON OTRAS TECNOLOGÍAS .....	56
2.3.13 FACILIDAD DE PUESTA EN PRODUCCIÓN.....	56
2.4 ANÁLISIS DE LAS CARACTERÍSTICAS DE EXT JS .....	57
2.4.1 APARIENCIA EN LOS DIFERENTES NAVEGADORES .....	57
2.4.2 INDEPENDENCIA DE LA RESOLUCIÓN.....	57

2.4.3 REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA EL DESARROLLO.....	57
2.4.4 COSTOS .....	58
2.4.5 PATRONES DE DISEÑO .....	58
2.4.6 ESTANDARIZACIÓN DEL LENGUAJE .....	58
2.4.7 SOPORTE DEL PRODUCTO .....	59
2.4.8 ROAD MAP TECNOLÓGICO.....	59
2.4.9 TIPO DE LICENCIAMIENTO .....	59
2.4.10 CURVA DE APRENDIZAJE .....	60
2.4.11 HERRAMIENTAS PARA EL DESARROLLO .....	60
2.4.12 FACILIDAD DE INTERACCIÓN CON VARIAS TECNOLOGÍAS .....	60
2.4.13 FACILIDAD DE PUESTA EN PRODUCCIÓN .....	61
CAPITULO 3. DESARROLLO DE PROTOTIPOS.....	62
3.1 PROTOTIPO USANDO ADOBE FLEX.....	62
3.1.1 REGISTRAR PEDIDO DE EXAMEN.....	62
3.1.2 REGISTRAR PACIENTE.....	64
3.1.3 INGRESAR RESULTADOS DE EXAMEN.....	65
3.1.4 CONSULTAR RESULTADOS DE EXAMEN .....	67
3.2 PROTOTIPO USANDO JAVA RICH FACES.....	69
3.2.1 REGISTRAR PEDIDO DE EXAMEN .....	69
3.2.2 REGISTRAR PACIENTE .....	72
3.2.3 INGRESAR RESULTADOS DE EXAMEN.....	73
3.2.4 CONSULTAR RESULTADOS DE EXAMEN .....	75

3.3	PROTOTIPO USANDO EXT JS.....	77
3.3.1	REGISTRAR PEDIDO DE EXAMEN.....	77
3.3.2	REGISTRAR PACIENTE.....	80
3.3.3	INGRESAR RESULTADOS DE EXAMEN.....	81
3.3.4	CONSULTAR RESULTADOS DE EXAMEN .....	82
	CAPITULO 4. ANÁLISIS COMPARATIVO.....	84
4.1	EVALUACIÓN DE PROTOTIPOS.....	84
4.1.1	AMBIENTE PARA EL PROCESO DE DESARROLLO Y EVALUACIÓN.....	84
4.1.2	HERRAMIENTAS PARA LA EVALUACIÓN.....	85
4.1.3	AMBIENTE PARA LA EVALUACIÓN.....	86
4.1.4	EVALUACIÓN DEL PROTOTIPO DESARROLLADO CON ADOBE FLEX.....	88
4.1.5	EVALUACIÓN DEL PROTOTIPO DESARROLLADO CON RICH FACES.....	97
4.1.6	ANÁLISIS DE LAS CARACTERÍSTICAS DE EXT JS .....	104
4.2	COMPARACIÓN DE FRAMEWORKS .....	111
4.3	ANÁLISIS DE RESULTADOS.....	113
	CAPITULO 5. CONCLUSIONES Y RECOMENDACIONES.....	115
5.1	CONCLUSIONES.....	115
5.2	RECOMENDACIONES .....	116
	REFERENCIAS.....	117
	ANEXOS .....	120

## LISTA DE TABLAS

TABLA 1. PRINCIPALES EMPRESAS QUE USAN ADOBE FLEX.....	10
TABLA 2. PRINCIPALES EMPRESAS QUE USAN RICH FACES .....	16
TABLA 3. PRINCIPALES EMPRESAS QUE USAN EXT JS.....	22
TABLA 4. RESUMEN DE CRITERIOS DE EVALUACIÓN .....	24
TABLA 5. VALORACIÓN APARIENCIA .....	25
TABLA 6. VALORACIÓN INDEPENDENCIA DE RESOLUCIÓN .....	25
TABLA 7. VALORACIÓN RECURSOS HW Y SW .....	27
TABLA 8. VALORACIÓN COSTOS .....	27
TABLA 9. VALORACIÓN PATRONES DE DISEÑO .....	28
TABLA 10. ESTANDARIZACIÓN DE LENGUAJES .....	28
TABLA 11. VALORACIÓN DE SOPORTE.....	29
TABLA 12. VALORACIÓN ROAD MAP TECNOLÓGICO .....	30
TABLA 13. VALORACIÓN TIPO DE LICENCIAMIENTO .....	30
ELABORADO POR: EL AUTOR.....	30
TABLA 14. VALORACIÓN CURVA DE APRENDIZAJE .....	31
TABLA 15. VALORACIÓN HERRAMIENTAS DE DESARROLLO.....	32
TABLA 16. VALORACIÓN INTEGRACIÓN CON OTRAS TECNOLOGÍAS .....	33
TABLA 17. VALORACIÓN DESPLIEGUE DE APLICACIÓN .....	33
TABLA 18. ESCALA DE VALORES DE PREFERENCIA .....	35
TABLA 19. MATRIZ DE COMPARACIÓN.....	36
TABLA 20. MATRIZ DE COMPARACIÓN NORMALIZADA .....	37
TABLA 21. RESUMEN DE MATRIZ DE COMPARACIÓN .....	38
TABLA 22. DESCRIPCIÓN DE ACTORES.....	41
TABLA 23. CARACTERÍSTICAS DE LA MÁQUINA DE DESARROLLO .....	84
TABLA 24. CARACTERÍSTICAS DE LA MÁQUINA GNU/LINUX .....	84
TABLA 25. CARACTERÍSTICAS DE LA MÁQUINA SERVIDOR.....	86
TABLA 26. CARACTERÍSTICAS DE LA MÁQUINA CLIENTE.....	87
TABLA 27. MATRIZ DE PRIORIZACIÓN SIN VALORES DE PREFERENCIA .....	112
TABLA 28. MATRIZ DE PRORIZACIÓN CON VALORES DE PREFERENCIA .....	112

## LISTA DE FIGURAS

FIGURA 1.1. DOMINIO DE LAS APLICACIONES RIA .....	1
FIGURA 1.2. ARQUITECTURAS CLIENTE-SERVIDOR SEGÚN MARC DOMENING.....	3
FIGURA 1.3. ARQUITECTURA DE CAIRNGORM .....	7
FIGURA 1.4. LÍNEA DE PRODUCTOS DE ADOBE PARA APLICACIONES RIA.....	7
FIGURA 1.5. ESTRUCTURA DE COMPONENTES DEL NUCLEO DE RICH FACES .....	13
FIGURA 1.6. ESTRUCTURA DE DIRECTORIOS DE EXT JS .....	20
FIGURA 2.1. EJEMPLO DE ESCALA DE VALORES.....	34
FIGURA 2.2. MATRIZ DE COMPARACIÓN.....	34
FIGURA 2.3. MATRIZ DE COMPARACIÓN NORMALIZADA .....	35
FIGURA 2.4. DIAGRAMA DE CASOS DE USO .....	40
FIGURA 2.5. ESTRUCTURA GENERAL DE UNA MATRIZ DE PRIORIZACIÓN.....	46
FIGURA 3.1 PANTALLA DE INGRESO DE CÉDULA DEL PACIENTE .....	62
FIGURA 3.2 PANTALLA DE DESPLIEGUE DE LISTA DE EXÁMENES.....	62
FIGURA 3.3 PANTALLA DE SELECCIÓN DE EXÁMENES SOLICITADOS.....	63
FIGURA 3.4 PANTALLA DE PEDIDO HA SIDO GUARDADO.....	63
FIGURA 3.5 PANTALLA DE EXCEPCIÓN PACIENTE NO HA SIDO REGISTRADO .....	64
FIGURA 3.6 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE .....	64
FIGURA 3.7 PANTALLA DE PACIENTE GUARDADO.....	65
FIGURA 3.8 PANTALLA DE CAPTURA DE CÓDIGO DE MUESTRA.....	65
FIGURA 3.9 PANTALLA DE LISTA DE EXÁMENES REGISTRADOS .....	66
FIGURA 3.10 PANTALLA DEL FLUJO ALTERNO1 MUESTRA NO ESTÁ REGISTRADA.....	66
FIGURA 3.11. PANTALLA DEL FLUJO ALTERNO2 PACIENTE NO ESTÁ REGISTRADO.....	67
FIGURA 3.12 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE Y NÚMERO DE MUESTRA .....	67
FIGURA 3.13 PANTALLA DE DELPLIEGUE DE RESULTADOS EXAMEN.....	68
FIGURA 3.14. PANTALLA DE MENSAJE “PACIENTE O MUESTRA INCORRECTA” .....	68
FIGURA 3.15 PANTALLA DE DESPLIEGUE DE LISTA DE RESULTADOS SIN VALORES.....	69
FIGURA 3.16 PANTALLA DE INGRESO DE CÉDULA DEL PACIENTE .....	69
FIGURA 3.17 PANTALLA DE DESPLIEGUE DE LISTA DE EXÁMENES.....	70
FIGURA 3.18. PANTALLA DE SELECCIÓN DE EXÁMENES SOLICITADOS.....	70
FIGURA 3.19. PANTALLA DE MENSAJE “PEDIDO HA SIDO GUARDADO” .....	71
FIGURA 3.20 PANTALLA DE EXCEPCIÓN PACIENTE NO HA SIDO REGISTRADO .....	71
FIGURA 3.21 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE .....	72
FIGURA 3.22 PANTALLA DE PACIENTE GUARDADO.....	72
FIGURA 3.23 PANTALLA DE CAPTURA DE CÓDIGO DE MUESTRA.....	73

FIGURA 3.24 PANTALLA DE LISTA DE EXÁMENES REGISTRADOS .....	73
FIGURA 3.25 PANTALLA DE ASIGNAR VALORES A LOS EXAMENES SOLICITADOS .....	74
FIGURA 3.26 PANTALLA DEL FLUJO ALTERNO1 MUESTRA NO ESTÁ REGISTRADA.....	74
FIGURA 3.27 PANTALLA DEL FLUJO ALTERNO2 PACIENTE NO ESTÁ REGISTRADO.....	75
FIGURA 3.28. PANTALLA DE CAPTURA DE DATOS DEL PACIENTE .....	75
FIGURA 3.29 PANTALLA DE DESPLIEGUE DE EXAMEN.....	76
FIGURA 3.30 PANTALLA DE MENSAJE “PACIENTE O MUESTRA INCORRECTA” .....	76
FIGURA 3.31 PANTALLA DE DESPLIEGUE DE LISTA DE RESULTADOS SIN VALORES.....	77
FIGURA 3.32 PANTALLA DE INGRESO DE CÉDULA DEL PACIENTE .....	77
FIGURA 3.33 PANTALLA DE DESPLIEGUE DE LISTA DE EXÁMENES.....	78
FIGURA 3.34 PANTALLA DE SELECCIÓN DE EXÁMENES SOLICITADOS.....	78
FIGURA 3.35 PANTALLA DE PEDIDO HA SIDO GUARDADO.....	79
FIGURA 3.36 PANTALLA DE EXCEPCIÓN PACIENTE NO HA SIDO REGISTRADO .....	79
FIGURA 3.37 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE .....	80
FIGURA 3.38 PANTALLA DE PACIENTE GUARDADO.....	80
FIGURA 3.39 PANTALLA DE CAPTURA DE CÓDIGO DE MUESTRA.....	81
FIGURA 3.40 PANTALLA DE LISTA DE EXÁMENES REGISTRADOS .....	81
FIGURA 3.41 PANTALLA DEL FLUJO ALTERNO1 MUESTRA NO ESTÁ REGISTRADA.....	82
FIGURA 3.42 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE Y NÚMERO DE MUESTRA .....	82
FIGURA 3.43 PANTALLA DE DELPLIEGUE DE RESULTADOS EXAMEN.....	83
FIGURA 4.1 DIAGRAMA DE RED EVALUACIÓN .....	86
FIGURA 4.2 PROTOTIPO DE ADOBE FLEX EN INTERNET EXPLORER.....	88
FIGURA 4.3 PROTOTIPO DE ADOBE FLEX EN SAFARI .....	89
FIGURA 4.4 PROTOTIPO DE ADOBE FLEX EN MOZILLA FIREFOX .....	89
FIGURA 4.5 PROTOTIPO DE ADOBE FLEX EN GOOGLE CHROME.....	90
FIGURA 4.6 PROTOTIPO DE ADOBE FLEX A RESOLUCIÓN DE 1024X 768 PÍXELES.....	91
FIGURA 4.7 PROTOTIPO DE ADOBE FLEX A RESOLUCIÓN DE 1280 X 768 PÍXELES .....	91
FIGURA 4.8 PROTOTIPO DE ADOBE FLEX A RESOLUCIÓN DE 1280 X 800 PÍXELES .....	92
FIGURA 4.9 FLEX BUILDER INSTALADO EN UN AMBIENTE WINDOWS.....	93
FIGURA 4.10 INSTALACIÓN DE FLEX BUILDER EN UN AMBIENTE GNU/LINUX.....	94
FIGURA 4.11 FLEX BUILDER INSTALADO EN UN AMBIENTE GNU/LINUX .....	94
FIGURA 4.12 PROTOTIPO DE RICH FACES EN SAFARI .....	98
FIGURA 4.13 PROTOTIPO DE RICH FACES EN MOZILLA FIREFOX.....	98
FIGURA 4.14 PROTOTIPO DE RICH FACES EN GOOGLE CHROME .....	99
FIGURA 4.15 PROTOTIPO DE RICH FACES EN INTERNET EXPLORER V7 .....	99
FIGURA 4.16 PROTOTIPO DE RICH FACES A RESOLUCIÓN DE 1024 X 768 PÍXELES .....	100
FIGURA 4.17 PROTOTIPO DE RICH FACES A RESOLUCION DE 1280 X 768 PÍXELES .....	101

FIGURA 4.18	PROTOTIPO DE RICH FACES A RESOLUCIÓN DE 1280 X 800 PÍXELES .....	101
FIGURA 4.19	PROTOTIPO DE EXT JS EN MOZILLA FIREFOX.....	105
FIGURA 4.20	PROTOTIPO DE EXT JS EN GOOGLE CHROME.....	105
FIGURA 4.21	PROTOTIPO DE EXT JS EN SAFARI.....	106
FIGURA 4.22	PROTOTIPO DE EXT JS EN SAFARI.....	106
FIGURA 4.23	PROTOTIPO DE EXT JS A RESOLUCIÓN DE 1280 X 768.....	107
FIGURA 4.24	PROTOTIPO DE EXT JS A RESOLUCIÓN DE 1280 X 768.....	107
FIGURA 4.25	PROTOTIPO DE EXT JS A RESOLUCIÓN DE 1280 X 800.....	108
FIGURA A2.1	CONFIGURACIÓN DEL ESCENARIO DE PRUEBAS WAPT PASO 1.....	121
FIGURA A2.2	CONFIGURACIÓN DEL ESCENARIO DE PRUEBAS WAPT PASO 2.....	122
FIGURA A2.3	CONFIGURACIÓN DEL ESCENARIO DE PRUEBAS WAPT PASO 3.....	122
FIGURA A2.4	RESUMEN Y FINALIZACIÓN DE CONFIGURACIÓN DE LA PRUEBA .....	123
FIGURA A2.5	ADMINISTRADOR DE REDES DE NET LIMITER .....	123
FIGURA A2.6	CONFIGURACIÓN DE ANCHO DE BANDA .....	124
FIGURA A2.7	PANTALLA PRINCIPAL DE PERFORMANCE MONITOR .....	124
FIGURA A2.8	SELECCIÓN DE CONTADORES PARA LA INTERFAZ DE RED .....	125
FIGURA A2.9	SELECCIÓN DE CONTADORES PARA LA INTERFAZ DE RED .....	126
FIGURA A2.10	RESUMEN DEL REGISTRO CREADO .....	126

## INTRODUCCIÓN

Debido a la creciente interacción de usuarios con sistemas web, surge la necesidad de combinar las funcionalidades e interfaces de usuario de las aplicaciones clásicas de escritorio, con la accesibilidad y bajo costo de publicación de las aplicaciones basadas en la Web; dando origen a las Aplicaciones Enriquecidas en Internet (RIA).

En el presente Proyecto de Titulación se establece un conjunto de criterios que permiten evaluar frameworks de desarrollo de aplicaciones RIA, en base a estos criterios se valoró los frameworks Adobe Flex, Java Rich faces y ExtJS y se identificó fortalezas y debilidades de cada uno de ellos tanto a nivel teórico como práctico a través del desarrollo de prototipos. Los resultados obtenidos permitirán seleccionar el framework más idóneo considerando las necesidades específicas que una organización tenga para desarrollar aplicaciones RIA.

En el Capítulo Uno se realiza una descripción de las generalidades de las aplicaciones RIA así como de cada uno de los frameworks, considerando aspectos técnicos y no técnicos que permiten conocer el framework de manera descriptiva.

En el Capítulo Dos se definen los criterios referenciales y su forma de evaluación. Se definen como mecanismos para determinar el framework más apto el uso de la matriz de priorización y la matriz de comparación.

Posteriormente se define el caso de aplicación que permitirá un análisis profundo de rendimiento de los componentes de cada framework.

El Capítulo Tres describe la implementación del caso de aplicación con cada uno de los frameworks.

El Capítulo Cuatro permite realizar la evaluación, basada en los criterios definidos en el capítulo dos y la experiencia obtenida durante el proceso de implementación del capítulo tres, para cada criterio se obtienen valoraciones finales que permiten determinar la conveniencia de los frameworks para el caso de aplicación escogido.

Finalmente, en el Capítulo Cinco se presentan las conclusiones y recomendaciones, resultado del trabajo desarrollado.

# CAPITULO 1. FRAMEWORKS DE DESARROLLO PARA APLICACIONES RIA

## 1.1 CARÁCTERÍSTICAS DE LAS APLICACIONES RIA

Las aplicaciones RIA (Rich Internet Applications) nacen de la convergencia de dos paradigmas, las aplicaciones de escritorio y las aplicaciones web; gracias a la evolución de ambos cada día se hace más difícil distinguir entre uno y otro.

Las tecnologías comprendidas por la definición de RIA, varían en un rango amplio entre las aplicaciones tradicionales de escritorio y las simples páginas estáticas, tal como se muestra en la figura 1.1.

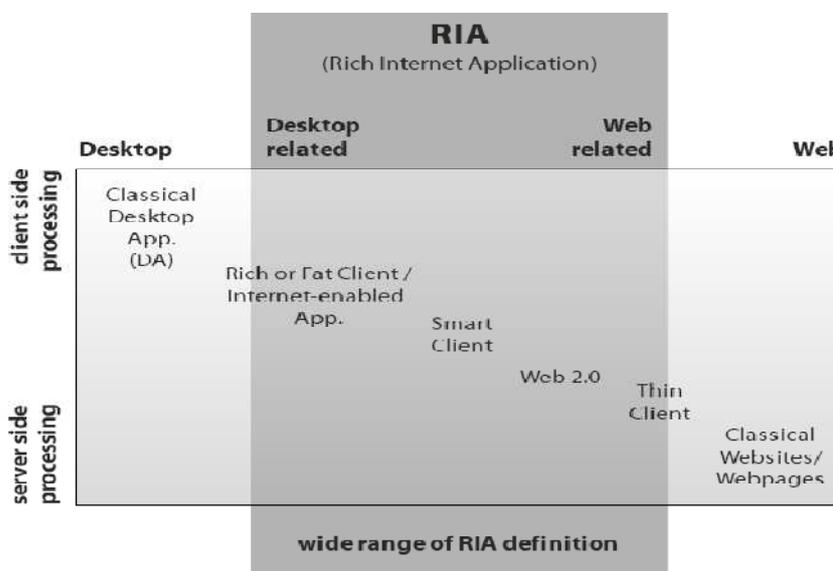


FIGURA 1.1. DOMINIO DE LAS APLICACIONES RIA

Fuente: <http://www.flomedia.de/diploma/>.

Se puede notar, de esta manera, que es posible determinar que existen aplicaciones RIA que corren sobre un navegador de Internet y otras no. Las aplicaciones RIA basadas en un navegador son sitios web que mejoran la experiencia de los usuarios aumentando facilidades de personalización, tiempo de respuesta y riqueza multimedia, como ejemplos de estas tenemos Facebook y Google Maps.

Las aplicaciones RIA que no necesitan de un navegador, gracias a su apariencia y gran cantidad de contenidos, hacen que el usuario las perciba como aplicaciones de escritorio, el principal ejemplo de este tipo de aplicaciones es el

reproductor de audio y video Apple iTunes que además incluye iTunes Store. iTunes puede ser visto como una aplicación de escritorio por la necesidad de una instalación local, pero puede ser vista como una aplicación web que actúa como un sitio de comercio electrónico.

Aunque existen varias tecnologías y aplicaciones que pueden ser consideradas dentro de la categoría de aplicaciones RIA dependiendo del punto de vista, la característica común es que una aplicación RIA junta las bondades de las aplicaciones de escritorio con los beneficios de las aplicaciones web.

De las aplicaciones web se toman las siguientes características:

- Tiempo corto para la obtención de un producto final gracias a la estandarización de scripts y etiquetas
- Las aplicaciones que corren sobre un servidor no necesitan instalación de parches ni actualizaciones.
- Independencia de la plataforma desde la cual son accedidas
- Alta disponibilidad
- Interfaz simple y estandarizada disminuyendo la curva de aprendizaje.

De las aplicaciones de escritorio se toman las siguientes características:

- Gran contenido multimedia (audio, video).
- No existe la necesidad de recargar la aplicación, (recarga de página).
- Soporte on line y off line.
- Mayor complejidad en las aplicaciones.
- Gran familiaridad con los usuarios.

Se considerará ahora los aspectos fundamentales que distinguen a las aplicaciones RIA:

1. Proporcionan un gran rendimiento, esto debido a que el procesamiento se puede hacer en el lado del cliente, en lugar que todos los procesos se realicen en el lado del servidor. De igual manera mientras se usa la aplicación el usuario no experimenta ninguna clase de refresco de página.

Sin embargo la decisión de que procesos se realizan en la parte del cliente y que en la parte del servidor, dependen de varios aspectos, en la figura 1.2 se puede

notar la separación de las capas de una aplicación, junto con la distribución de procesamiento.

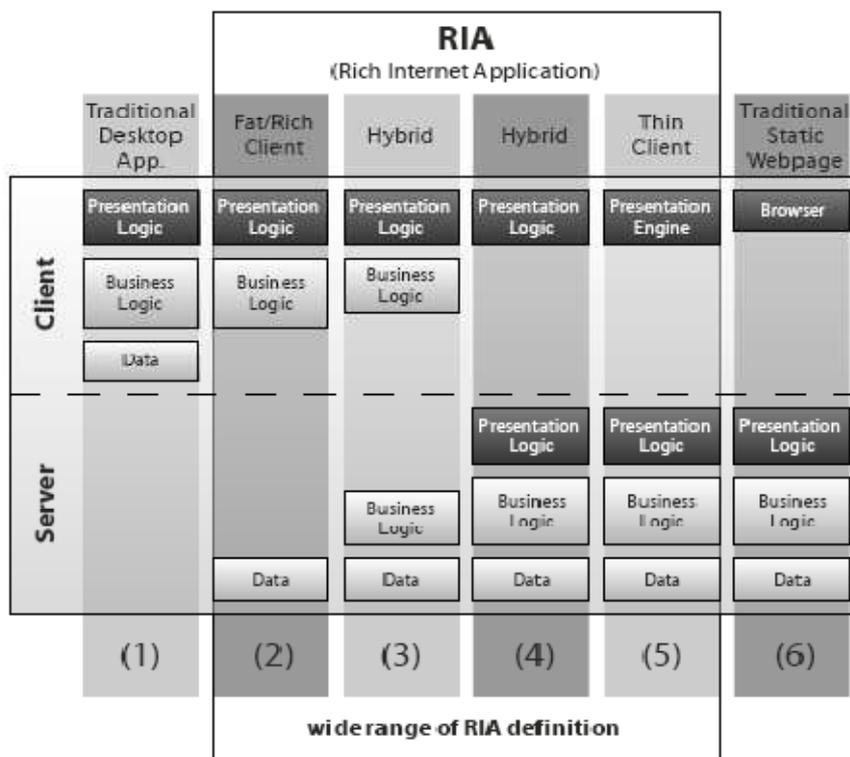


FIGURA 1.2. ARQUITECTURAS CLIENTE-SERVIDOR SEGÚN MARC DOMENING

Fuente: <http://www.flomedia.de/diploma/>.

2. Priorizan la interacción con el usuario, por la gran riqueza de recursos proporcionan una experiencia más atractiva y entretenida.

A continuación se describen algunos ejemplos:

- Gracias a las herramientas gráficas para la visualización de datos, permite al nivel gerencial interactuar con datos complejos y realizar análisis más profundos.
- Permite a los usuarios personalizar un producto en línea para satisfacer sus necesidades específicas.

3. Acoplan de manera natural varias tecnologías para conseguir la aplicación final que cumpla con las necesidades del usuario. Esto se hace sin olvidar la consigna de independencia de plataforma de acceso del usuario. En este punto es necesario considerar que el uso de varias tecnologías versus el uso de una sola, determina que la complejidad y tiempo de desarrollo sufren un claro incremento.

## **1.2 FRAMEWORKS DE DESARROLLO**

### **1.2.1 ADOBE FLEX**

Adobe Flex es un framework compuesto por un conjunto de tecnologías encaminadas al desarrollo de aplicaciones RIA. Estas tecnologías son Adobe Flash, ActionScript y AIR (Adobe Integrated Runtime)

El release inicial se dio en Marzo de 2004 de la mano de Macromedia, en este se incluía el SDK (Software Development Kit), el IDE (Integrated Development Environment) y una aplicación para la integración con J2EE a la cual denominaron Flex Data Services. Macromedia mantuvo el producto en el mercado hasta la versión 1.5 con el nombre de Macromedia Flex Server. Desde entonces Flash Player fue el elegido para poder visualizar las aplicaciones en el navegador.

Cuando Adobe compró Macromedia continuó con el proyecto, siendo el primer producto de Macromedia que fue renombrado por Adobe. En la versión 2 del nuevo Adobe Flex se cambio de manera considerable el sistema de licenciamiento así como el IDE para el desarrollo, Flex Builder, el cual se baso en la plataforma de código abierto Eclipse.

El cambio más significativo para la versión 2, fue la inclusión del renovado ActionScript, conocido ahora como ActionScript3, como lenguaje de programación, se mantuvo sin embargo la integración con J2EE mediante Flex Data Services.

El último release en el mercado es la versión Adobe Flex 3, la cual incluye un IDE desarrollado completamente sobre Eclipse y varios cambios importantes, el primero de estos fue la liberación del código del SDK mediante la licencia pública de Mozilla. El segundo fue el reemplazo de Flex Data Service por LiveCycle Data Services. El cambio final es la integración con AIR lo cual permite realizar aplicaciones RIA de manera tal que estas se ejecuten de manera directa sobre el sistema operativo sin la necesidad de un navegador. De esta manera se extendió el campo de acción que estaba limitado al uso de un navegador con Adobe Flash Player.

### **1.2.1.1 MXML (Multimedia Extensible Markup Language)**

MXML es un lenguaje basado en XML (eXtensible Markup Language) y se halla estructurado en base a etiquetas, permite la creación de interfaces de usuario para aplicaciones RIA.

Las etiquetas de MXML son muy similares a las del conocido HTML (HyperText Markup Language), sin embargo contemplan un marco más amplio de etiquetas como: soporte nativo para representación de cubos de datos, grids o tablas de datos, arboles, etc. De manera particular la ventaja de MXML es la posibilidad de crear componentes personalizados basados en los componentes básicos.

### **1.2.1.2 ActionScript**

Se trata de un lenguaje de programación orientado a objetos, diseñado en sus inicios para la creación de aplicaciones Flash dinámicas, su objetivo fue facilitar el proceso de desarrollo de las mismas. ActionScript fue concebido para correr sobre su propia máquina virtual, la cual se incluye en el reproductor Flash Player.

Dentro del framework de Adobe para aplicaciones RIA, Flex, se incluye la tercera versión de este lenguaje, la cual provee una versión mejorada de la máquina virtual de ActionScript la cual aumenta el rendimiento de las aplicaciones Flash.

ActionScript 3.0 adopta el estándar ECMA-226 <sup>1</sup>, el cual es la base para los lenguajes de Scripting. Un lenguaje de scripting permite manipular, personalizar y ejecutar cálculos usando las funcionalidades ya disponibles en un sistema en el ambiente de host. Con la implementación del estándar, ActionScript incluye características como el manejo de excepciones mediante las cláusulas try/catch, mejor manejo de strings y de errores.

### **1.2.1.3 Patrones de Diseño**

Para el desarrollo de aplicaciones RIA utilizando Adobe Flex, existen varias opciones que permiten aplicar el patrón de diseño MVC (Model View Controller); sin embargo dentro de estas opciones, la micro arquitectura Cairngorm es la más

---

<sup>1</sup> A partir de 1994 la *European Computer Manufacturers Association (ECMA)* pasó a denominarse Ecma International con lo cual ECMA deja de considerarse un acrónimo.

difundida dentro de la comunidad de desarrolladores, esto debido a que Adobe no ha adoptado un patrón de diseño de manera formal.

La descripción de los componentes de la micro arquitectura Cairngorm como arquitectura se describirá en la sección 1.2.1.4

#### 1.2.1.4 Descripción de componentes

Cairngorm es un proyecto open source mantenido por Adobe, en sus inicios fue desarrollado por la empresa Iteration Two basado en patrones arquitectónicos de J2EE, actualmente quienes emplean y desarrollan Cairngorm lo consideran como “un framework arquitectónico, compuesto por un conjunto de patrones de diseño que provee de un punto genérico de partida para la arquitectura técnica de cualquier aplicación.”<sup>2</sup>

Cairngorm define la arquitectura de una aplicación de la siguiente manera:

- **Front Controller:** Componentes que se encuentran en escucha de las peticiones de usuario y que mapea las mismas a un componente command.
- **Command:** Componentes que permiten procesar los eventos de usuario y permite la actualización de los Value Objects.
- **Value Object:** Componentes que se conservan de los patrones J2EE y que permiten el manejo de los objetos del negocio, sus atributos conservan el estado en todos los lugares desde donde son invocados, esto se debe a su propiedad bindable. El conjunto de value objects suele encontrarse dentro de un contenedor que recibe el nombre Model Locater.
- **Delegate:** Utilizados por los command para eventos determinados y que permiten invocar el componente service adecuado, así como los parámetros necesarios para la invocación.

---

<sup>2</sup> Developing Flex RIAs with Cairngorm microarchitecture – Part 1: Introducing Cairngorm  
[http://www.adobe.com/devnet/flex/articles/cairngorm\\_pt1\\_02.html](http://www.adobe.com/devnet/flex/articles/cairngorm_pt1_02.html)

- **Service:** Componente constituido por todos los servicios remotos, sean estos WebServices, LiveCycle Data Services, Enterprise JavaBeans o ColdFusion components.

La figura 1.3 describe el uso de cada uno de los componentes de la arquitectura Cairngorm en una aplicación.

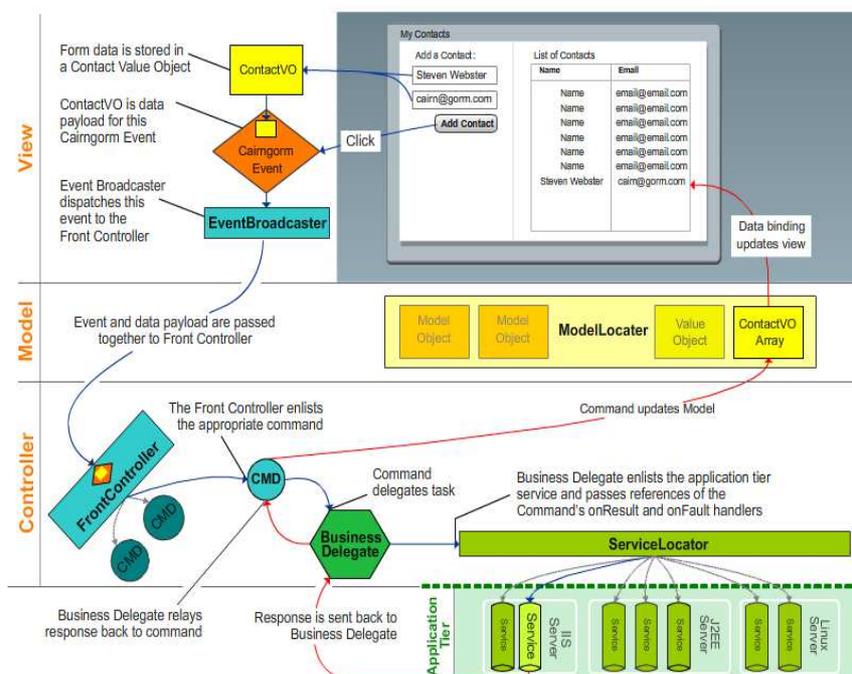


FIGURA 1.3. ARQUITECTURA DE CAIRNGORM

Fuente: <http://www.cristianhernandez.org/blog/?p=78>

### 1.2.1.5 Plataforma Tecnológica de Desarrollo

Adobe propone una gama de productos para el desarrollo y la puesta a producción de aplicaciones RIA, esta se encuentra descrita en la figura 1.4



FIGURA 1.4. LÍNEA DE PRODUCTOS DE ADOBE PARA APLICACIONES RIA

Fuente: Creating rich Internet applications with the Adobe technology platform. ([www.adobe.com](http://www.adobe.com))

Los productos servidores corren sobre las plataformas:

- IBM AIX®
- Red Hat® Enterprise GNU/Linux
- SUSE GNU/Linux Enterprise Server
- Sun™ Solaris™
- Microsoft® Windows®

El framework Adobe Flex en la parte de SDK open source puede ejecutarse sobre plataformas GNU/Linux , Windows y Mac, sin embargo el Flex Builder con sus características completas solo se ejecuta sobre Windows y Mac, aunque existe una versión con las características básicas para plataformas GNU/Linux.

Los clientes son multiplataforma, en especial Flash Player que se integra con la totalidad de navegadores de internet, respecto a AIR funciona sobre plataformas Windows, GNU/Linux y Mac.

Los requerimientos para Flex Builder 3 dependen del sistema operativo sobre el cual se va a ejecutar así tenemos

#### Microsoft Windows

- Intel® Pentium® 4 processor
- Microsoft® Windows® XP with Service Pack 2 or Windows Vista® Home (Premium o Basic), Business, o Ultimate
- 1GB de RAM (2GB recomendado)
- 500MB de espacio libre en disco ( 500MB adicionales para la configuración como plug-in)
- Java™ Virtual Machine: Sun™ JRE 1.4.2, Sun JRE 1.5 , IBM® JRE 1.5, o Sun JRE 1.6
- Eclipse™ 3.2.2 (3.3 y 3.4 para la configuración como plug-in).
- Adobe® Flash® Player 9

#### MAC OS

- PowerPC® G4 1.25GHz o Procesador Intel
- Mac OS X v10.4.7–10.4.10 o 10.5
- 1GB de RAM (2GB of RAM recomendado)

- 500MB de espacio libre en disco duro
- Java Virtual Machine: JRE 1.5 or JRE 1.6 de Apple.
- Eclipse 3.2.2 , 3.3 y 3.4 (para configuración como plug-in)
- Adobe Flash Player 9 software

Si se desea utilizar únicamente el SDK se tiene los siguientes requerimientos

- Windows 2000, Windows XP, o Windows Server® 2003.
- Java 1.4 (Sun, IBM, o BEA) o 1.5 (Sun)
- Mac OS X v10.4.x, Java 1.5 ( Apple) PowerPC
- Red Hat® Enterprise GNU/Linux® 3 o 4, SUSE® 10, Java 1.4 (Sun, IBM, or BEA) o 1.5 (Sun)
- Solaris™ 9, 10, Java 1.4 o 1.5 (Sun)
- 512MB de RAM (1GB recomendado).
- 200MB de espacio libre en disco duro.

#### **1.2.1.6 Requerimientos de Hardware y Software de Cliente**

Tal como las aplicaciones que corren sobre internet, las aplicaciones RIA se caracterizan por la independencia de plataforma, sin embargo las aplicaciones desarrolladas con Adobe Flex requieren del siguiente software para funcionar:

- Un navegador de internet que soporte Flash.
- Flash Player
- AIR (para las aplicaciones que no requieren de un navegador).

Respecto al hardware no existen especificaciones.

#### **1.2.1.7 Costos de Licencias**

Como se menciono anteriormente el SDK de Adobe Flex es open source y no tiene ningún costo, la última versión se encuentra disponible para su descarga en [www.adobe.com](http://www.adobe.com).

Sin embargo Adobe Flex Builder versión Standard tiene un costo de 249 dólares y la versión Professional 699 dólares, si se trata de adquirir una nueva licencia mientras la actualización de la versión standard 2 de Flex Builder a la versión 3

tiene un costo de 99 dólares y la actualización de la versión Professional tiene un costo de 299 dólares.<sup>3</sup>

Además Adobe promueve la investigación y tiene una distribución gratuita de Flex Builder para estudiantes.

El soporte para Flex Builder lo brinda Adobe, de manera similar que las certificaciones y entrenamiento. Las certificaciones tienen un costo básico de 150 dólares y el entrenamiento y capacitación tiene varias etapas y modalidades que comprenden cursos en línea y cursos presenciales.

### 1.2.1.8 Principales empresas que utilizan Adobe Flex

Empresa	Aplicación	URL
<b>Discovery Channel</b> 	Discovery crea la aplicación Discovery earthlive en la cual se puede monitorear eventos como erupciones volcánicas, conocer la fauna de determinadas regiones, exploración de la posición de la tierra en el espacio, etc	<a href="http://dsc.discovery.com/discovery-earth-live/">http://dsc.discovery.com/discovery-earth-live/</a>
<b>Honda</b> 	Honda para Europa provee la facilidad a los usuarios de crear su motocicleta, escoger su modelo, personalizar los accesorios y seleccionar los colores	<a href="http://www.buildyourhonda.eu/">http://www.buildyourhonda.eu/</a>
<b>Philips</b> 	Web Site	<a href="http://www.philips.es/">http://www.philips.es/</a>
<b>NASA</b> 	La NASA permite acceder a vistas del clima espacial, en especial de la tierra y el sol	<a href="http://sunearth.gsfc.nasa.gov/spaceweather/FlexApp/bin-debug/index.htm">http://sunearth.gsfc.nasa.gov/spaceweather/FlexApp/bin-debug/index.htm</a>

TABLA 1. PRINCIPALES EMPRESAS QUE USAN ADOBE FLEX

Elaborado por: El autor

<sup>3</sup> Los precios se obtuvieron de <http://www.adobe.com/products/flex/> el 25 de mayo de 2009

### 1.2.2 JAVA RICH FACES

Rich Faces es un framework open source que añade las capacidades de Ajax (Asynchronous JavaScript and XML) en aplicaciones JSF (Java Server Faces) existentes sin la necesidad de usar JavaScript.<sup>4</sup>

Rich Faces se crea a partir del framework Ajax4jsf, el cual fue diseñado y desarrollado por Alexander Smirnov en el año 2005. En 2006 Smirnov formando parte de Exadel, continua con el proyecto y para Marzo se da a conocer la primera versión de Ajax4jsf; mas adelante el mismo año Exadel decide dividir al framework obteniendo de esta manera Ajax4jsf y Rich Faces.

Ajax4jsf se convierte en un proyecto open source, mientras Rich Faces se convierte en una librería de componentes comercial.

En el 2007 JBoss (desde 2006 JBoss es una división de Red Hat) y Exadel firman un contrato de colaboración, donde los proyectos Ajax4jsf y Rich Faces serán mantenidos por JBoss. JBoss convierte a Rich Faces en un proyecto open source, y para septiembre de 2007 por una decisión bilateral se unifican Ajax4jsf y Rich Faces bajo el único nombre de Rich Faces.

JBoss y Red Hat mantienen a Rich Faces como open source hasta la actualidad, sin embargo el soporte técnico y parches se dan mediante la suscripción como desarrollador o como consumidor de los productos en la versión empresarial.

Las diferencia entre el desarrollo de las librerías que componen Rich Faces ocasionó un conjunto de dificultades en la integración, estas dificultades fueron resueltas y la última versión de Rich Faces es la versión 3.3.0 que fue lanzada en enero de 2009.

Rich Faces se integra con JSF mediante el empleo de 2 librerías de componentes, la primera Ajax4jsf, la cual permite integrar funcionalidad Ajax sin la necesidad de escribir código JavaScript, la segunda es la librería de interfaz de

---

<sup>4</sup> Anónimo, Rich Faces Developer Guide [http://www.jboss.org/file-access/default/members/jbossrichfaces/freezone/docs/devguide/en/html\\_single/index.html](http://www.jboss.org/file-access/default/members/jbossrichfaces/freezone/docs/devguide/en/html_single/index.html)

usuario Rich FacesUI diseñada para mejorar las características de JSF en lo referente a interfaz de usuario, permitiendo además una integración con otras librerías de componentes.

#### **1.2.2.1 Lenguaje de Programación**

Rich Faces al integrarse con JSF usa el lenguaje Java para su funcionamiento, la manera en la que lo hace es por medio de tag libraries. Una tag library no es más que un conjunto de clases compiladas en un archivo de extensión .jar, que se incluyen dentro del código HTML mediante la etiqueta `<@taglib uri=" " >`, con la finalidad de extender las capacidades de las JSP (Java Server Pages) e invocar funcionalidad de las clases que conforman la mencionada tag library.

Rich Faces se integra de manera nativa con las implementaciones de la especificación JSF 1.2.

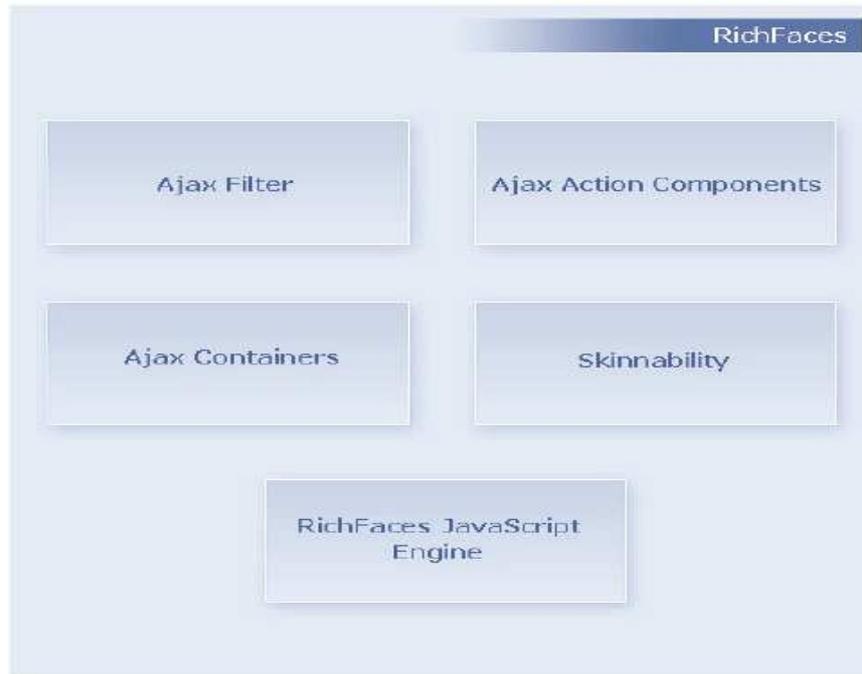
#### **1.2.2.2 Patrones de Diseño**

Debido a que Rich Faces extiende las capacidades de las JSF, implementa el patrón de diseño MVC (Model View Controller), para el patrón de diseño MVC, el Modelo es la representación de la persistencia (datos) de la aplicación, la vista corresponde a la interfaz de usuario, mientras el controlador sirve como nexo entre el modelo y la vista.

Rich Faces como framework se corresponde de manera específica a la sección de Vista -Controlador con las JSP y los Backing Beans.

#### **1.2.2.3 Descripción de componentes**

Como ya se mencionó Rich Faces ofrece el soporte para el uso de Ajax sin el uso de JavaScript, este soporte es núcleo del framework, y se halla conformada por un conjunto de componentes fundamentales que se muestran en la figura 1.4.



**FIGURA 1.5. ESTRUCTURA DE COMPONENTES DEL NUCLEO DE RICH FACES**

**Fuente:** [http://www.jboss.org/file-access/default/members/jbossRichFaces/freezone/docs/devguide/en/html\\_single/index.html](http://www.jboss.org/file-access/default/members/jbossRichFaces/freezone/docs/devguide/en/html_single/index.html)

**Ajax Filter.** Es necesario registrar un filtro para reconocer los diferentes tipos de request que se realizan desde la página web, el registro se hace en el archivo web.xml de nuestra aplicación.

El filtro permite determinar si se trata de un request del tipo Ajax o un request clásico de JSF, dependiendo del tipo se determinará el tipo de respuesta que el servidor envíe al cliente.

**Ajax Action Components.** Se usan para enviar solicitudes o request Ajax desde el cliente.

**Ajax Containers.** Se trata de un componente que permite definir una o varias áreas de la página JSF que debe ser decodificada durante un request Ajax.

**JavaScript Engine.** Es un componente de Rich Faces que corre del lado del cliente y permite actualizar las diferentes áreas de la página JSF basado en la información de la respuesta del servidor.

**Skinnability.** Este componente permite establecer características visuales para los componentes de las páginas JSF, extienden los beneficios de las CSS (Cascade Style Sheets) permitiendo definir estilos visuales para ventanas emergentes, colores para cabeceras de secciones, etc. que no son posibles de lograr con una CSS.

Rich Faces usa Skinnability para facilitar el trabajo de ajustes de efectos visuales sobre la marcha para una interfaz de usuario, cada componente en Rich Faces puede obtener su estilo de un skin predefinido o de una CSS definida por el usuario.

Skinnability no puede reemplazar definitivamente las CSS, ni elimina su uso, más bien pueden ser usadas de manera conjunta para obtener una mejor apariencia visual en lo referente a combinación de colores y fuentes.

#### **1.2.3.4 Plataforma Tecnológica de Desarrollo**

Al momento de elegir una plataforma de desarrollo Rich Faces nos proporciona una amplia gama de opciones, de manera nativa se puede utilizar JBoss Developer Studio, sin embargo se puede integrar mediante el uso de plugins con Eclipse y Netbeans, para lo cual en ambos casos es necesario de las librerías de Rich Faces Rich Faces-api.jar, Rich Faces-impl.jar, Rich Faces-ui.jar.

En el presente trabajo se utilizará Netbeans como IDE para el desarrollo a lo cual se le añadirá el plugin de Rich Faces, los requerimientos son los siguientes:

Netbeans para JEE (requisitos recomendados)

Microsoft Windows XP/ Windows Vista

- Procesador: Intel Pentium IV 2.6 GHz o superior
- Memoria: 1GB
- Espacio en disco: 1GB libre

Ubuntu

- Procesador: Intel Pentium IV 2.6 GHz o superior
- Memoria: 1GB
- Espacio en disco: 850MB libre

Solaris

- Procesador: UltraSPARC IIIi 1 GHz

- Memoria: 1GB
- Espacio en disco: 850MB libre

Macintosh OS X 10.5

- Procesador: Dual Core Intel
- Memoria: 1GB
- Espacio en disco: 850MB libre

### 1.2. 3.5 Requerimientos de Hardware y Software de Cliente

Las aplicaciones desarrolladas con Rich Faces requieren únicamente un navegador web como software de cliente.

Respecto al hardware no existen especificaciones.

### 1.2. 3.6 Costos de Licencias

Todas las licencias de Rich Faces son del tipo LGPL (GNU Lesser General Public License), con lo cual le permiten integrarse en proyectos de software no libre.

Netbeans como IDE de desarrollo es open source por lo cual su licencia no tiene costo.

Glassfish es también un proyecto open source razón por la cual tampoco tiene costo.

### 1.2.3.7 Principales empresas que utilizan Java Rich Faces

Empresa	Aplicación	URL
<b>J4Care</b> 	Se trata de una aplicación médica que muestra información vital de pacientes, resultados de sus exámenes, imágenes de radiografías, tomografías, etc	<a href="http://www.j4care.com/Products/J4Care%20Viewer">http://www.j4care.com/Products/J4Care%20Viewer</a>
<b>tinyPM</b> 	Se trata de una herramienta orientada al proceso de desarrollo de software, permite la captura de requerimientos, planeación de iteraciones, definición y división de tareas para el equipo de desarrollo.	<a href="http://www.tinypm.com/overview">http://www.tinypm.com/overview</a>

<p><b>Zesta Solutions</b></p>  	<p>Javelin es un CRM (Customer Relationship Manager) completo desarrollado de manera integra con Rich Faces por Zestia solutions</p>	<p><a href="http://www.javelincrm.com/">http://www.javelincrm.com/</a></p>
<p><b>Suntrip Recors</b></p> 	<p>Web Site y tienda en línea</p>	<p><a href="http://www.suntriprecords.com/">http://www.suntriprecords.com/</a></p>

**TABLA 2. PRINCIPALES EMPRESAS QUE USAN RICH FACES**

**Elaborado por:** El autor

### 1.2.3 Ext Js

Ext Js es un conjunto de librerías JavaScript que permite el desarrollo de aplicaciones RIA basadas en un navegador, sin embargo se integra mediante plug-ins con Adobe AIR para generar aplicaciones RIA basadas en escritorio.

Ext Js se concibe como una extensión de la librería de Yahoo YUI (Yahoo User Interface), que posteriormente se convierte en un proyecto independiente sin perder la integración con YUI.

Ext Js ofrece al desarrollador un gran conjunto de widgets (componentes como por ejemplos grids, ventanas de dialogo, etc) plenamente integrados y un API (Application Program Interface) para conseguir interfaces web más dinámicas e interactivas con el usuario.

A inicios del 2006 de la mano de Jack Solum nace lo que actualmente se conoce como Ext-JS, su proyecto era una “extension” de utilidades de YUI. Estas utilidades se agrupan en una librería denominada yui-ext que para mediados de 2006 en su versión 0.33 se convierte en open source, bajo el tipo de licenciamiento BSD (Berkeley Software Distribution) permitiendo el uso del código en software no libre.

Con el incremento de popularidad y contribuciones de la comunidad open source, yui-ext se convierte en Ext no solo una librería sino un framework completo para el desarrollo de aplicaciones RIA.

Con estos antecedentes para 2007 se funda la compañía Ext teniendo como productos principales ExtJS y ExtGWT; ExtJS es el framework producto de la librería inicial de extensión de YUI, mientras ExtGWT es un framework optimizado para trabajar con Google Web Toolkit.

La versión 1.0 de Ext JS fue lanzada al mercado en Abril de 2007, esta versión se mantuvo vigente casi un año ya que para Abril de 2008 se lanza Ext JS 2.1 que debido a sus grandes cambios de widgets y replanteo de librerías se vuelve incompatible con la versión 1.

Para Agosto de 2008 se lanza la versión 2.2. que mantiene la incompatibilidad con la versión 1 sin embargo aumenta las capacidades de soporte de navegadores, de esta manera Ext Js se convierte en un framework que mantiene el mismo look and feel independientemente del navegador y la plataforma de cliente.

Para la fecha actual mayo de 2009 se encuentra disponible la versión 3.0 RC (Release Candidate), se espera que en los meses próximos esta salga como release definitivo, con esta versión se elimina por completo el soporte a la versión 1 y se evita la incompatibilidad por la salida total del mercado de esta versión.

#### **1.2.3.1 Lenguaje de Programación**

Ext Js usa el lenguaje JavaScript junto con HTML para la creación de las interfaces de usuario, así como para el manejo de eventos en cada una de las páginas que comprenden una aplicación desarrollada con Ext Js.

Al usar JavaScript, Ext Js trae consigo los beneficios propios del lenguaje tal como:

- La orientación a objetos,
- La manipulación del DOM (Ext Js extiende esta capacidad con su propia implementación para el manejo de DOM)
- El soporte de múltiples navegadores como Internet Explorer, Opera, Safari y Mozilla Firefox y la concordancia con el estándar ECMA-226

### 1.2.3.2 Patrones de Diseño

Ext Js al ser un framework estructurado de JavaScript puede agregarse a los diferentes patrones de diseño que se proponen para el lenguaje si se trata de la creación de nuevos componentes.

Estos se dividen en:

#### **Patrones Creacionales**

Los cuales determinan la manera de creación e instanciación de clases de objetos.

#### **Patrones Estructurales**

Determinan el tipo de composición de los objetos, se basan principalmente en la herencia para establecer una estructura funcional.

#### **Patrones de Comportamiento**

Determinan el tipo de comunicación entre los objetos.

Si se trata del uso de los componentes ya existentes dentro del framework no se ha definido un patrón de diseño específico, esto debido a que es susceptible de integrarse con aplicaciones desarrolladas utilizando la especificación JEE o con aplicaciones desarrolladas con .Net o PHP.

### 1.2.3. 3 Descripción de componentes

Para el desarrollo y puesta en producción de una aplicación utilizando ExtJs es necesario contar con una serie de elementos estructurales, estos son los siguientes:

#### **ext-all.css:**

Se trata de la hoja de estilos, inglés CSS (Cascade Style Sheet), que controla el look and feel de los widgets de Ext, esta debe mantenerse invariante y estar presente siempre en un proyecto. Si es necesario dar un toque personalizado a los widgets se debe incluir la nueva hoja de estilos además de la ext-all.css.

**ext-base.js y ext-core:**

Es la librería núcleo de Ext, ofrece un conjunto de funcionalidades que permiten una relación sencilla con paginas html, agregar interacción con AJAX y la manipulación del árbol DOM (Document Object Model) con lo cual es posible manipular la estructura y estilo de documentos HTML y XML.

La manipulación del árbol DOM se la realiza mediante el objeto Ext.Element el cual envuelve la mayoría de propiedades de DOM, con lo cual la interacción se vuelve más sencilla por las funcionalidades del framework.

Finalmente Ext conserva su compatibilidad original con librerías externas de características similares a las descritas para ext-base.js; estas librerías son JQuery y Prototype. Esta compatibilidad se debe a sus orígenes como una extensión de YUI.

**ext-all-debug/ext-all.js**

Se trata de la librería que contiene los widgets predefinidos por Ext. La versión debug se utiliza para la creación de widgets personalizados, mientras que la otra versión se utiliza para la puesta en producción con los elementos ya existentes.

**1.2.3.4 Plataforma Tecnológica de Desarrollo**

El framework Ext no posee de manera oficial un IDE que permita un desarrollo ágil de aplicaciones, sin embargo existen varias opciones compatibles, entre estas se tiene Spket, Aptana Studio, Komodo Edit, Dreamweaver usando los plugins de Spket.

El IDE escogido para el desarrollo del presente trabajo es Spket versión 1.6 de manera conjunta con Netbeans para el despliegue de la aplicación.

Spket es un IDE de desarrollo construido sobre eclipse, de esta manera puede integrarse como un plugin o instalarse de manera independiente, soporta la edición de Ext Js de manera nativa permitiendo disminuir el tiempo de desarrollo, para realizar un debugging del código generado se utilizará Mozilla Firefox junto con su complemento firebug.

Principalmente para el desarrollo se requiere el SDK de Ext, el mismo que incluye incluye la documentación del framework, ejemplos y código fuente.

La última versión estable y oficialmente lanzada al mercado es la 2.2.1 la cual tiene la estructura que se muestra en la figura 1.5

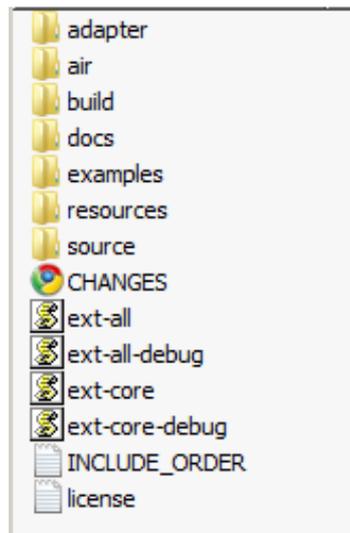


FIGURA 1.6. ESTRUCTURA DE DIRECTORIOS DE EXT JS

Fuente: Elaborada por el Autor

A continuación se describe la estructura que se muestra en la figura 1.5:

- El directorio adapter contiene los archivos de core de Ext, además de JQuery y Prototype que son las posibles opciones de integración.
- Air que incluye los archivos necesarios para la integración de Ext con adobe Air.
- Build contiene los archivos que permiten construir widgets personalizados.
- Doc contiene la documentación de los componentes del framework.
- Examples varios ejemplos de páginas que utilizan el framework
- Resources contiene imágenes y la hoja de estilos ext-all.css
- Source contiene el código fuente de Ext

Spket como IDE de desarrollo es gratuito para fines no comerciales, si se requiere utilizar para estos fines el precio de la licencia fluctúa entre los \$19.50 a \$29.90 dólares dependiendo del número de usuarios. Los requerimientos son los siguientes:

Windows XP, Vista y GNU/Linux arquitecturas x86

- Java <sup>TM</sup> JVM 1.5 o posterior
- 10.3 MB de espacio disponible en disco
- No hay requerimientos mínimos de RAM y procesador

Plug in Eclipse (independiente de la plataforma)

- Eclipse 3.2 mínimo

- 4.42 MB de espacio disponible en disco
- No hay requerimientos mínimos de RAM y procesador

Para que una aplicación desarrollada con Ext Js pueda ser accedida por clientes de manera remota, la aplicación puede ser colocada en cualquier servidor web, por lo tanto no es necesaria una especificación detallada de que servidores utilizar para la puesta en producción.

### **1.2. 3.5 Requerimientos de Hardware y Software de Cliente**

Los requerimientos para el acceso de los clientes a aplicaciones Ext se limitan únicamente a navegadores que soporten JavaScript, como se mencionó anteriormente las posibilidades son muchas, entre estas se tiene:

- Internet Explorer.
- Mozilla Firefox.
- Chrome.
- Opera.
- Safari.
- Konqueror.

No existen especificaciones para el hardware de cliente.

### **1.2. 3.6 Costos de Licencias**

Ext Js proporciona un tipo de licenciamiento dual, un tipo de licencia para comercializar el producto y otra para distribución del tipo open source normada por GPL (GNU Public License), con lo cual se asegura que el producto de software se mantenga como open source

Los costos de las licencias comerciales varían dependiendo del número de desarrolladores y del tipo de soporte.

Así se tiene que los precios de las licencias son los siguientes:<sup>5</sup>

Developer License (para un solo desarrollador)

Licencia Comercial	\$259
Licencia Comercial + Silver Support	\$509
Actualización de versión	\$197

---

<sup>5</sup> Precios obtenidos de <http://extjs.com/store/extjs/> el 9 de junio de 2009

Team License (para un equipo de 5 a 25 desarrolladores)

Licencia Comercial	\$1,029
Licencia Comercial + Gold Support	\$1,869
Actualización de versión	\$779

Workgroup License (para un equipo de 25 a 100 desarrolladores)

Licencia Comercial	\$4,589
Licencia Comercial + Platinum Support	\$6,719
Actualización de versión	\$3,449

Enterprise License (para un equipo de más de 100 desarrolladores)

Licencia Comercial	\$13,159
Licencia Comercial + Diamond Support	\$17,409
Actualización de versión	\$ 9,869

Adicionalmente el soporte también se encuentra disponible de manera separada de las licencias y los costos son los siguientes:

- Silver \$ 299
- Gold \$ 999
- Platinum \$2,499
- Diamond \$4,999

### 1.2.3.7 Principales empresas que utilizan Ext Js

Empresa	Aplicación	URL
<b>CISCO Systems</b> 	Usada como librería de JavaScript en la página principal de CISCO	<a href="http://www.cisco.com/">http://www.cisco.com/</a>
<b>Pixar Animation Studios</b> <b>P I X A R</b>	Usada como librería de JavaScript en la página principal de Pixar	<a href="http://www.pixar.com">http://www.pixar.com</a>
<b>T-Mobile</b> 	Usada como librería de JavaScript en la página principal de T-Mobile	<a href="http://www.t-mobile.com/">http://www.t-mobile.com/</a>
<b>CANON</b> 	Usada como librería de JavaScript en la página principal de T-Mobile	<a href="http://www.canon.ca/english/">http://www.canon.ca/english/</a>

TABLA 3. PRINCIPALES EMPRESAS QUE USAN EXT JS

Elaborado por: El autor

## CAPITULO 2. ANÁLISIS DE LOS FRAMEWORKS DE DESARROLLO PARA APLICACIONES RIA

### 2.1 DEFINICIÓN DE CRITERIOS DE EVALUACIÓN DE LOS FRAMEWORKS

Se definen a continuación los criterios que permitirán evaluar los frameworks objetos de estudio de este trabajo, se definen criterios para las características técnicas descritas en el capítulo 1, así como los que se utilizarán para el desarrollo de los prototipos.

Para los criterios cuya valoración se puede obtener basada en la descripción del capítulo 1, se dará una valoración final mientras que aquellos criterios que requieren de comprobación mediante la ejecución de pruebas sobre los prototipos obtendrán una valoración preliminar.

#### 2.1.1 CRITERIOS DE EVALUACIÓN CONSIDERADOS

Los criterios considerados para la evaluación de los frameworks se resumen en la tabla 4, el detalle de cada criterio se realiza de manera inmediata a la tabla de resumen

Categoría	Criterio	Valoración
1. Apariencia	Apariencia en los diferentes navegadores	1-4
	Independencia de la resolución	1-2
2. Rendimiento	Tiempo de descarga o tiempo de inicialización	
	Consumo de canal de red	
3. Recursos	Requerimientos de Hardware y Software para el desarrollo	1-4
	Costos	1-4
4. Arquitectura	Patrones de Diseño	1-4
	Estandarización del Lenguaje	1-4
5. Soporte	Soporte del producto	1-4
	Road Map Tecnológico	1-4

	Tipo de Licenciamiento	1-2
<b>6. Facilidades para el desarrollo</b>	Curva de aprendizaje	1-4
	Herramientas de desarrollo	1-4
	Facilidad de interacción con otras tecnologías	1-4
	Facilidad de puesta en producción	1-2

TABLA 4. RESUMEN DE CRITERIOS DE EVALUACIÓN

Elaborado por: El autor

## 1. APARIENCIA

### 1.1 Apariencia en los diferentes navegadores (AP1)

Este criterio toma en cuenta la necesidad de conservar el look & feel de una aplicación, independientemente del navegador del cual ha sido accedido.

El criterio se refiere además a evaluar la necesidad de configuración adicional por parte del usuario para mantener el estado de su aplicación en los diferentes navegadores.

De manera adicional se incluye la existencia de certificaciones de los frameworks sobre los navegadores.

#### VALORACIÓN

Para la valoración se considerarán los siguientes navegadores:

- Internet Explorer versión 7 y 8.
- Mozilla Firefox versión 3.0.17.
- Apple Safari versión 4.0.0.
- Google Chrome 3.0.195.38

Valor	Significado
1	La apariencia de la aplicación es totalmente diferente en los navegadores considerados y no incluye ningún certificado sobre los navegadores. (Ver anexos Listado de Navegadores más usados a nivel mundial)
2	La apariencia de la aplicación es la misma en 2 de los

	navegadores considerados, sin incluir ninguna certificación sobre alguno de los 2 navegadores.
<b>3</b>	La apariencia de la aplicación es la misma en 2 de los navegadores considerados, e incluye al menos una certificación sobre alguno de los 2 navegadores.
<b>4</b>	La apariencia de la aplicación es idéntica en al menos 3 de los navegadores considerados, sin incluir certificaciones sobre los navegadores

**TABLA 5. VALORACIÓN APARIENCIA**

**Elaborado por:** El autor

## 1.2 Independencia de la resolución (AP2)

De manera complementaria al criterio anterior (Apariencia en los diferentes navegadores), el criterio considerado se refiere a la capacidad de los frameworks para representar correctamente los controles gráficos, de manera que estos conserven su apariencia independientemente de la resolución de la pantalla.

### VALORACIÓN

Valor	Significado
<b>1</b>	Los controles se distorsionan al cambiar la resolución de la pantalla en todos los exploradores.
<b>2</b>	Los controles conservan sus características gráficas independientemente de la resolución de pantalla desde la cual se accede a la aplicación

**TABLA 6. VALORACIÓN INDEPENDENCIA DE RESOLUCIÓN**

**Elaborado por:** El autor

## 2. RENDIMIENTO

### 2.1 Tiempo de descarga o tiempo de inicialización (RE1)

Se trata tal vez de uno de los aspectos más importantes desde el punto de vista del usuario, se refiere al tiempo que debe esperar antes de poder utilizar la

aplicación. Para las aplicaciones RIA basadas en un navegador este factor depende de manera directa de la velocidad de la conexión.

Las páginas HTML simples tienen siempre un tiempo de descarga menor al de aplicaciones con alto contenido multimedia, sin embargo con las velocidades de las conexiones actuales esta diferencia se vuelve menos evidente.

#### VALORACIÓN

La valoración de este criterio se dará en función del menor tiempo de descarga obtenido en la evaluación de prototipos en el capítulo 4, asignándole el valor más alto al que tome menor tiempo en la inicialización.

### 2.2 Consumo de canal de red (RE2)

Se refiere al ancho de banda necesario para poder interactuar con la aplicación, considera el tráfico cliente-servidor y el procesamiento que se lleva a cabo en cada uno de los componentes necesarios para el funcionamiento de las aplicaciones.

## 3. RECURSOS

### 3.1 Requerimientos de Hardware y software para el desarrollo (REC1)

Criterio que evalúa las características requeridas para el desarrollo, esto debido a que en cuanto a requerimientos de cliente no existen mayores especificaciones. En el tema de software considera principalmente la integración con diferentes sistemas operativos como plataforma de soporte para la ejecución del framework, considera además independencia de software adicional requerido para el funcionamiento del framework.

#### VALORACIÓN

Valor	Significado
1	El framework requiere de hardware especializado (ej. Tarjeta de video) para el desarrollo de aplicaciones, y además es dependiente del sistema operativo.
2	El framework es limitado en cuanto al soporte de sistemas operativos, sin embargo no requiere de hardware adicional.
3	El framework soporta múltiples sistemas operativos, sin embargo requiere del soporte de software adicional para su

	correcto funcionamiento.
<b>4</b>	Soporta múltiples sistemas operativos y no se requiere hardware adicional al requerido por el sistema operativo sobre el cual se encuentra instalado. De la misma manera el framework no requiere de software adicional para su funcionamiento.

**TABLA 7. VALORACIÓN RECURSOS HW Y SW**

**Elaborado por:** El autor

### 3.2 Costos (REC 2)

Criterio que se refiere primordialmente a toda la inversión necesaria para desarrollar una aplicación RIA con cualquiera de los frameworks, toma en cuenta costo de licencias así como costo del soporte del producto. El costo a evaluar será el costo individual de una licencia de desarrollo y el costo de soporte promedio para un periodo de un año.

#### VALORACIÓN

Valor	Significado
<b>1</b>	Costo total mayor a 5000 dólares
<b>2</b>	Costo total entre 3000 y 5000 dólares
<b>3</b>	Costo total entre 1500 y 3000 dólares
<b>4</b>	Costo total menor a 1500 dólares

**TABLA 8. VALORACIÓN COSTOS**

**Elaborado por:** El autor

## 4. Arquitectura

### 4.1 Patrones de diseño (ARQ1)

Considera la capacidad de los frameworks para desarrollar aplicaciones RIA siguiendo patrones probados de diseño, que faciliten la implementación mediante una organización estructurada de los componentes.

#### VALORACIÓN

Valor	Significado
<b>1</b>	No se considera patrón de diseño alguno para el desarrollo de aplicaciones.

2	Existe al menos un patrón de diseño para el desarrollo de aplicaciones, con algún tipo de documentación
3	Existe un patrón de diseño aceptado por la comunidad, la cual le ha dado soporte y documentado el proceso de diseño.
4	El patrón de diseño se encuentra aceptado de manera formal por la empresa desarrolladora y con documentación formal.

**TABLA 9. VALORACIÓN PATRONES DE DISEÑO**

**Elaborado por:** El autor

## 4.2 Estandarización del Lenguaje (ARQ2)

Bajo este criterio se considerará el lenguaje utilizado en el desarrollo de las aplicaciones, se tomará en cuenta la implementación de estándares como XML, estándares de lenguajes de scripting, etc. Si no se aplica estándares se considerará la documentación existente sobre el lenguaje que se utiliza.

### VALORACIÓN

Valor	Significado
1	El framework utiliza un lenguaje propietario del cual no se apega a estándares.
2	El framework utiliza un lenguaje propietario que si implementa estándares.
3	El framework utiliza un lenguaje libre propietario que implementa estándares reconocidos por organizaciones internacionales.
4	El framework utiliza un lenguaje libre apegado a estándares.

**TABLA 10. ESTANDARIZACIÓN DE LENGUAJES**

**Elaborado por:** El autor

## 5. Soporte

### 5.1 Soporte del producto (SOP1)

Criterio en cual que se toma en cuenta las responsabilidades que cada que cada organización a cargo del desarrollo del framework tiene en la solución de problemas o bugs propios de sus productos, a la vez se refiere al acceso a documentación, cursos de capacitación y posibilidades de certificación.

## VALORACIÓN

Valor	Significado
1	No existe soporte alguno para el producto.
2	El soporte lo brinda únicamente la comunidad de desarrolladores.
3	El soporte es exclusivo de la empresa desarrolladora, y no existe apoyo formal para la comunidad de desarrolladores.
4	Existe soporte formal para los errores propios del producto, además de una comunidad que respalda el uso, esta comunidad recibe apoyo de la empresa desarrolladora

TABLA 11. VALORACIÓN DE SOPORTE

Elaborado por: El autor

**5.2 Road Map Tecnológico (SOP2)**

Este criterio considera la planificación tecnológica del producto, considera el desenvolvimiento desde la salida al mercado del producto y las perspectivas al próximo año, considera la planificación de nuevas versiones o cambios que permitan adaptar al producto para ajustarse a nuevas necesidades.

## VALORACIÓN

Valor	Significado
1	La primera versión del producto no ha sufrido modificaciones y la documentación no ha sido actualizada, y no existen perspectivas de modificaciones al producto.
2	El producto ha pasado por varias versiones, sin embargo no se conocen perspectivas de nuevas modificaciones y adaptaciones.
3	El producto ha pasado por varias modificaciones, acorde a necesidades del cliente y nuevos requerimientos de estos. Sin embargo no existe documentación que respalde y verifique los cambios.

- |          |  |
|----------|--|
| <b>4</b> | El producto cuenta con una planificación de versiones, implementación de requerimientos, mantenimiento de la documentación. Las perspectivas del producto están establecidas formalmente y se encuentran documentadas. |
|----------|--|

**TABLA 12. VALORACIÓN ROAD MAP TECNOLÓGICO**

**Elaborado por:** El autor

### 5.3 Tipo de Licenciamiento (SOP3)

Considera las diferencias en licenciamiento de los productos de software desarrollados a partir del uso de los frameworks, las limitaciones y las perspectivas para comercializar dichos productos. Permite hacer una diferenciación entre la orientación de los frameworks con respecto a software libre y software privativo.

#### VALORACIÓN

Valor	Significado
1	Software libre GPL
2	software libre sin obligación de generar software libre (LGPL)
1	Software privativo
2	código propietario abierto

**TABLA 13. VALORACIÓN TIPO DE LICENCIAMIENTO**

**Elaborado por:** El autor

## 6. Facilidades para el desarrollo

### 6.1 Curva de aprendizaje (DES1)

Determina la facilidad e inversión de tiempo necesarios para obtener un nivel de conocimiento aceptable del uso de los componentes de cada uno de los frameworks.

Consideremos que los lenguajes de scripting como JavaScript y ActionScript son menos populares que los lenguajes orientados a la construcción de aplicaciones de escritorio, sean estos por ejemplo C#, Java o Visual Basic; lo cual trae par el desarrollador que no se encuentra familiarizado con estos, un esfuerzo para acoplarse al lenguaje y explotar sus capacidades e identificar debilidades.

## VALORACIÓN

Valor	Significado
1	El framework utiliza componentes cuyo manejo requiere un profundo conocimiento y años de estudio.
2	El lenguaje que utiliza el framework es una implementación propietaria que requiere del aprendizaje de la misma, tomando un tiempo de aprendizaje de varios meses.
3	Si se conoce lenguajes como C# y Java, la utilización del framework a un nivel intermedio requiere de pocas semanas.
4	Sin experiencia previa en lenguajes de programación, únicamente conociendo los fundamentos, el desarrollador es capaz de utilizar y conocer los componentes de una manera adecuada en pocas semanas.

TABLA 14. VALORACIÓN CURVA DE APRENDIZAJE

Elaborado por: El autor

**6.2 Herramientas para el desarrollo (DES2)**

Criterio que busca determinar la productividad en el desarrollo de aplicaciones RIA, fijándose en la existencia de IDE's, herramientas para depuración de código y herramientas para realización de pruebas.

Es necesario anotar que buenas herramientas facilitan al desarrollador la ejecución de tareas recurrentes, considerando de forma especial la posibilidad de aplicar el paradigma WYSIWYG (What You See Is What You Get).

## VALORACIÓN

Valor	Significado
1	Los componentes del framework se encuentran aislados y no existe una herramienta que mejore la productividad, permitiendo el diseño de aplicaciones, ejecución de pruebas y depuración de código.
2	El framework se integra con IDE's de terceros mediante

	complementos permitiendo la edición de aplicaciones a nivel de análisis sintáctico de código
<b>3</b>	Existe un IDE perteneciente a la misma empresa que desarrollo el framework o de terceros que permite el desarrollo de aplicaciones, así como la ejecución de pruebas, al menos al nivel de editor de código
<b>4</b>	Existe un IDE que permite el desarrollo de aplicaciones a nivel gráfico además de la edición de código. El IDE considera la integración para la ejecución de pruebas y despliegue de la aplicación.

**TABLA 15. VALORACIÓN HERRAMIENTAS DE DESARROLLO**

**Elaborado por:** El autor

### **6.3 Facilidad de interacción con varias tecnologías (DES3)**

Criterio que se refiere a la capacidad que brindan los frameworks para interactuar con bases de datos, web services y acoplamiento con aplicaciones desarrolladas con lenguajes distintos a los nativos de cada framework, con esto comprendemos la facilidad de acoplamiento del Front-end RIA con la capa de negocio, la cual puede estar desarrollada utilizando tecnologías como Java o .Net.

Considera también la facilidad para agregar contenido multimedia dentro de las aplicaciones, así como también la incorporación de AJAX en las aplicaciones que corren sobre un navegador.

#### **VALORACIÓN**

<b>Valor</b>	<b>Significado</b>
<b>1</b>	No existen API's de integración con tecnologías de terceros
<b>2</b>	Las API's de integración abarcan gran cantidad de tecnologías, mas requieren de un conocimiento profundo del framework.
<b>3</b>	Las API's de integración son limitadas y se suscriben a tecnologías específicas, sin embargo no requieren de un gran

	esfuerzo para conseguir esta integración.
<b>4</b>	Existe un gran número de API's que soporta un conjunto amplio de tecnologías estándar.

**TABLA 16. VALORACIÓN INTEGRACIÓN CON OTRAS TECNOLOGÍAS**

**Elaborado por:** El autor

#### **6.4 Facilidad de puesta en producción (DES4)**

Criterio que considera la complejidad para que una aplicación pueda ser puesta en un servidor para ser accedida desde los clientes. Considera la diversidad de servidores de aplicaciones que soportan.

VALORACIÓN

<b>Valor</b>	<b>Significado</b>
<b>1</b>	La aplicación depende de un servidor especializado y requiere de configuraciones especiales.
<b>2</b>	La aplicación puede ser puesta en producción en diversos servidores, sin requerir configuraciones especiales.

**TABLA 17. VALORACIÓN DESPLIEGUE DE APLICACIÓN**

**Elaborado por:** El autor

#### **2.1.2 PONDERACIÓN DE CRITERIOS**

Para dar una ponderación y estimar que criterios tendrán mayor relevancia en la evaluación, se utilizará una parte del proceso analítico jerárquico o proceso analítico de jerarquías.

##### **2.1.2.1 Proceso Analítico jerárquico**

El proceso analítico jerárquico (AHP Analytical Hierarchy Process) considera que para la toma de una decisión o elección de una alternativa, esta puede ser dividida en varios niveles jerárquicos, el nivel final corresponde a las alternativas, los niveles intermedios a los criterios y el nivel superior a la meta final.

Los criterios son evaluados por pares mediante una matriz de comparación y una escala de valores numéricos establecida en base a expresiones verbales. Los valores de la escala son números enteros, no necesariamente secuenciales para tener la posibilidad de colocar valores intermedios que representen un punto medio entre los valores de la escala.

Así tenemos por ejemplo la escala de la figura 2.1, para la cual una valoración de 4 representa el valor intermedio entre moderada y fuerte.

Importancia o preferencia	Valor cardinal
Igual	1
Moderada	3
Fuerte	5
Muy Fuerte	7
Extremadamente fuerte	9

**FIGURA 2.1. EJEMPLO DE ESCALA DE VALORES**

**Fuente:** [http://www.multicriteria.org/ejemplo\\_ahp.htm](http://www.multicriteria.org/ejemplo_ahp.htm)

Para la matriz de comparación se coloca los criterios en la primera fila y en la primera columna de la matriz, en cada una de las celdas se coloca el valor de la escala obtenido de la comparación entre un elemento de la fila con el correspondiente elemento de la columna, para elementos ya comparados se coloca el valor del inverso multiplicativo del valor previamente asignado.

Posteriormente se realiza la suma de los valores por columna, y se realiza una nueva matriz normalizada obtenida de dividir el valor de cada celda con el correspondiente valor de la suma de su columna.

Finalmente se obtiene el promedio de los valores de la fila y este es el factor que determinará la ponderación de los criterios. Las figuras 2.2 y 2.3 muestran un ejemplo de la matriz de comparación

	C1	C2	C3
C1	1	3	6
C2	1/3	1	2
C3	1/6	1/2	1

**FIGURA 2.2. MATRIZ DE COMPARACIÓN**

**Fuente:** Grafico mejorado de: [http://www.multicriteria.org/ejemplo\\_ahp.htm](http://www.multicriteria.org/ejemplo_ahp.htm)

	Calidad	Precio	Servicio	Entrega	Ponderaciones promedio de fila
Calidad	12/25	6/11	8/17	1/3	0,457
Precio	6/25	3/11	6/17	1/3	0,300
Servicio	3/25	1/11	2/17	2/9	0,138
Entrega	4/25	1/11	1/17	1/9	0,105
Totales	1	1	1	1	1,000

FIGURA 2.3. MATRIZ DE COMPARACIÓN NORMALIZADA

Fuente: El autor

En el presente trabajo únicamente se utilizará la matriz de comparación y la escala para establecer pesos a los diferentes criterios.

### 2.1.2.2 Elaboración de la matriz de comparación y escala de valores

Escala de valores de preferencia

Preferencia	Valor
Igual	1
Media	3
Fuerte	5
Muy Fuerte	7

TABLA 18. ESCALA DE VALORES DE PREFERENCIA

Elaborado por: El autor

	AP1	AP2	RE1	RE2	REC1	REC2	ARQ1	ARQ2	SOP1	SOP2	SOP3	DES1	DES2	DES3	DES4
AP1	1	3	1/7	1/5	1/6	2	1/3	1/5	1/3	2	1	1	1/5	1	1
AP2	1/3	1	1/7	1/7	1/5	1/5	1/3	1/3	1/3	1/3	1/3	1/5	1/5	1/3	1/3
RE1	7	7	1	1	3	1	2	2	1	5	1	1	1	1	1
RE2	5	7	1	1	5	1	2	2	1	5	2	2	1	2	1
REC1	6	5	1/3	1/5	1	½	1/3	1/3	1	2	2	1/5	1	2	1
REC2	1/2	5	1	1	2	1	1/2	1/2	1	1	1	1	1/3	1/2	½
ARQ1	3	3	½	1/2	3	2	1	1	3	5	3	1	3	1/2	2
ARQ2	5	3	½	1/2	3	2	1	1	1/3	3	3	1	1	1/2	1
SOP1	3	3	1	1	1	1	1/3	3	1	3	3	1/2	1	1/2	½
SOP2	1/2	3	1/5	1/5	½	1	1/5	1/3	1/3	1	3	1/3	1/3	1/5	1/3
SOP3	1	3	1	1/2	½	1	1/3	1/3	1/3	1/3	1	1/5	1/5	1/6	6
DES1	1	5	1	1/2	5	1	1	1	2	3	5	1	1	4	2
DES2	5	5	1	1	1	3	1/3	1	1	3	5	1	1	1	1
DES3	1	3	1	1/2	½	2	2	2	2	5	6	1/4	1	1	½
DES4	1	3	1	1	1	2	1/2	1	2	3	1/6	1/2	1	2	1
	40 1/3	59	10 5/6	9 1/4	26 7/8	20 2/3	12 1/5	16	16 2/3	41 2/3	36 1/2	11 1/5	13 1/4	16 2/3	19 1/6

TABLA 19. MATRIZ DE COMPARACIÓN

Elaborado por: El autor

	AP1	AP2	RE1	RE2	REC1	REC2	ARQ1	ARQ2	SOP1	SOP2	SOP3	DES1	DES2	DES3	DES4	
<b>AP1</b>	3/121	3/59	4/303	14/647	5/806	20/207	5/183	6/481	1/50	6/125	2/73	60/671	3/199	10/167	6/115	0,0386
<b>AP2</b>	1/121	1/59	4/303	10/647	3/403	2/207	5/183	10/481	1/50	1/125	2/219	12/671	3/199	10/501	2/115	0,0156
<b>RE1</b>	21/121	7/59	11/119	70/647	45/403	10/207	10/61	60/481	3/50	3/25	2/73	60/671	15/199	10/167	6/115	0,0894
<b>RE2</b>	15/121	7/59	11/119	70/647	75/403	10/207	10/61	60/481	3/50	3/25	4/73	120/671	15/199	20/167	6/115	0,1074
<b>REC1</b>	18/121	5/59	24/779	14/647	15/403	5/207	5/183	10/481	3/50	6/125	4/73	12/671	15/199	20/167	6/115	0,0482
<b>REC2</b>	3/242	5/59	11/119	70/647	30/403	10/207	5/122	15/481	3/50	3/125	2/73	60/671	5/199	5/167	3/115	0,0544
<b>ARQ1</b>	9/121	3/59	11/238	35/647	45/403	20/207	5/61	30/481	9/50	3/25	6/73	60/671	45/199	5/167	12/115	0,0954
<b>ARQ2</b>	15/121	3/59	11/238	35/647	45/403	20/207	5/61	30/481	1/50	9/125	6/73	60/671	15/199	5/167	6/115	0,0661
<b>P1</b>	9/121	3/59	11/119	70/647	15/403	10/207	5/183	90/481	3/50	9/125	6/73	30/671	15/199	5/167	3/115	0,0673
<b>SOP2</b>	3/242	3/59	11/595	14/647	15/806	10/207	1/61	10/481	1/50	3/125	6/73	20/671	5/199	2/167	2/115	0,0290
<b>SOP3</b>	3/121	3/59	11/119	35/647	15/806	10/207	5/183	10/481	1/50	1/125	2/73	12/671	3/199	5/501	36/115	0,0517
<b>DES1</b>	3/121	5/59	11/119	35/647	75/403	10/207	5/61	30/481	3/25	9/125	10/73	60/671	15/199	40/167	12/115	0,1034
<b>DES2</b>	15/121	5/59	11/119	70/647	15/403	10/69	5/183	30/481	3/50	9/125	10/73	60/671	15/199	10/167	6/115	0,0788
<b>DES3</b>	3/121	3/59	11/119	35/647	15/806	20/207	10/61	60/481	3/25	3/25	12/73	15/671	15/199	10/167	3/115	0,0850
<b>DES4</b>	3/121	3/59	11/119	70/647	15/403	20/207	5/122	30/481	3/25	9/125	1/219	30/671	15/199	20/167	6/115	0,0698
	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00

**TABLA 20. MATRIZ DE COMPARACIÓN NORMALIZADA**

Elaborado por: El autor

A continuación se muestra el resumen de la matriz de comparación normalizada, indicando el orden de preferencia de los criterios, cada valor se ha multiplicado por 100 con el fin de apreciar las diferencias en las preferencias.

<b>Criterio</b>	<b>Valor (normalizado x 100)</b>
<b>Consumo de canal de red</b>	10,7378
<b>Curva de aprendizaje</b>	10,3405
<b>Patrones de diseño</b>	9,5415
<b>Tiempo de descarga o tiempo de inicialización</b>	8,9440
<b>Facilidad de interacción con varias tecnologías</b>	8,4954
<b>Herramientas para el desarrollo</b>	7,8789
<b>Facilidad de puesta en producción</b>	6,9803
<b>Soporte del producto</b>	6,7267
<b>Estandarización del Lenguaje</b>	6,6063
<b>Costos</b>	5,4447
<b>Tipo de Licenciamiento</b>	5,1699
<b>Requerimientos de Hardware y software para el desarrollo</b>	4,8191
<b>Apariencia en los diferentes navegadores</b>	3,8590
<b>Road Map Tecnológico</b>	2,8969
<b>Independencia de la resolución</b>	1,5591

**TABLA 21. RESUMEN DE MATRIZ DE COMPARACIÓN**

Elaborado por: El autor

## **2.1.3 DEFINICIÓN DEL CASO DE APLICACIÓN**

### **2.1.3.1 Propósito:**

El caso de aplicación se consiste en proporcionar una solución al manejo de información de resultados de exámenes de laboratorio, mediante la automatización del proceso de ingreso y reporte de los mismos con el fin facilitar el acceso de pacientes y médicos a los reportes de resultados.

Para la selección del caso de aplicación, se consideró la cantidad de controles que se pueden utilizar y la riqueza gráfica para una aplicación medianamente transaccional; en lugar de realizar una aplicación netamente informativa o promocional con mayor riqueza gráfica.

### **2.1.3.2 Alcance:**

El sistema es responsable del manejo de la información, de los resultados de exámenes de laboratorio, este manejo incluye:

- El registro de información del paciente.
- Registro del pedido de exámenes, incluyendo la identificación de la muestra del paciente.
- Ingreso de valores de resultados para cada uno de los exámenes registrados en el pedido.
- Consulta de valores de resultados de los exámenes.

El sistema no se encarga de la actualización de información de exámenes que el laboratorio ofrece.

El sistema no se encargará del registro de usuarios autorizados y definición de roles de usuarios. El sistema no actuará como intermediario para la recepción de pagos por concepto de solicitud de exámenes.

### 2.1.3.3 Esquema General: Casos de Uso

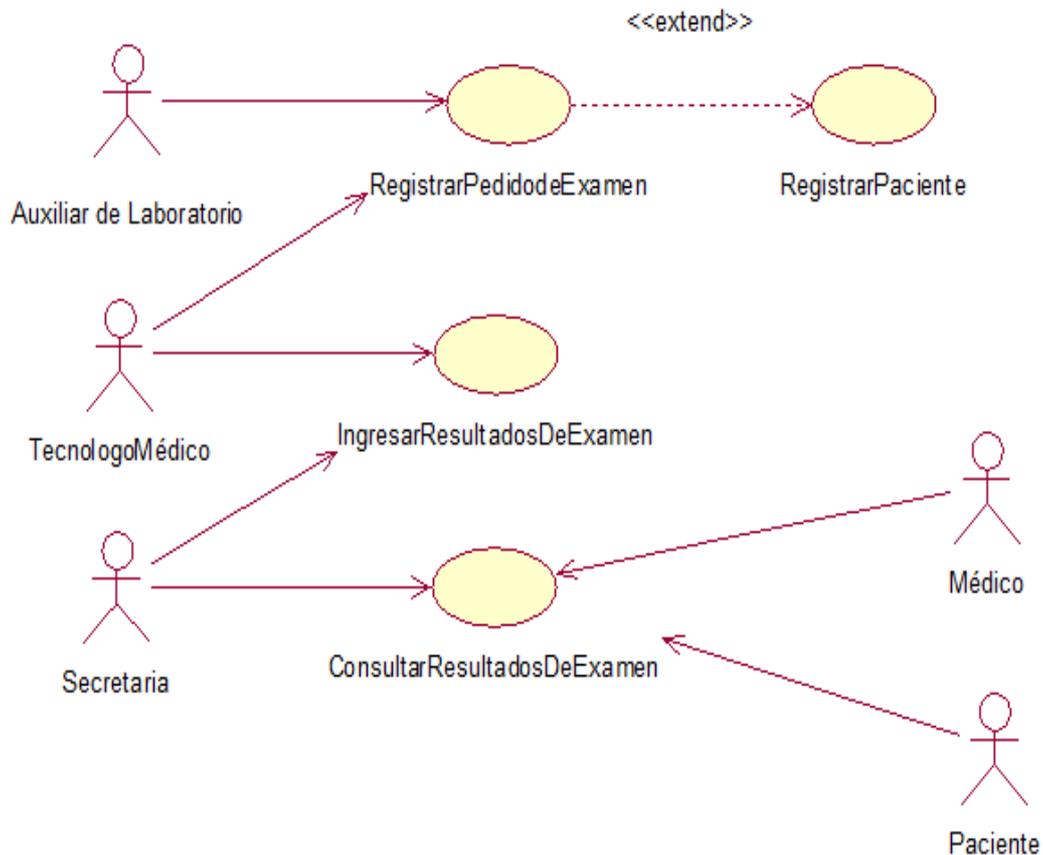


FIGURA 2.4. DIAGRAMA DE CASOS DE USO

Fuente: El autor

### 2.1.3.4 Descripción de Actores

ACTOR	DESCRIPCIÓN
1 <b>Tecnólogo Médico</b>	Es la persona encargada de registrar los exámenes que el paciente desee, se encarga de la identificación y selección de las muestras.
2 <b>Auxiliar</b>	El Auxiliar de Laboratorio es quién apoya en las diferentes funciones al Tecnólogo Médico, es quién inicia el proceso de solicitud de exámenes de los pacientes.
3 <b>Secretaria</b>	Persona encargada de transcribir al sistema los resultados de los exámenes entregados

	por el auxiliar o el tecnólogo. También puede consultar resultados de exámenes.
<b>4 Paciente</b>	Es la persona que accede al sistema para consultar resultados de exámenes y detalles de exámenes o servicios que ofrece el laboratorio. Para la consulta de los resultados el paciente necesita su número de cédula y una clave.

TABLA 22. DESCRIPCIÓN DE ACTORES

Elaborado por: El autor

### 2.1.3.5 Descripción de Casos de Uso

<i>Caso de Uso:</i>	<b>Registrar Pedido de Examen</b>	<i>ID:</i> 1
<i>Actor:</i>	Tecnólogo Médico / Auxiliar	
<i>Objetivo:</i>	Permitirá al tecnólogo médico o al auxiliar de laboratorio registrar los exámenes que el paciente desee realizarse.	
<i>Pre Condiciones:</i>	El paciente debe estar registrado.	
<i>Post Condiciones:</i>	<ul style="list-style-type: none"> <li>• El paciente queda registrado.</li> <li>• Se crea un código único para la muestra del paciente.</li> <li>• El pedido de exámenes queda registrado.</li> </ul>	
<i>Escenarios:</i>	<p>NORMAL</p> <p>10. Se ingresa la cédula del paciente que solicita el o los exámenes de laboratorio.</p> <p>20. El sistema despliega lista de exámenes que el laboratorio</p>	

ofrece agrupados por tipo de exámenes.

30. El tecnólogo médico o el auxiliar seleccionan los exámenes solicitados por el paciente.

40. El sistema informa el número de muestra asignado así como el número de resultado correspondiente a la muestra del paciente.

50. El sistema informa que el pedido ha sido guardado

#### EXCEPCION

10.1. El sistema indica que el paciente aún no ha sido registrado.

10.2 Se extiende el caso de uso Registrar Paciente.

10.3. Regresa a 10.

<i>Caso de Uso:</i>	<b>Registrar Paciente</b>	<i>ID:</i> 2
<i>Actor:</i>	Tecnólogo Médico / Auxiliar	
<i>Objetivo:</i>	Ingreso de los datos del paciente.	
<i>Pre Condiciones:</i>	<ul style="list-style-type: none"> <li>• El paciente no debe haber sido registrado previamente.</li> </ul>	
<i>Post Condiciones:</i>	<ul style="list-style-type: none"> <li>• El paciente queda registrado.</li> </ul>	
<i>Escenarios:</i>	<p>NORMAL</p> <p>10. Capturar nombres del paciente.</p> <p>20. Capturar apellidos del paciente.</p>	

<p>30. Capturar cédula del paciente.</p> <p>40. Capturar edad del paciente.</p> <p>50. Capturar dirección del paciente.</p> <p>60. Capturar teléfono del paciente.</p> <p>80. Termina caso de uso.</p>
--

<i>Caso de Uso:</i>	<b>Ingresar Resultados de Examen</b>	<i>ID: 3</i>
<i>Actor:</i>	Secretaria/Tecnólogo Médico	
<i>Objetivo:</i>	<p>Permitir a la secretaria o tecnólogo médico el registro de los resultados de los exámenes de laboratorio realizados a un determinado paciente, este proceso incluye:</p> <p>Selección de un paciente previamente registrado.</p> <p>Ingreso de los valores de los resultados de los análisis realizados a la muestra del paciente.</p>	
<i>Pre Condiciones:</i>	<ul style="list-style-type: none"> <li>• Debe existir un paciente registrado en el sistema al cual asociar los resultados.</li> <li>• El pedido debe estar registrado.</li> </ul>	
<i>Post Condiciones:</i>	<p>Los resultados de los exámenes descritos en el pedido quedan registrados.</p>	

*Escenarios:*

## NORMAL

- 10. Capturar código de muestra del paciente.
- 20. Capturar la cédula del paciente.
- 30. Generar lista de los exámenes registrados en el pedido para el paciente.
- 40. Asignar valores a cada uno de los exámenes solicitados.
- 50. Termina caso de uso.

## FLUJO ALTERNO 1

- 10.1 La muestra no está registrada en el sistema.
- 10.2. Termina caso de uso.

## FLUJO ALTERNO 2

- 20.1 El paciente no está registrado en el sistema.
- 20.2. Termina caso de uso.

<i>Caso de Uso:</i>	<b>Consultar Resultados de Examen</b>	<i>ID:</i> 4
<i>Actor:</i>	Secretaria/Paciente/Médico	
<i>Descripción:</i>	Despliega los resultados de los exámenes, una vez que hayan sido ingresados en el sistema.	

<p><i>Pre Condiciones:</i></p> <ul style="list-style-type: none"> <li>• Debe existir un pedido de exámenes asociado a la muestra de un paciente.</li> </ul>
<p><i>Post Condiciones:</i></p>
<p><i>Escenarios:</i></p> <p>NORMAL</p> <p>10. Capturar cédula del paciente.</p> <p>20. Capturar código asignado a la muestra.</p> <p>30. Sistema despliega resultados del examen.</p> <p>40. Termina caso de uso.</p> <p>FLUJO ALTERNO 1</p> <p>20.1. Mensaje “Paciente o muestra incorrecta”.</p> <p>20.2. Retorna a caso de uso <i>Consultar Resultados de Examen</i>.</p> <p>FLUJO ALTERNO 2</p> <p>30.1 El sistema despliega la lista de resultados sin ningún valor</p> <p>Debido a que aún no se han ingresado los resultados.</p>

#### 2.1.4 MATRIZ DE PRIORIZACIÓN

Como herramienta para determinar el framework más adecuado para el caso de aplicación utilizando los criterios que serán considerados y sus ponderaciones se utilizará una matriz de priorización.

Una matriz de priorización permite clasificar opciones, problemas o asuntos basados en un conjunto de criterios que permiten fijar la prioridad de los mismos, estos criterios son particulares dependiendo de la situación considerada.

La forma general de una matriz de priorización es la que se describe en la figura 2.5

Opciones	Criterios				Total
	C#1	C#2	C#3	C#4	
Opción 1					
Opción 2					
Opción 3					

**FIGURA 2.5. ESTRUCTURA GENERAL DE UNA MATRIZ DE PRIORIZACIÓN**

**Fuente:** <http://www.ongconcalidad.org/priorizacion.pdf>

La primera columna se compone de las opciones a ser priorizadas, en el caso particular del presente trabajo, esta columna estará compuesta por los frameworks a ser evaluados.

El resto de columnas se componen por los criterios descritos en la sección 2.1.

La columna final corresponde a los totales de cada opción, luego de ordenar las opciones de mayor a menor se obtendrá una matriz que refleje el framework más adecuado de acuerdo a los criterios.

## **2.2 ANÁLISIS DE LAS CARACTERÍSTICAS DE ADOBE FLEX**

### **2.2.1 APARIENCIA EN LOS DIFERENTES NAVEGADORES**

Flex hace uso de Flash Player para su despliegue en los navegadores y AIR en aplicaciones de escritorio, pero en ambos casos usan máquinas virtuales que son independientes tanto del sistema operativo como del navegador, con sus propias API's para la creación y manipulación de gráficos y recursos de hardware; por estas razones las aplicaciones desarrolladas sobre Flex se consideran independientes del navegador. Esta afirmación es lo más cercano a la realidad debido a que la independencia no es en un ciento por ciento ya que requiere de Flash Player, pero si consideramos que un promedio del 99%<sup>6</sup> de computadores con acceso a internet cuentan ya con este reproductor.

<sup>6</sup> [http://www.adobe.com/products/player\\_census/flashplayer/version\\_penetration.html](http://www.adobe.com/products/player_census/flashplayer/version_penetration.html)

**Valoración preliminar: 3**

### **2.2.2 INDEPENDENCIA DE LA RESOLUCIÓN**

Dado que las aplicaciones que corren sobre un navegador y construidas utilizando Adobe Flex utilizan las API's nativas de Flash, cuentan con las mismas características propias, entre estas la creación de gráficos vectoriales los cuales requieren menor cantidad de almacenamiento en disco, no dependen de la resolución de la pantalla ya que son calculadas en base a formulas matemáticas y que se complementan con técnicas de render basadas en pixeles para obtener una mejor calidad de imagen. Esto hace que la resolución no sea un factor determinante al momento de presentar la aplicación.

**Valoración preliminar: 3**

### **2.2.3 REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA EL DESARROLLO**

Considerando únicamente el SDK de Flex como el software necesario para el desarrollo, este es completamente independiente del sistema operativo, lo cual le otorgaría a Flex la mayor valoración para este criterio; sin embargo el uso de SDK únicamente trae consigo ciertas consideraciones, la primera de ellas la disminución en el costo por no adquirir el IDE y la segunda es la necesidad de suplir el IDE gráfico Flex Builder por un editor que mejore la productividad, la mejor opción sería utilizar la versión beta para GNU/Linux, lo cual resultaría en una dependencia a nivel de sistema operativo.

Tomando en cuenta ahora, el software adicional al framework, Flex requiere de la versión 1.5 del JDK (Java Development Kit), ya sea solo el SDK o el IDE, la dependencia de java sin embargo no presenta de ninguna manera adicional la dependencia de la plataforma.

Respecto al hardware para desarrollo, los requerimientos se podrían considerar estándar si tomamos en cuenta las características de los computadores actuales.

**Valoración preliminar: 4**

#### 2.2.4 COSTOS

Flex requiere de especial atención debido a que el SDK es totalmente libre y se puede desarrollar aplicaciones sin la compra de una licencia de desarrollo, sin embargo para el análisis se considerará el valor de la licencia de Flex Builder como IDE y del soporte anual para el producto la licencia tiene un costo de \$699 la versión Professional y el soporte de \$1495 dando un total de \$2194 dólares.

**Valoración final: 3**

#### 2.2.5 PATRONES DE DISEÑO

Considerando la descripción de Adobe Flex podemos notar que no propone formalmente un patrón de diseño, dejando abierta la posibilidad de ajustarse a propuestas de diseño existentes tales como MVC o MVP (Model View Presenter) o un desarrollo sin un patrón de diseño.

De las alternativas propuestas, este trabajo recogió la descripción de Cairngorm a la cual la comunidad ha denominado una micro arquitectura más que un patrón de diseño, como se pudo notar en la descripción, Cairngorm se basa en guías formales reconocidas (J2EE) lo cual le da un aspecto sólido y aceptado y propone una distribución de componentes orientada a objetos. Consideremos además que Cairngorm es un framework open source y que como tal tiene las ventajas y desventajas del propias.

**Valoración preliminar: 2**

#### 2.2.6 ESTANDARIZACIÓN DEL LENGUAJE

Adobe Flex usa como lenguajes para el desarrollo MXML y ActionScript 3, el primero es una extensión del estándar XML, lo cual le da ya un soporte amplio y facilidad de uso, el segundo es un lenguaje que para mencionada versión 3 se apega ya al estándar ECMA-226 para lenguajes de scripting y que sumada la amplia difusión con Flash tiene un gran nivel de aceptación.

Ambos lenguajes son propietarios, sin embargo existe una gran cantidad de documentación lo cual junto con el cumplimiento de estándares hace más sencilla su utilización.

**Valoración final: 2**

### 2.2.7 SOPORTE DEL PRODUCTO

Adobe Systems Incorporated es la empresa a cargo del soporte de Adobe Flex, la adquisición de la licencia da ciertas facilidades de soporte, ya sea mediante el sitio web, documentos digitales, e-mail o centros de atención al cliente, para soporte más especializado ofrece lo que denominan “Programas de Soporte”, los cuales se obtienen mediante el pago de estos servicios.

A este soporte se le debe agregar la gran comunidad de desarrolladores que utiliza el framework, comunidad que aumento en número desde la salida de Flex como open source y la cantidad de bibliografía existente en la actualidad.

En lo referente al uso del IDE Flex Builder o de los componentes, Adobe propone laboratorios prácticos complementados con videos tutoriales.

Entre los sitios más relevantes con documentación e información de Adobe Flex se puede mencionar:

<http://forums.adobe.com/community/flex> Sitio de foros para la comunidad de desarrolladores.

<http://livedocs.adobe.com/flex/3/html/help.html> Sitio oficial de ayuda y ejemplos de uso de componentes.

**Valoración final: 4**

### 2.2.8 ROAD MAP TECNOLÓGICO

Desde la compra de Adobe a Macromedia, Flex ha estado en constante renovación, lleva en el mercado aproximadamente 5 años y medio y actualmente se encuentra en la transición entre la tercera y cuarta versiones para Flex Builder así como para su SDK. Las versiones de Flex han tenido importantes variantes, entre las principales de ellas la liberación del código a nivel de SDK y la adopción de Eclipse como plataforma para la construcción del IDE Flex Builder, lo cual le ha permitido consolidarse dentro de la comunidad de software libre.

El punto más fuerte del Framework es tal vez el apoyo sobre el motor de Adobe Flash, lo cual lo hace prácticamente compatible con la totalidad de navegadores y además le proporciona la riqueza visual del mencionado motor. Dentro de las perspectivas ya cumplidas encontramos la independencia de ejecución en un navegador, mediante el empleo de Adobe Air se logro esta independencia. Gracias a esto las aplicaciones RIA desarrolladas en Flex pueden construirse con

los mismos componentes y funcionar sobre un navegador o instaladas directamente como aplicaciones de escritorio, sin afectar su rendimiento.

Considerando ahora las perspectivas encontramos entre ellas la puesta al mercado de la cuarta versión de Flex Builder, la cual se prevé unificará el desarrollo de aplicaciones RIA y animaciones Flash, se espera además Flex Builder consiga un funcionamiento óptimo sobre sistemas GNU/Linux.

**Valoración final: 4**

### **2.2.9 TIPO DE LICENCIAMIENTO**

De igual manera que Adobe ha optado por dejar al SDK de Flex bajo MPL (Mozilla Public License) y a Flex Builder como software privativo (incluso la distribución para sistemas GNU/Linux), las aplicaciones desarrolladas utilizando componentes de Flex tienen ambas opciones para licenciarse.

Flex de esta manera permite aportes de la comunidad sobre su SDK, sin dejar de lado la posibilidad de desarrollar software personalizado, o incluir componentes en software privativo.

La valoración se hará considerando el tipo de licenciamiento dual del framework.

**Valoración final: 2**

### **2.2.10 CURVA DE APRENDIZAJE**

Flex considera el aprendizaje como un eje en lo referente al soporte al usuario, propone talleres guiados mediante la utilización de videos, proporciona código fuente de ejemplo, componentes reutilizables; a esto se le debe sumar el gran soporte de la comunidad con foros, blogs y proyectos open source de librerías de componentes.

Flex gracias a las similitudes entre ActionScript Java y C# propone un tiempo de aprendizaje básico de entre una y dos semanas, el uso de Flex Builder es otro punto a favor gracias a la amplia difusión de Eclipse.

Para usuarios sin mayor experiencia en lenguajes de programación, la utilización a nivel de SDK resultará una ardua labor, para este tipo de usuarios sería recomendable la utilización del IDE el cual mediante el arrastre de componentes obtendrá una aplicación en pocos minutos.

De todas maneras si se desea una interacción con componentes transaccionales o con lógica de negocio más elaborada, se requería de un estudio más profundo y mayor experiencia en la creación de estos componentes.

**Valoración preliminar: 3**

### **2.2.11 HERRAMIENTAS DE DESARROLLO**

Adobe ha puesto gran énfasis y esfuerzo para la realización de un IDE que permita mejorar la productividad, construido sobre Eclipse Flex Builder permite el desarrollo tanto como para aplicaciones que usan AIR y Flash, el desarrollo de interfaces se realiza mediante la selección y arrastre de componentes en un entorno totalmente gráfico, permitiendo a la vez desarrollar componentes personalizados que pueden ser reutilizados.

Para interfaces de usuario se complementa la edición gráfica con edición de código fuente, de igual manera Flex Builder soporta la edición de código para generación de ActionScript.

Flex Builder ofrece además como una opción integrada la depuración de código, sin embargo no provee la opción de ejecución de pruebas unitarias.

Además el IDE desarrollado por Adobe existen otros proyectos como Aptana Studio y Spket que permiten la compilación y edición de aplicaciones pero únicamente a nivel de código, eso sí integrando el SDK propio de Flex.

**Valoración preliminar: 3**

### **2.2.12 FACILIDAD DE INTERACCIÓN CON OTRAS TECNOLOGÍAS**

Flex puede integrarse prácticamente con cualquier tipo de aplicaciones web mediante la generación de widgets que mejoren la presentación visual de las páginas, los widgets sin embargo solo actúan a nivel de presentación.

Si se considera a Flex únicamente como Front-End de una aplicación cuya lógica de negocio requiera de componentes más elaborados, existe una clara limitación si se toma en cuenta que no es posible realizarlos con elementos propios del framework; para resolver esta limitación Flex propone la utilización de web services, los cuales pueden estar desarrollados en cualquier lenguaje.

Para aplicaciones previamente desarrolladas y que se desean migrar para implementar interfaces desarrolladas en Flex, es necesario un gran esfuerzo para migrar toda la lógica de negocio y exponerla como web services.

**Valoración preliminar: 2**

### **2.2.13 FACILIDAD DE PUESTA EN PRODUCCIÓN**

Flex puede ser desplegado en cualquier servidor de aplicaciones capaz de soportar componentes Flash, si se trata de poner en producción una aplicación que corre sobre AIR, esta requiere de un certificado digital, el cual puede ser creado utilizando el IDE o escoger uno existente.

Sin embargo si se requiere la interacción con elementos de lógica de negocio, se puede utilizar Adobe ColdFusion o un servidor especializado para el lenguaje en el cual se desarrollara los mencionados componentes o un servidor compatible para acceder a web services externos.

Flex propone el uso del IDE para la puesta en producción de manera directa, sin embargo si se trata de un desarrollo en cual únicamente se utilice el SDK, este requiere que todos los recursos se encuentren bajo el mismo directorio. En ambos casos sobre cualquier servidor de aplicaciones Flex no requiere de configuraciones adicionales, salvo las concernientes a la interacción con servidores de componentes de lógica de negocio.

**Valoración preliminar: 2**

## **2.3 ANÁLISIS DE LAS CARACTERÍSTICAS DE JAVA RICH FACES**

### **2.3.1 INDEPENDENCIA DE LA RESOLUCIÓN**

De las consultas realizadas y la descripción del capítulo 1 no se encontraron alusiones explícitas sobre la independencia de la resolución de los componentes, es por esta razón que este criterio será valorado en su totalidad en la realización del prototipo.

**Valoración preliminar: 1**

### **2.3.2 APARIENCIA EN LOS DIFERENTES NAVEGADORES**

La apariencia de las aplicaciones desarrolladas con Rich Faces es controlada por su componente de skinability, el cual es registrado en el archivo de configuración principal y puede ser cambiado según las necesidades y gustos del cliente incluso

cuando la aplicación este completamente en producción. Esta característica sin embargo posee 2 niveles que se diferencian en la capacidad de personalización de los componentes, niveles que a su vez pueden ser diferenciados dependiendo del tipo de navegador desde el cual son accedidos.

Los niveles y navegadores compatibles son los siguientes:

Nivel Standard (personalización de propiedades básicas)

- Internet Explorer 6.
- Internet Explorer 7 en modo de compatibilidad.
- Opera.
- Safari.

Nivel Extendido (personalización de todas las propiedades)

- Mozilla Firefox
- Internet Explorer 7 en modo CSS1(Cascade Styte Sheet level 1), esto significa que acoge las recomendaciones de W3C (World Wide Web Consortium) definidas en el documento CSS1 en cuanto a la aplicación de estilos a contenido HTML.

De esta manera se desprende que Rich Faces no es completamente independiente del navegador desde el cual son accedidas las aplicaciones.

**Valoración preliminar: 2**

### **2.3.3 REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA EL DESARROLLO**

En lo referente al software para el desarrollo, Rich Faces requiere de las librerías de componentes descritas en la sección 1.2.3.4 y la versión 1.5 del JDK de java. Esto de manera consiguiente lo libera de la dependencia de un determinado sistema operativo, para mejorar la productividad en el desarrollo, Rich Faces propone la integración con diversos IDE's, los cuales a su vez son independientes de la plataforma sobre la cual se ejecutan. Seleccionado el IDE, Rich Faces no requiere de software adicional para el desarrollo.

**Valoración preliminar: 4**

### **2.3.4 COSTOS**

Rich Faces no tiene costo de licenciamiento, Red Hat no ofrece un plan de soporte específico para Rich Faces como lo hace para sus sistemas operativos y software de middleware, sin embargo existe una modalidad de soporte como desarrollador la cual tiene un costo estándar de \$3500.

**Valoración final 2.**

### **2.3.5 PATRONES DE DISEÑO**

Rich Faces permite de manera nativa implementar el patrón de diseño MVC, el es un patrón ampliamente difundido y aceptado brindando el soporte adecuado a las aplicaciones desarrolladas con este framework. Los elementos que constituyen cada uno de los componentes de MVC se encuentran plenamente identificados dentro de la especificación JSF la cual a su vez ya es un estándar completamente aceptado.

De esta manera podemos estar seguros que la implementación del patrón de diseño MVC con Rich Faces estará totalmente asegurada; esto no quiere decir que no exista la posibilidad de implementar otros patrones de diseño, de hecho JSF y Rich Faces tienen como característica la flexibilidad en el diseño.

**Valoración Preliminar 4.**

### **2.3.6 ESTANDARIZACIÓN DEL LENGUAJE**

Rich Faces usa lenguajes como Java y HTML para el desarrollo de sus aplicaciones, ambos son estándares y con un aceptado reconocimiento a nivel mundial.

Java como lenguaje ha sido el promotor de varios estándares, no solo a nivel de lenguaje sino de patrones de arquitectura, varias de sus especificaciones como JSF han servido para el desarrollo de frameworks completos como el sujeto de estudio Rich Faces, a esto se le debe sumar el constante refinamiento del JDK y su independencia de plataforma.

HTML puede considerarse un lenguaje completamente libre, Java se puede considerar también un lenguaje libre ya que permite la creación de JDK's propios así como JVM's personalizadas.

**Valoración final: 4**

### **2.3.7 SOPORTE DEL PRODUCTO**

Rich Faces como producto de JBoss cuenta con el soporte de esta organización, mediante el contrato de estos servicios, ya sea en la modalidad desarrollador o empresarial.

JBoss acoge las sugerencias y aportes de la comunidad para mejorar y obtener nuevas versiones de Rich Faces, de igual manera Exadel figura aún como la organización encargada de la capacitación y seminarios sobre Rich Faces.

Notamos de esta manera que JBoss es formalmente la organización a cargo del soporte del producto, lanzamiento de nuevas versiones y corrección de errores; a esto se le debe sumar el respaldo de Red Hat como dueño de JBoss y de la comunidad de desarrolladores que contribuyen al mejoramiento del framework.

**Valoración final: 4**

### **2.3.8 ROAD MAP TECNOLÓGICO**

Rich Faces como proyecto lleva 4 años en el mercado, desde sus inicios como software privativo hasta la actualidad como software libre Rich Faces ha tenido amplias expectativas como uno de los frameworks que implementando la especificación de JSF ha dotado de una gran riqueza visual a sus aplicaciones. Las perspectivas del proyecto es mantener la compatibilidad con futuras especificaciones de JSF, mediante el refinamiento y mejora de componentes. Rich Faces además tiene como objetivo apoyar a la investigación manteniendo el proyecto como open source y ofreciendo un espacio para los aportes de la comunidad de desarrolladores.

**Valoración final: 4.**

### **2.3.9 TIPO DE LICENCIAMIENTO**

Rich Faces utiliza únicamente el licenciamiento del tipo LGPL, lo cual le permite formar parte de software privativo.

**Valoración final: 2**

### **2.3.10 CURVA DE APRENDIZAJE**

Rich Faces requiere de un conocimiento medio por lo menos de las especificaciones de los estándares JEE, esto implica una inversión de tiempo considerable, no es recomendable el uso por parte de usuarios novatos, debido a que requiere del conocimiento de la estructura de las aplicaciones web

desarrolladas con Java, modificación del contenido de archivos de configuración, empleo e interacción con componentes empresariales.

Rich Faces requiere de especial atención justamente por el cumplimiento de la especificación de JSF, la cual forma parte de una arquitectura orientada a aplicaciones empresariales generalmente grandes, lo cual le añade un grado mayor de complejidad.

**Valoración preliminar: 1**

### **2.3.11 HERRAMIENTAS PARA EL DESARROLLO**

Rich Faces cuenta con un gran soporte de herramientas para el desarrollo, se integra mediante plugins en Eclipse y Netbeans y de manera nativa son soportadas en JBoss Developer Studio, el uso de estos IDE's le hacen extensivo las herramientas de debug, test, control de versiones, etc.

Tal vez el punto negativo sea la falta de editores ciento por ciento gráficos, tanto como para eclipse y Netbeans la integración se da en una paleta de componentes que son arrastrados a nivel de código, lo cual si bien es mejor que la edición simple de código no llega al nivel de una edición de interfaces totalmente gráfica.

**Valoración preliminar: 3**

### **2.3.12 FACILIDAD DE INTERACCIÓN CON OTRAS TECNOLOGÍAS**

Rich Faces forma parte de una especificación que le permite interactuar con aplicaciones de backend desarrolladas en lenguaje java de manera nativa, sin embargo mediante el consumo de web services puede interactuar con otras tecnologías de backend.

Puede integrarse además con tecnologías de front-end como javascript, e incorpora AJAX igualmente de forma natural. El contenido multimedia se lo hace gracias a la etiqueta < aj4:mediaOutput >, la cual soporta videos, archivos de sonido, fotografías, etc.

**Valoración preliminar: 2**

### **2.3.13 FACILIDAD DE PUESTA EN PRODUCCIÓN**

Rich Faces es compatible con casi la totalidad de servidores de aplicaciones que soportan aplicaciones JEE, la portabilidad entre servidores y la independencia de los mismos depende de los distintos archivos de configuración que ligan a la aplicación a uno u otro servidor. Los diferentes IDE's permiten poner en

producción las aplicaciones mediante simples asistentes, sin embargo el mecanismo más utilizado y flexible es la puesta en producción mediante la ejecución de archivos ant.

**Valoración preliminar: 2**

## **2.4 ANÁLISIS DE LAS CARACTERÍSTICAS DE EXT JS**

### **2.4.1 APARIENCIA EN LOS DIFERENTES NAVEGADORES**

Ext como solución para mantener la apariencia de sus widgets en los diferentes navegadores, usa una imagen de 1 pixel por 1 pixel de tamaño. Esto debido a que una imagen es el único elemento cuyo tamaño es predecible de manera exacta en todos los navegadores. A esta imagen se le conoce como Spacer la cual de manera conjunta con la CSS mantiene el look & feel de los widgets de Ext.

**Valoración preliminar: 4**

### **2.4.2 INDEPENDENCIA DE LA RESOLUCIÓN**

De igual manera que para mantener la apariencia en los diferentes navegadores se utiliza la imagen Spacer, esta también se utiliza para conservar la relación de tamaños y espacios entre componentes, en las diferentes resoluciones de pantalla desde las cuales se accede a la aplicación. Con esta solución evita en parte la limitación de no representar los gráficos de forma vectorial.

**Valoración preliminar: 4**

### **2.4.3 REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA EL DESARROLLO**

Ext Js requiere únicamente de su SDK para el desarrollo de aplicaciones, se puede usar cualquier editor de texto para escribir las aplicaciones, esto representa una severa disminución de productividad al igual que si se usara únicamente el SDK de Flex, a diferencia de este Ext Js no posee un IDE que le de soporte de manera oficial; sin embargo varios proyectos como Aptana Studio y Spket contruidos sobre Eclipse proveen de un IDE que al menos a nivel de código soportan de manera nativa el desarrollo de aplicaciones utilizando Ext Js de manera exclusiva.

Considerando únicamente el SDK, este no requiere de ningún tipo de software adicional, lo cual lo hace totalmente independiente de la plataforma y de todo software adicional, a diferencia de Flex cuyo SDK depende del JDK de java. Si se trata de elegir un IDE que ayude a mejorar la productividad, entonces se presenta una dependencia del JDK de java, lo cual no afecta la condición de independencia de la plataforma.

#### **Valoración preliminar 4**

##### **2.4.4 COSTOS**

Ext Js tiene un costo de \$509 dólares entre el costo de licencia y soporte.

#### **Valoración final 4.**

##### **2.4.5 PATRONES DE DISEÑO**

Ext Js no considera un patrón de diseño para el desarrollo de aplicaciones, esto deja abierta la posibilidad de acoplarlo acorde con las necesidades del equipo o de la persona a cargo del desarrollo, y aún más importante permite integrarlo con cualquier lenguaje como back-end, y utilizar los patrones que soporten dichos lenguajes.

Bajo estas consideraciones se nota que Ext es mayoritariamente un framework destinado a la presentación o de front-end.

A estas observaciones se le debe agregar los patrones propuestos para el desarrollo de componentes, para los cuales existe también una amplia gama de opciones, todas ellas validas y aplicables.

#### **Valoración preliminar: 1**

##### **2.4.6 ESTANDARIZACIÓN DEL LENGUAJE**

ExtJs se deriva totalmente de JavaScript, lo cual le brinda las mismas capacidades y compatibilidad con el estándar ECMA-226. Las extensiones de funcionalidad implementadas por Ext Js como la manipulación del DOM, se apegan a las funcionalidades estándar con el afán de no afectar la compatibilidad, sino más bien simplificar el empleo del mismo. Pese a estas consideraciones hace falta la certificación por parte de ECMA.

#### **Valoración preliminar: 3**

#### **2.4.7 SOPORTE DEL PRODUCTO**

Ext es la compañía a cargo de Ext Js, los miembros de la compañía han participado desde la salida al mercado del producto como un agregado a YUI. El negocio de la compañía está totalmente ligado al soporte de sus productos, para esto existen varias modalidades y beneficios dependiendo de la opción de soporte contratada.

La adquisición de una licencia asegura ya el soporte del producto pero a un nivel limitado, si se quiere se puede optar por seleccionar un paquete con mayores beneficios, los cuales llegan incluso a la respuesta y solución de problemas en cuestión de horas. Además tiene posibilidad de utilizar el framework con fines educativos, existe también una comunidad que ofrece apoyo y soluciones a diversos inconvenientes.

**Valoración final: 4**

#### **2.4.8 ROAD MAP TECNOLÓGICO**

Ext es el framework más reciente en el mercado con casi tres años en el mercado, nace como extensión de YUI, lo cual lo hace compatible con otras librerías como prototype. Dentro de la evolución de Ext Js el punto más débil es la incompatibilidad con la primera versión del framework, cuestión que ocasiono la salida del mercado de la misma, sin embargo las versiones actuales son compatibles con las posteriores y se prevé as versiones futuras conserven esta característica. En el transcurso del año se espera la tercera versión del framework, la cual mejorará los controles existentes, así como la inclusión de nuevos con nuevas características.

**Valoración final: 3**

#### **2.4.9 TIPO DE LICENCIAMIENTO**

ExtJs ofrece un licenciamiento dual, licencia comercial y licencia de software libre. Dependiendo del número de desarrolladores las licencias comerciales oscilan entre los \$259 dólares hasta los \$17,000 dólares, con esto se asegura la posibilidad de venta del producto final desarrollado con el framework. A su vez la licencia de software libre es la GPL, lo cual asegura que bajo esta modalidad no se podrá comercializar el software.

**Valoración final: 1**

#### **2.4.10 CURVA DE APRENDIZAJE**

Ext basado completamente en el lenguaje JavaScript, considera que con experiencia en lenguajes como Java o C# requiere de un esfuerzo moderado para conseguir un manejo adecuado de sus componentes, es ideal para la interacción con PHP y aprovecha la capacidad de este para manipular datos.

Si se requiere de integración para alto soporte transaccional, puede integrarse en aplicaciones Java y .Net, sin embargo el proceso es más elaborado y requiere de mayor tiempo de aprendizaje. El soporte de foros es el principal medio de aprendizaje, varios ejemplos y tutoriales completan el panorama, sin embargo no existe un número considerable de libros que aporte al aprendizaje del framework.

**Valoración preliminar: 2**

#### **2.4.11 HERRAMIENTAS PARA EL DESARROLLO**

Ext Js no se provee de un IDE oficial que permita un ágil desarrollo de las interfaces de usuario final. Si se está usando Ext Js en conjunto con otra tecnología se puede utilizar el IDE de dicha tecnología para crear componentes de Ext únicamente a nivel de código y sin la opción de corrección de sintaxis.

Si se trata de desarrollar componentes o aplicaciones utilizando Ext de manera exclusiva, existen algunos proyectos como Aptana Studio y Spket que construidos sobre Eclipse integran Ext Js para la edición a nivel de código, incluyendo la corrección de sintaxis y sugerencia de código. Para la depuración de código y ejecución de pruebas, no se encontraron especificaciones en los IDE's existentes para la realización de estas tareas, por esta razón es necesario utilizar herramientas como firebug incluida en Mozilla Firefox para la depuración de código.

**Valoración preliminar: 2**

#### **2.4.12 FACILIDAD DE INTERACCIÓN CON VARIAS TECNOLOGÍAS**

Ext puede integrarse con cualquier aplicación realizada prácticamente en cualquier lenguaje, mediante la inclusión de las librerías necesarias en la cabecera del código html.

Ext es un framework orientado totalmente a la presentación, por esta razón requiere apoyarse en lenguajes que permitan el manejo de datos y lógica de negocio, de manera nativa no es posible la manipulación de datos salvo que estos

se encuentren en archivos XML o a través de web services tipo JSON (JavaScript Object Anotation). La integración con web services se requiere de una comprensión profunda y la correcta creación de las clases y métodos necesarios para consumir estos recursos, requiriendo un gran esfuerzo.

**Valoración preliminar: 2**

#### **2.4.13 FACILIDAD DE PUESTA EN PRODUCCIÓN**

Una aplicación desarrollada completamente en Ext Js requiere únicamente de la copia del directorio de la solución en la ubicación correspondiente del servidor de aplicaciones.

Si se trata de una aplicación que interactúa con otra tecnología, entonces la facilidad de puesta en producción estará ligada a la facilidad con la que dicha tecnología es puesta en producción.

Se nota que existe un comportamiento dependiente del tipo de aplicación para la puesta en producción, y esto se valorará luego del desarrollo del prototipo.

**Valoración preliminar: 2**

## CAPITULO 3. DESARROLLO DE PROTOTIPOS

### 3.1 PROTOTIPO USANDO ADOBE FLEX

#### 3.1.1 REGISTRAR PEDIDO DE EXAMEN

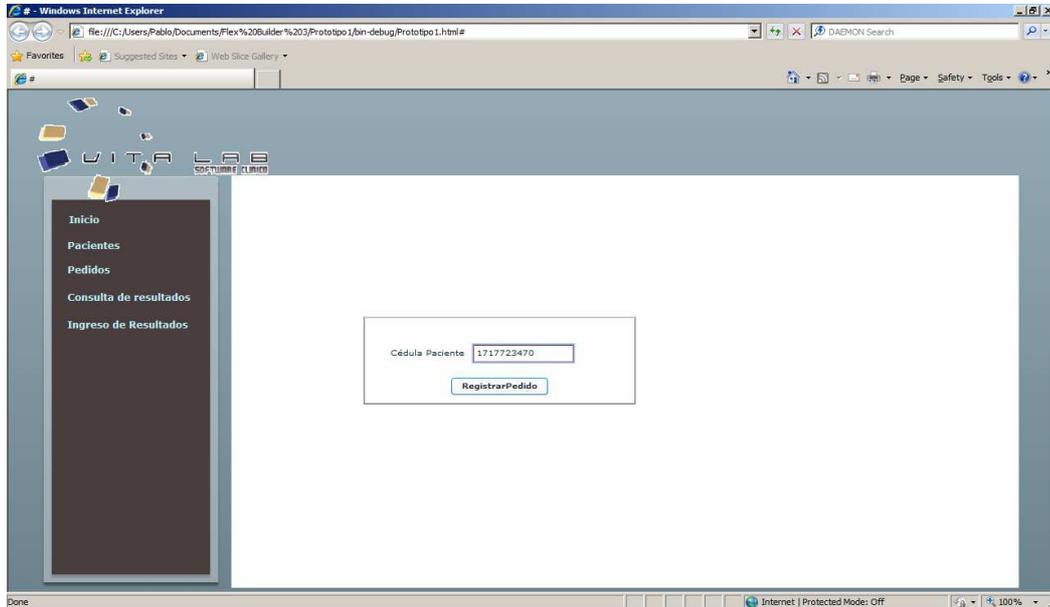


FIGURA 3.1 PANTALLA DE INGRESO DE CÉDULA DEL PACIENTE

Fuente: El Autor

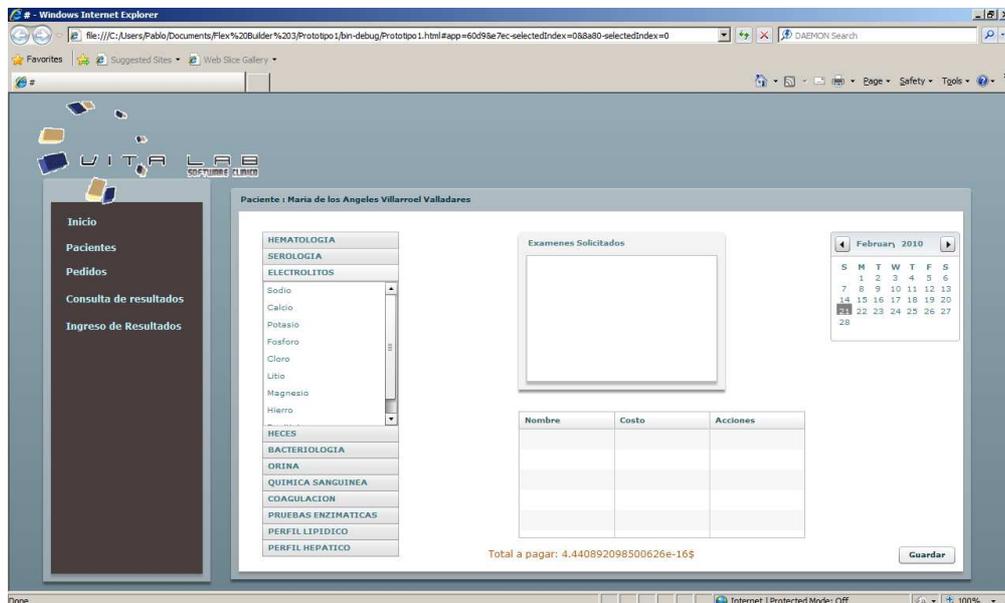


FIGURA 3.2 PANTALLA DE DESPLIEGUE DE LISTA DE EXÁMENES

Fuente: El Autor

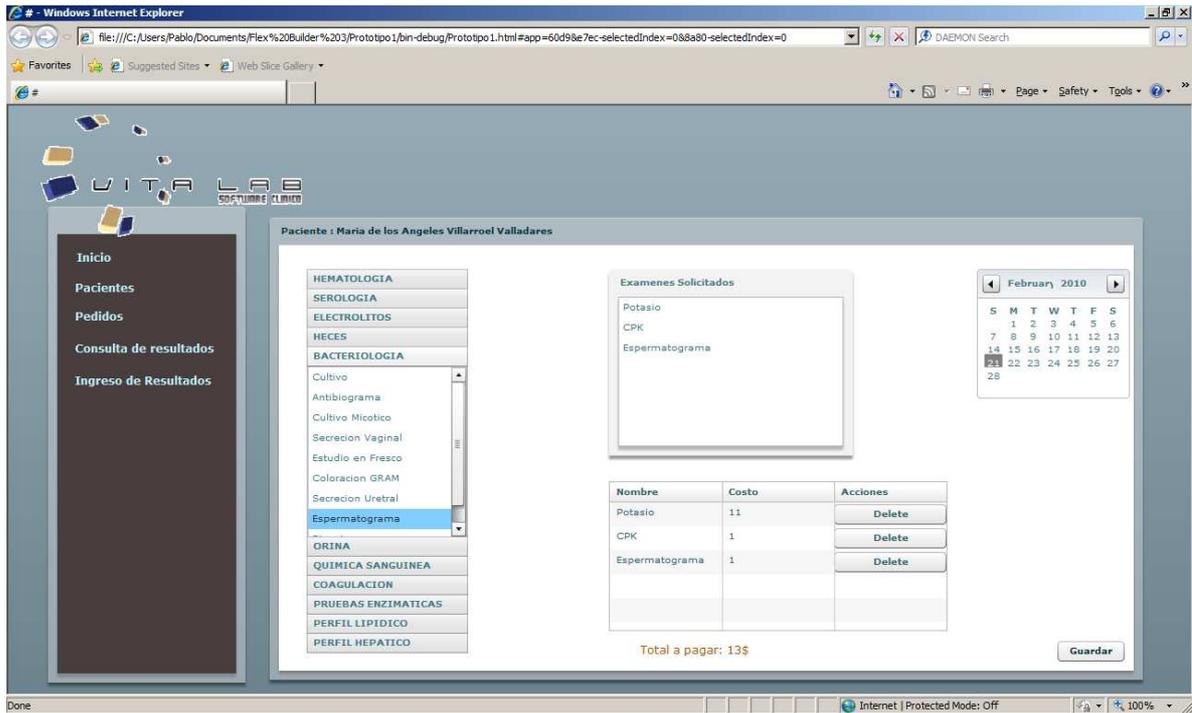


FIGURA 3.3 PANTALLA DE SELECCIÓN DE EXÁMENES SOLICITADOS

Fuente: El Autor

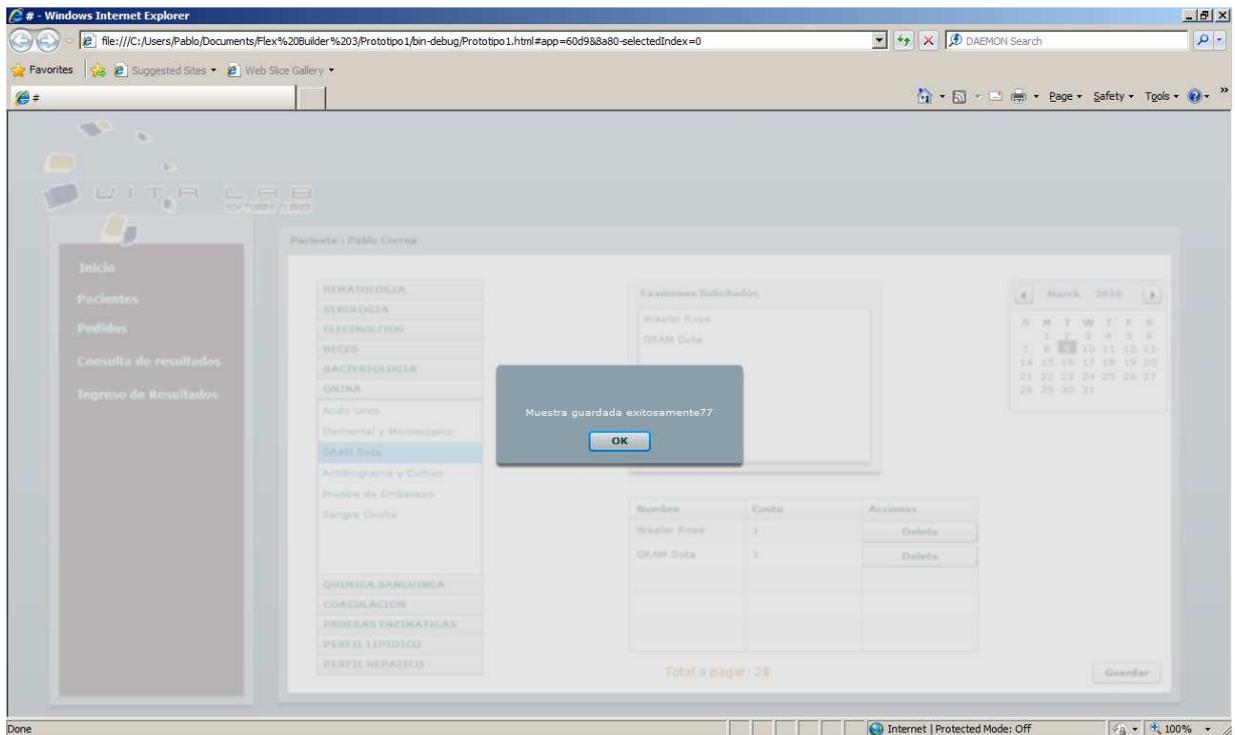
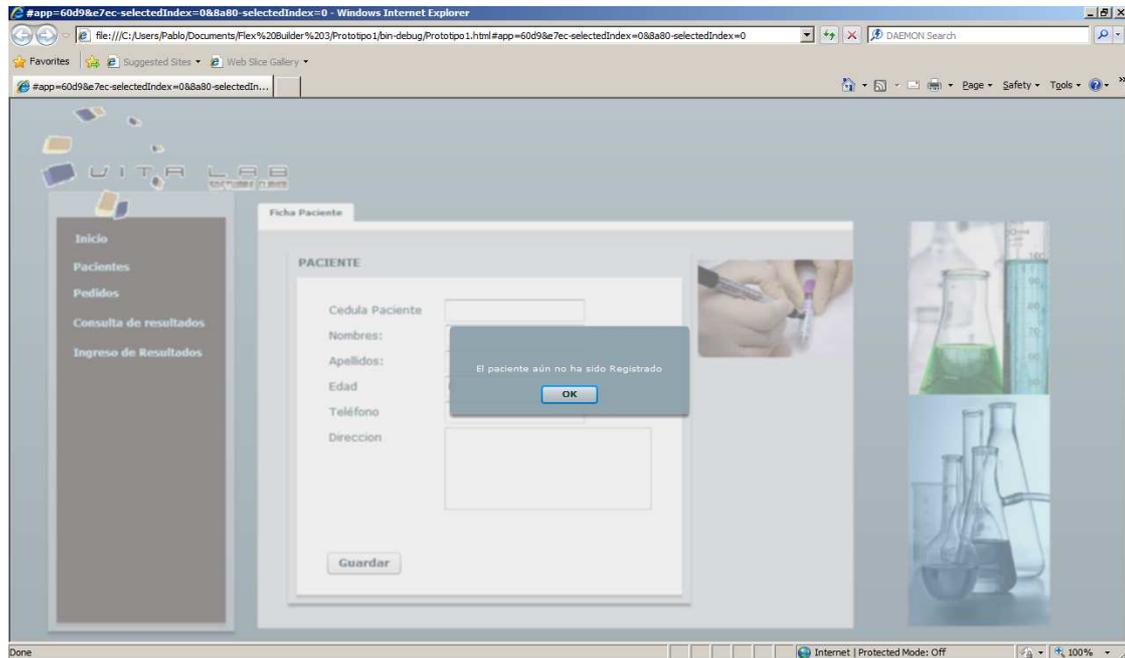


FIGURA 3.4 PANTALLA DE PEDIDO HA SIDO GUARDADO

Fuente: El Autor

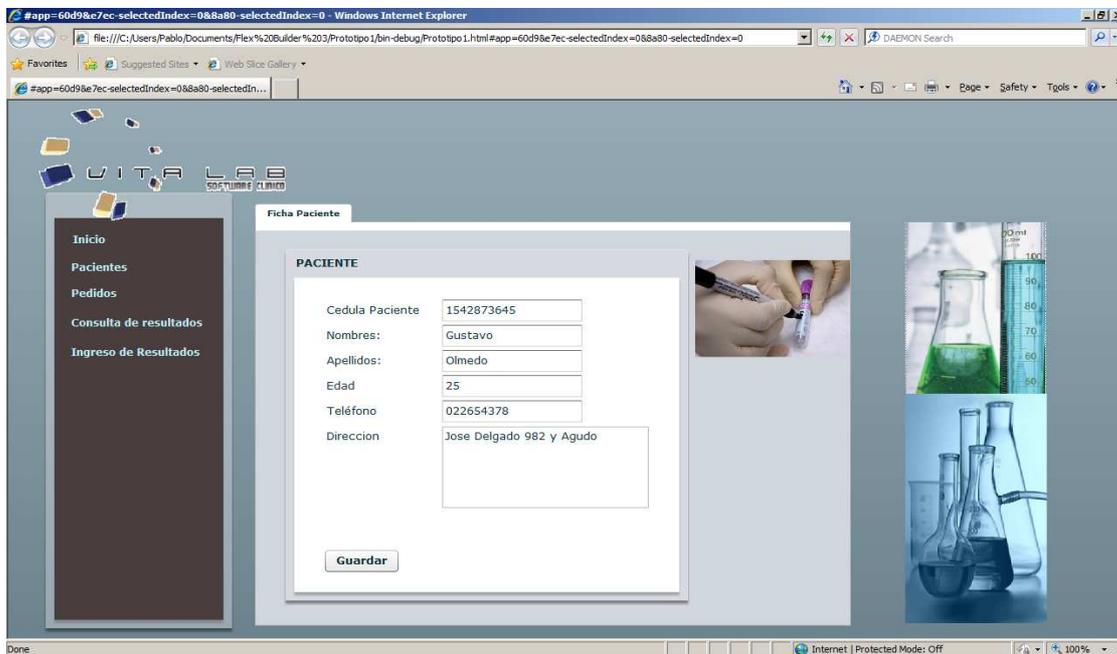
## Flujo de Excepción Paciente no registrado



**FIGURA 3.5 PANTALLA DE EXCEPCIÓN PACIENTE NO HA SIDO REGISTRADO**

**Fuente:** El Autor

### 3.1.2 REGISTRAR PACIENTE

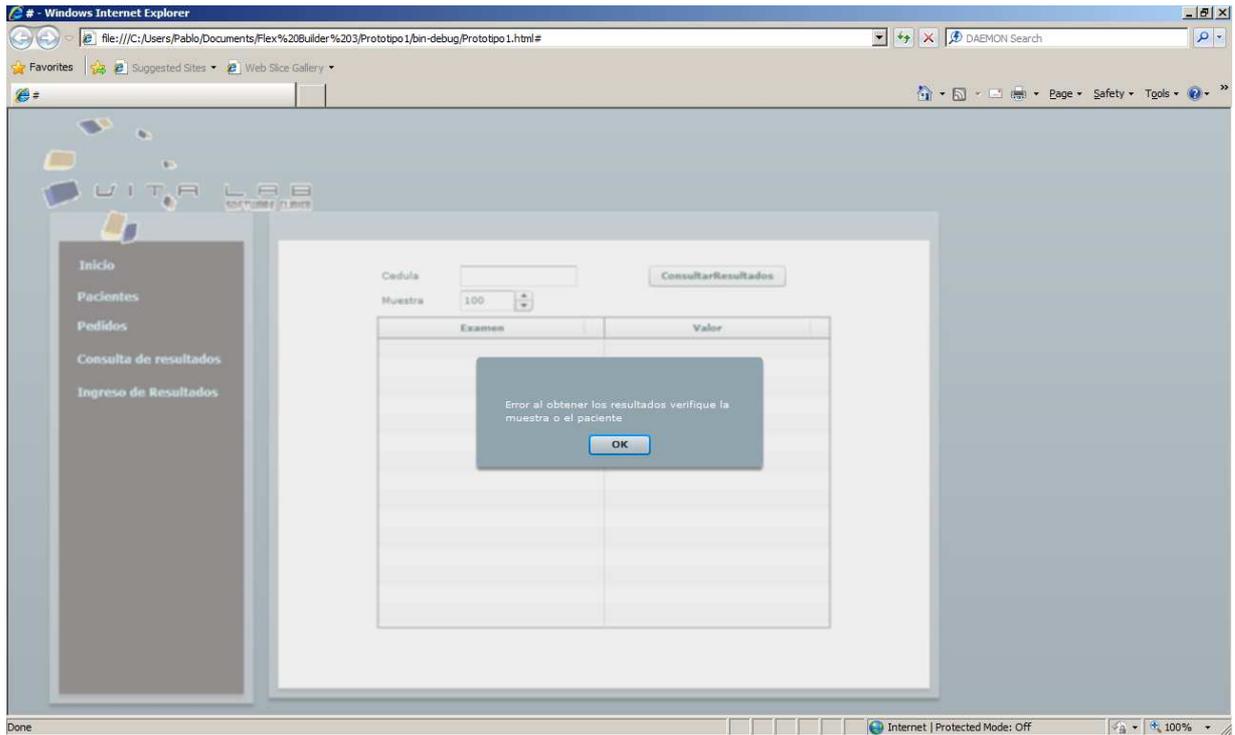


**FIGURA 3.6 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE**

**Fuente:** El Autor



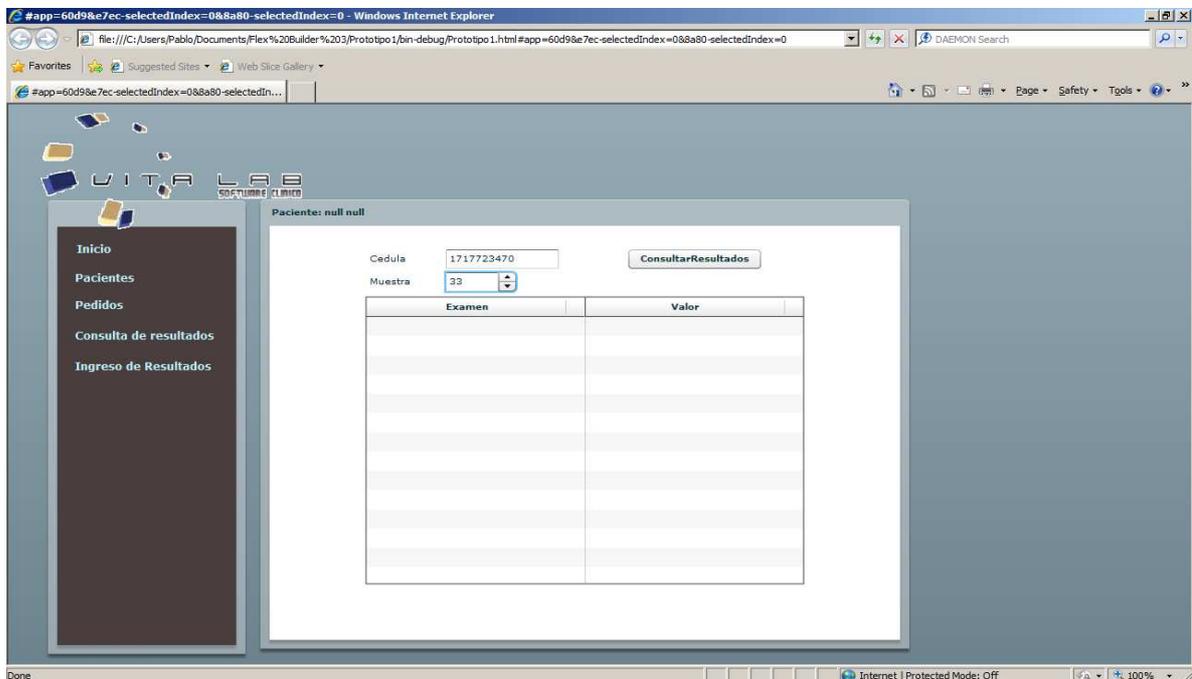




**FIGURA 3.11. PANTALLA DEL FLUJO ALTERNO2 PACIENTE NO ESTÁ REGISTRADO**

**Fuente:** El Autor

### 3.1.4 CONSULTAR RESULTADOS DE EXAMEN



**FIGURA 3.12 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE Y NÚMERO DE MUESTRA**

**Fuente:** El Autor

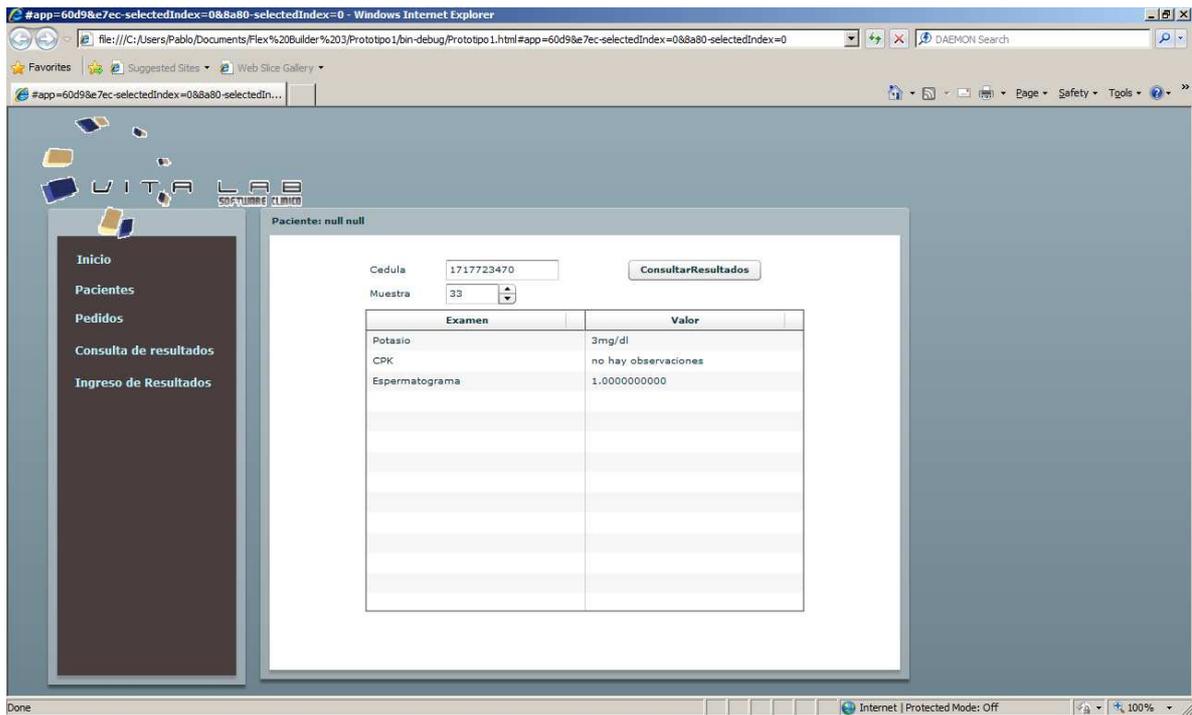


FIGURA 3.13 PANTALLA DE DELPLIEGUE DE RESULTADOS EXAMEN

Fuente: El Autor

## Flujo alternativo 1

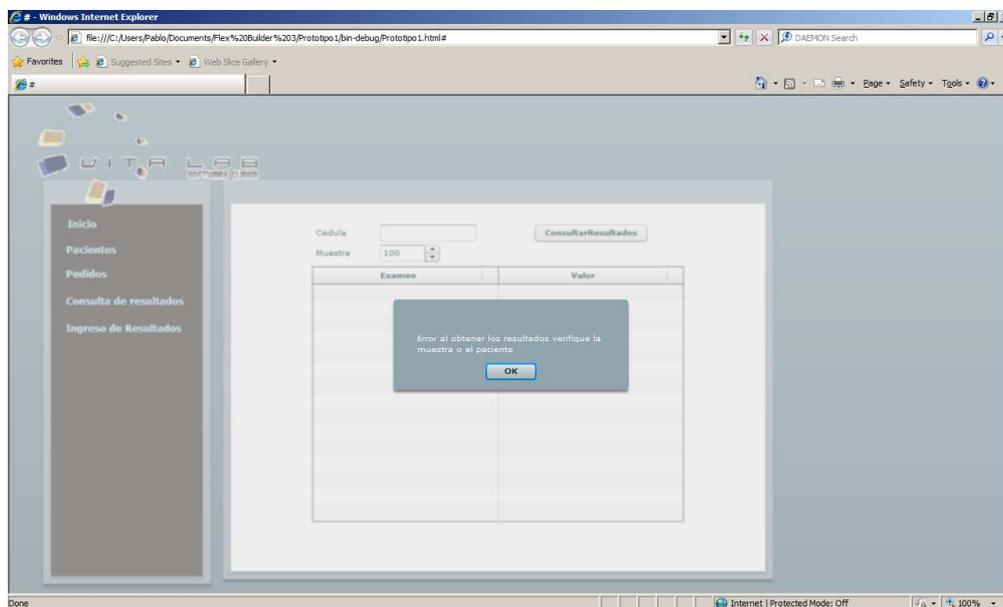


FIGURA 3.14. PANTALLA DE MENSAJE "PACIENTE O MUESTRA INCORRECTA"

Fuente: El Autor

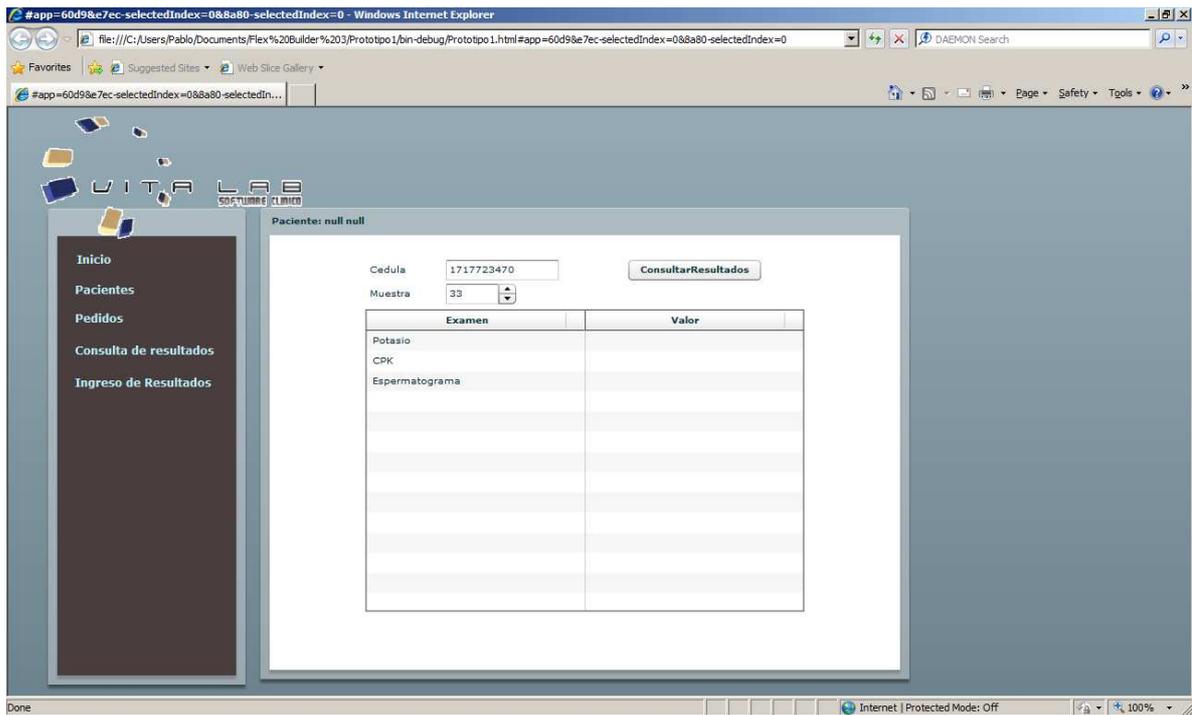


FIGURA 3.15 PANTALLA DE DESPLIEGUE DE LISTA DE RESULTADOS SIN VALORES

Fuente: El Autor

## 3.2 PROTOTIPO USANDO JAVA RICH FACES

### 3.2.1 REGISTRAR PEDIDO DE EXAMEN

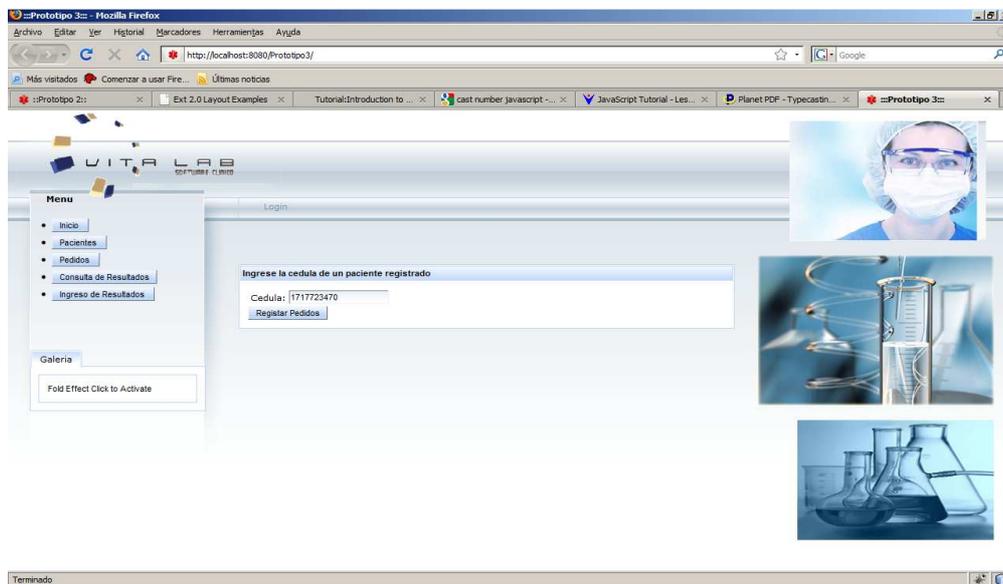


FIGURA 3.16 PANTALLA DE INGRESO DE CÉDULA DEL PACIENTE

Fuente: El Autor

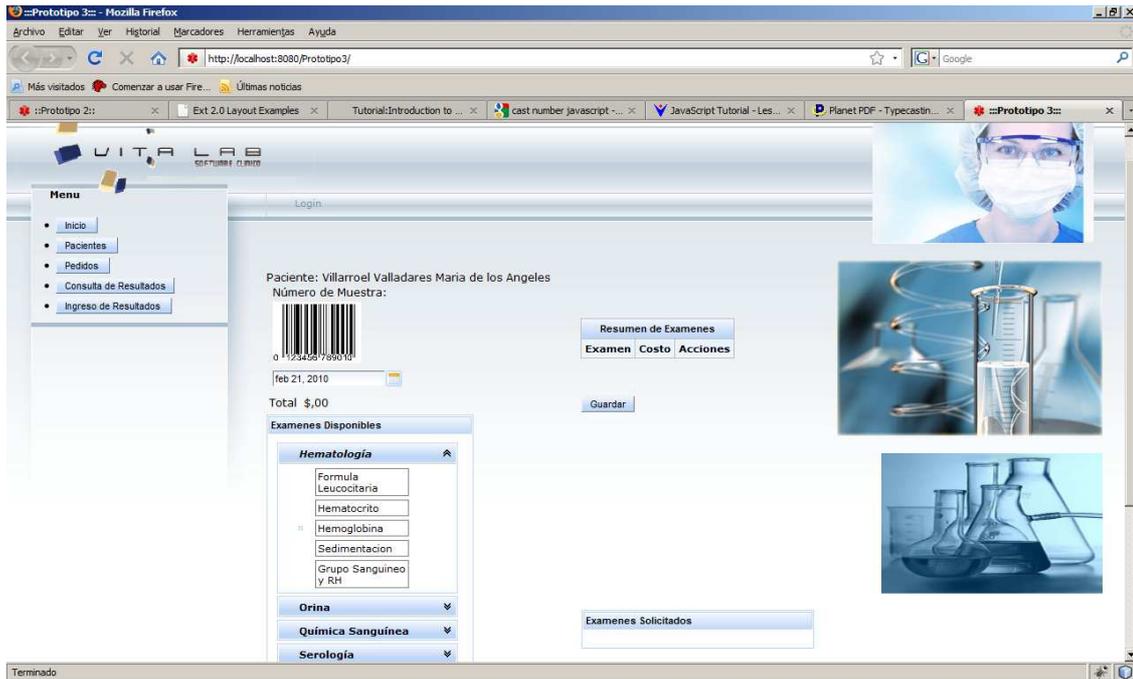


FIGURA 3.17 PANTALLA DE DESPLIEGUE DE LISTA DE EXÁMENES

Fuente: El Autor

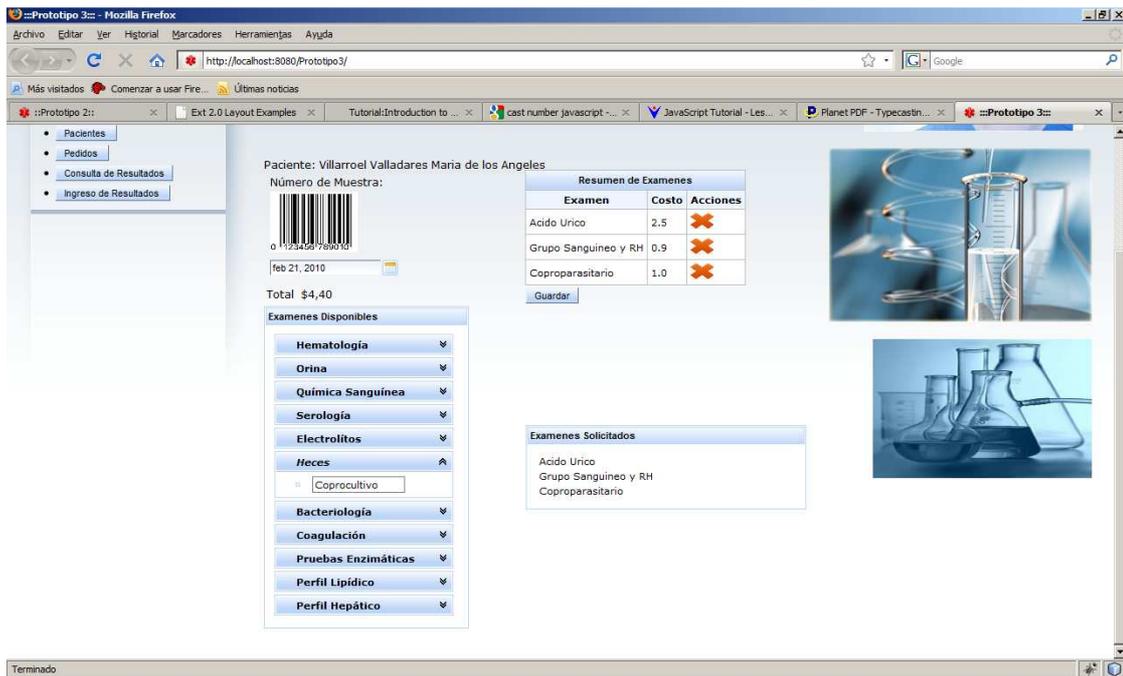
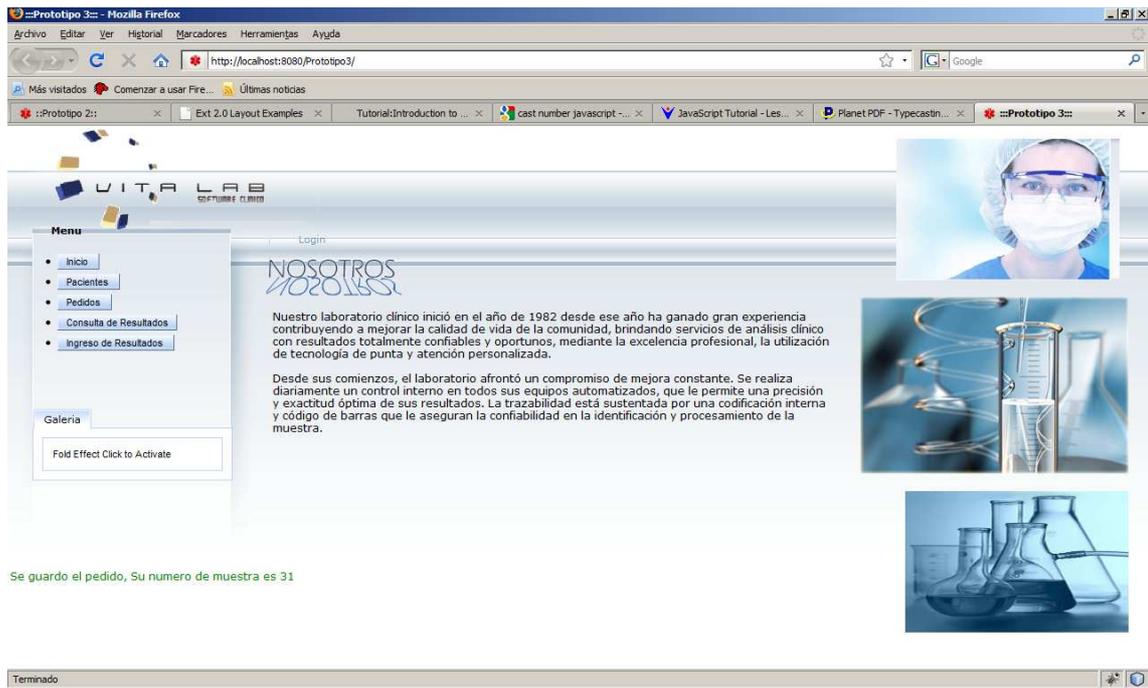


FIGURA 3.18. PANTALLA DE SELECCIÓN DE EXÁMENES SOLICITADOS

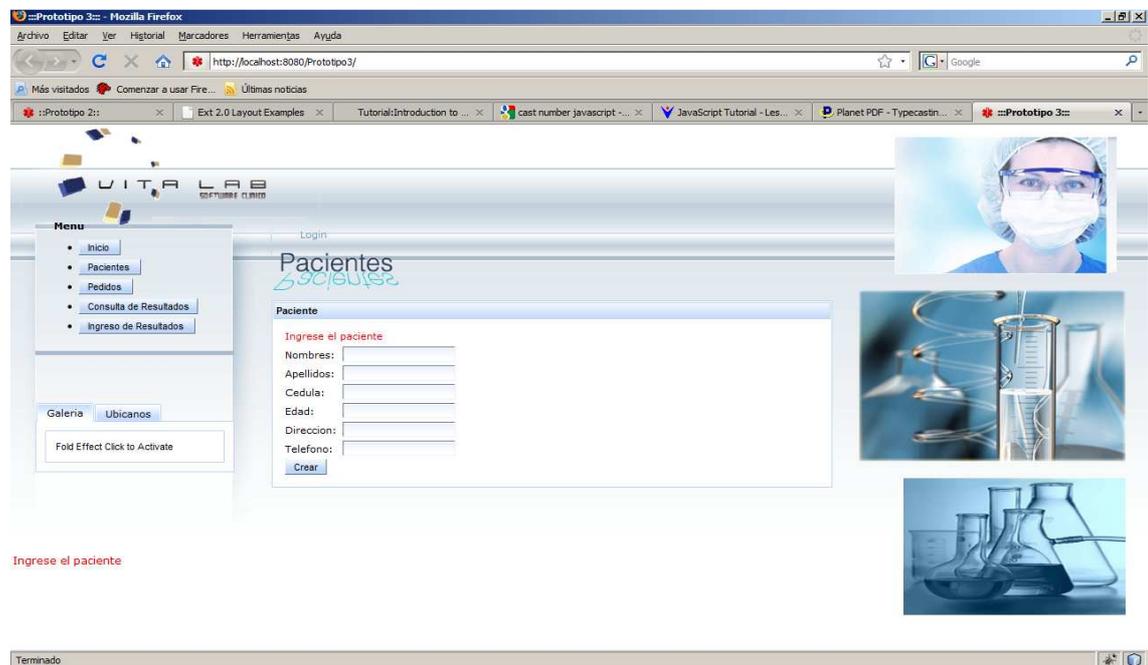
Fuente: El Autor



**FIGURA 3.19. PANTALLA DE MENSAJE “PEDIDO HA SIDO GUARDADO”**

**Fuente:** El Autor

### Flujo de Excepción Paciente no registrado



**FIGURA 3.20 PANTALLA DE EXCEPCIÓN PACIENTE NO HA SIDO REGISTRADO**

**Fuente:** El Autor

### 3.2.2 REGISTRAR PACIENTE

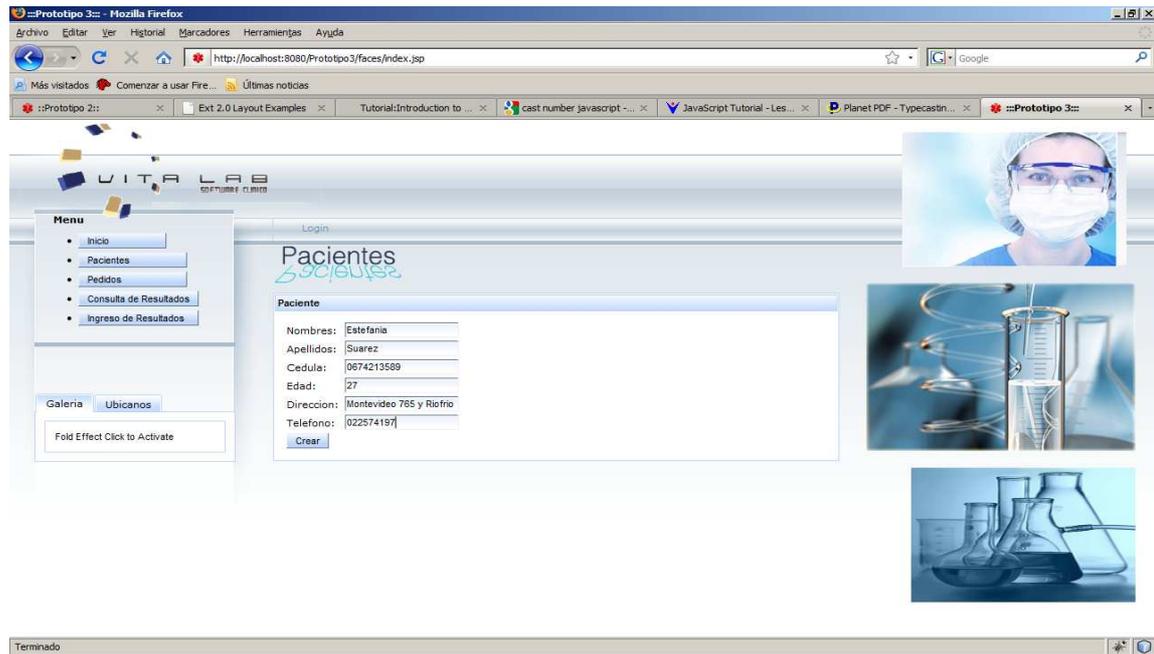


FIGURA 3.21 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE

Fuente: El Autor

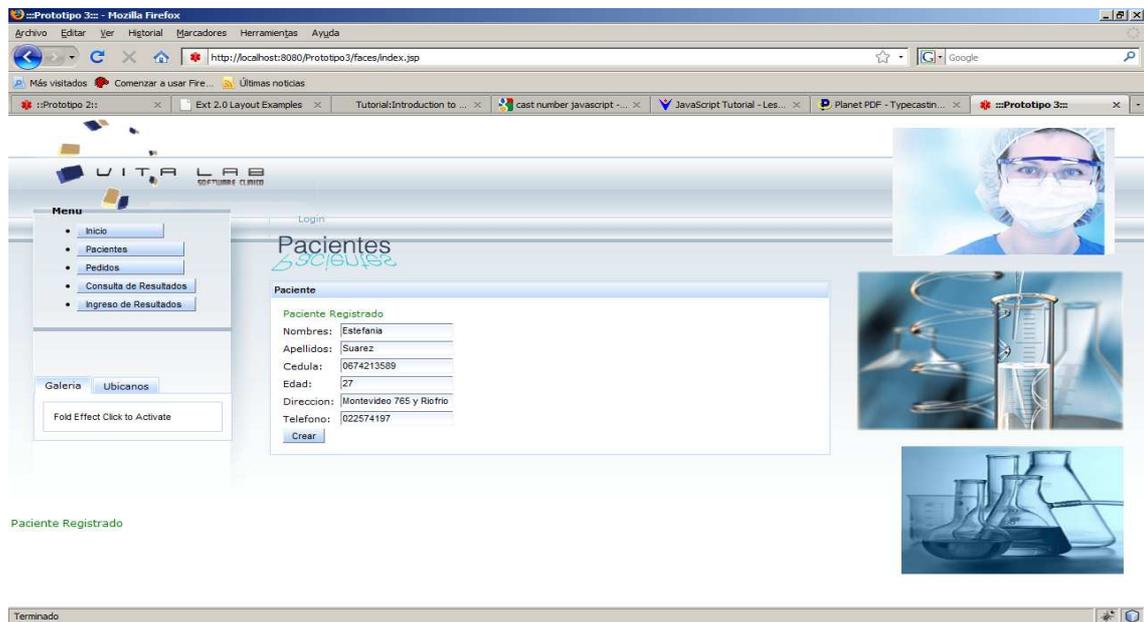


FIGURA 3.22 PANTALLA DE PACIENTE GUARDADO

Fuente: El Autor

### 3.2.3 INGRESAR RESULTADOS DE EXAMEN

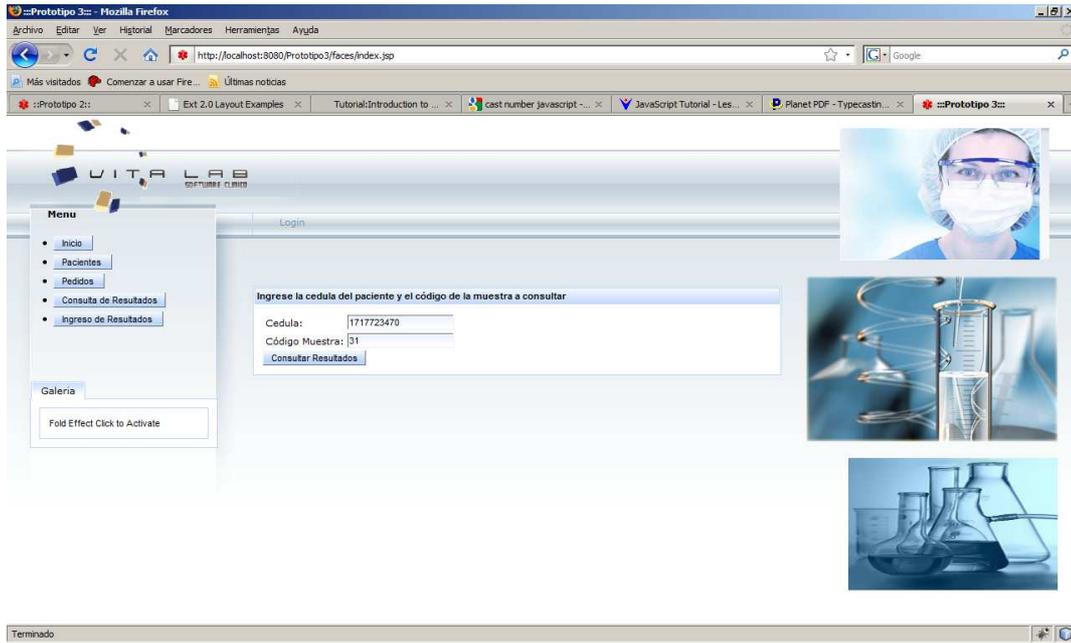


FIGURA 3.23 PANTALLA DE CAPTURA DE CÓDIGO DE MUESTRA

Fuente: El Autor

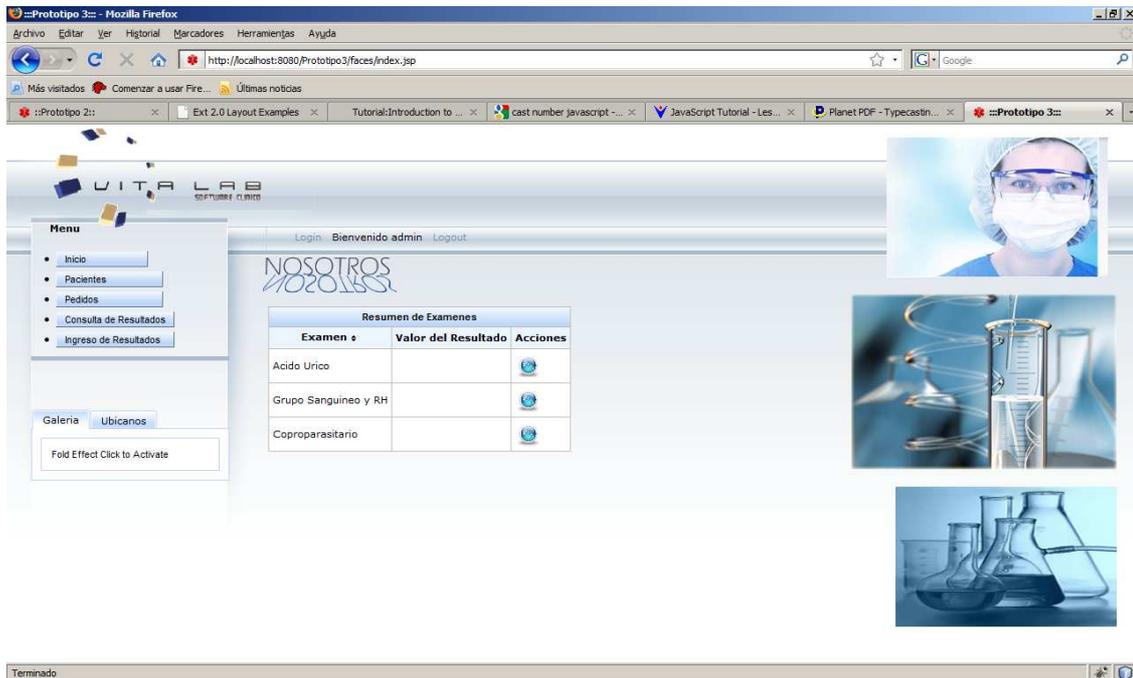


FIGURA 3.24 PANTALLA DE LISTA DE EXÁMENES REGISTRADOS

Fuente: El Autor

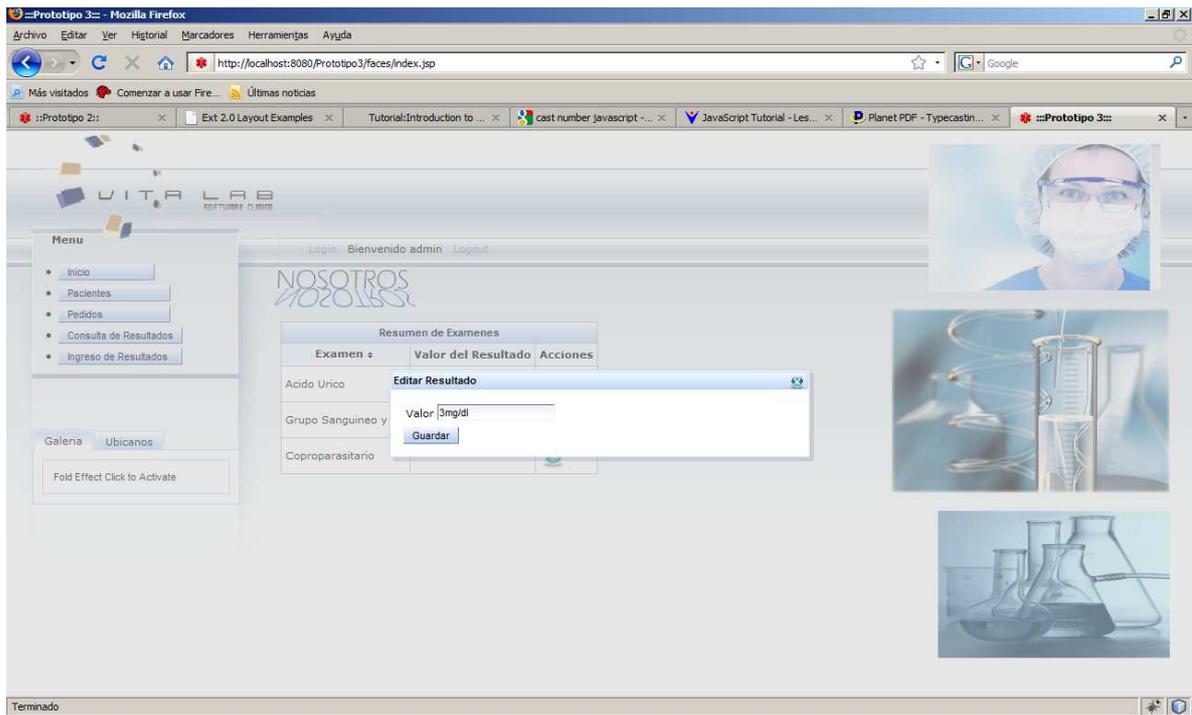


FIGURA 3.25 PANTALLA DE ASIGNAR VALORES A LOS EXAMENES SOLICITADOS

Fuente: El Autor

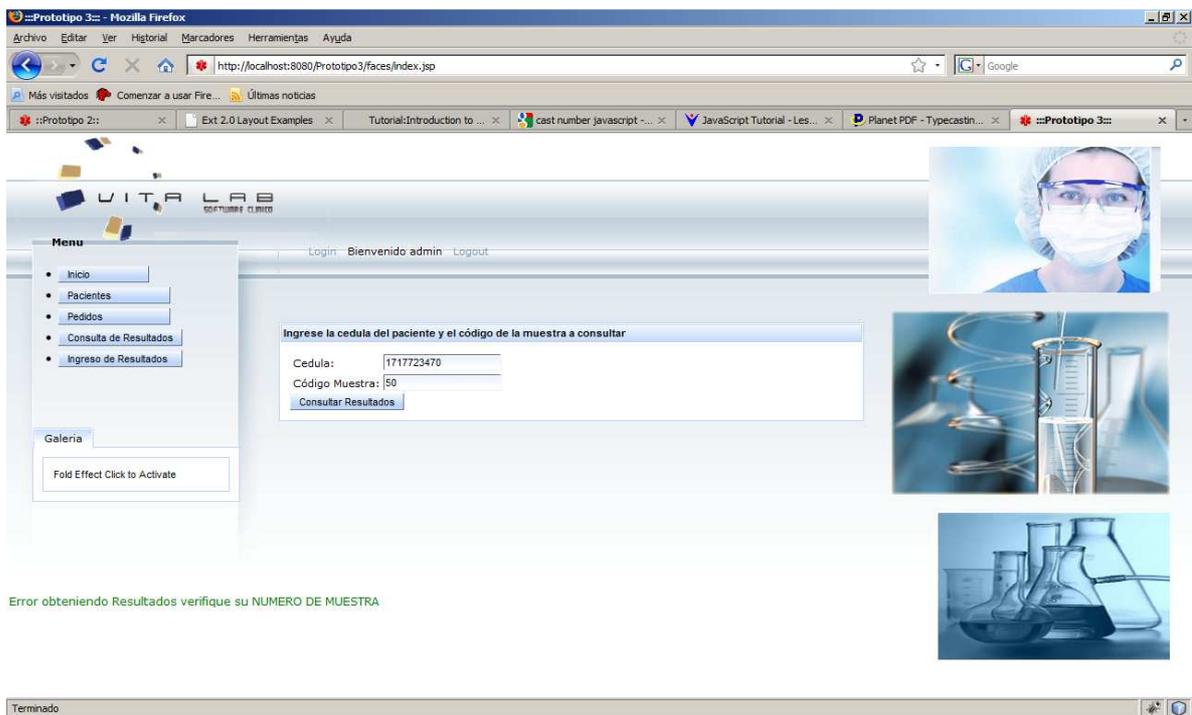
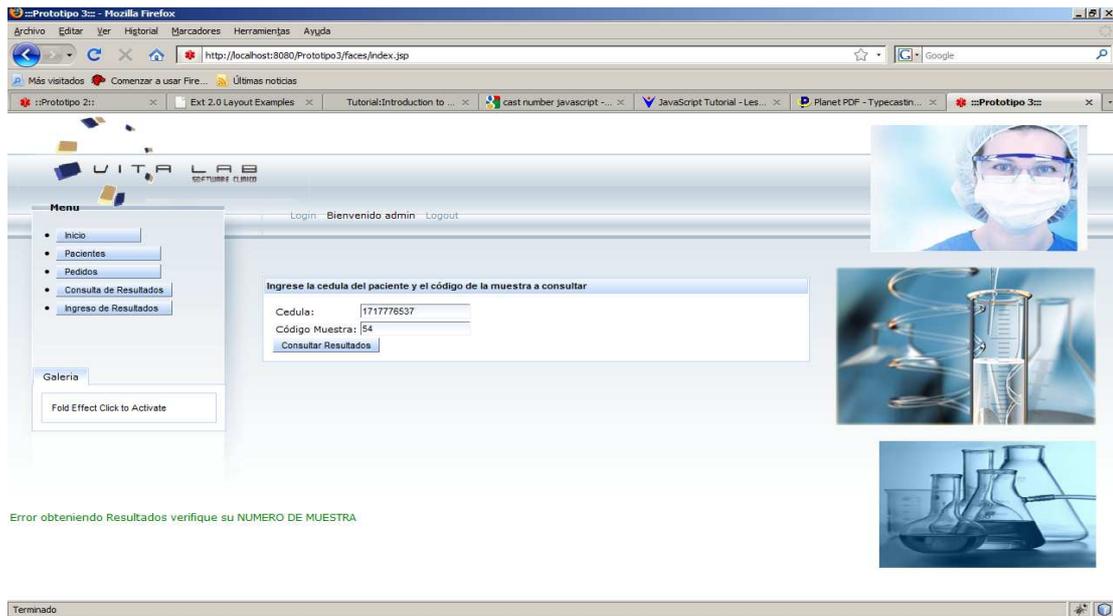


FIGURA 3.26 PANTALLA DEL FLUJO ALTERNO1 MUESTRA NO ESTÁ REGISTRADA

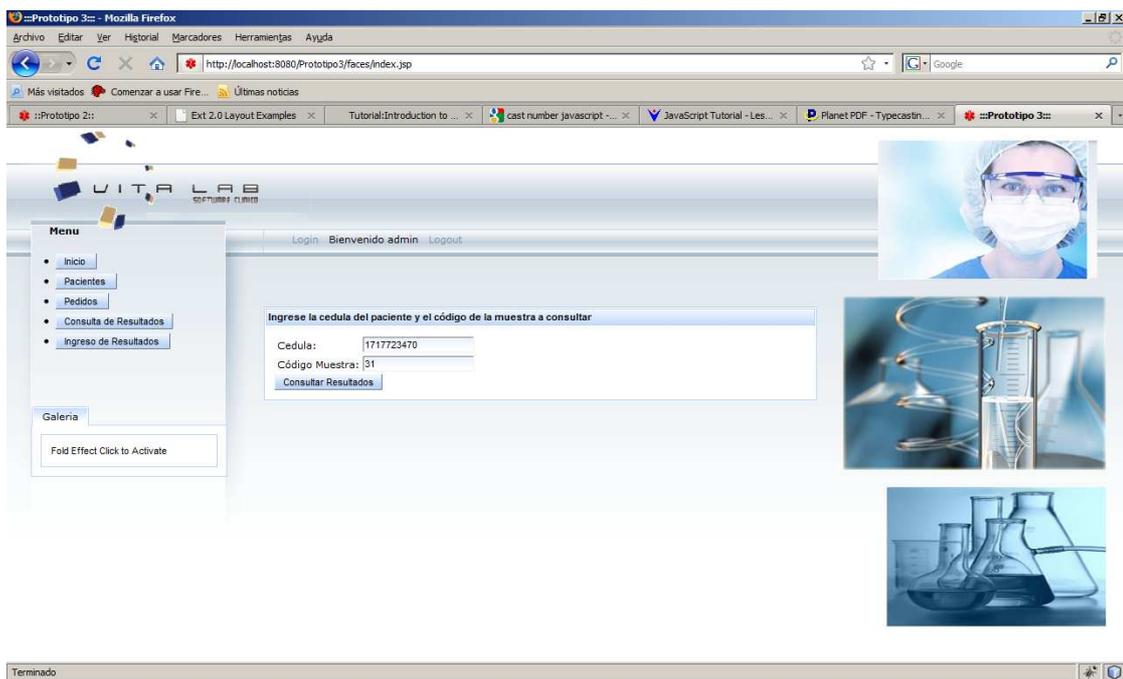
Fuente: El Autor



**FIGURA 3.27 PANTALLA DEL FLUJO ALTERNO2 PACIENTE NO ESTÁ REGISTRADO**

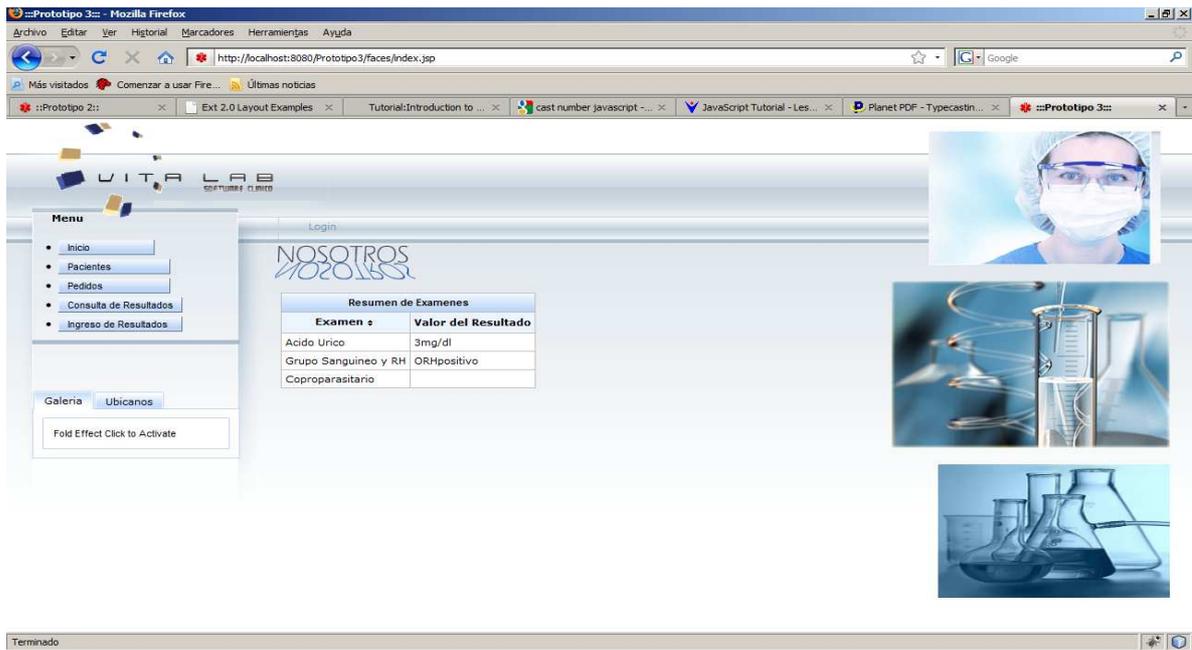
**Fuente:** El Autor

### 3.2.4 CONSULTAR RESULTADOS DE EXAMEN



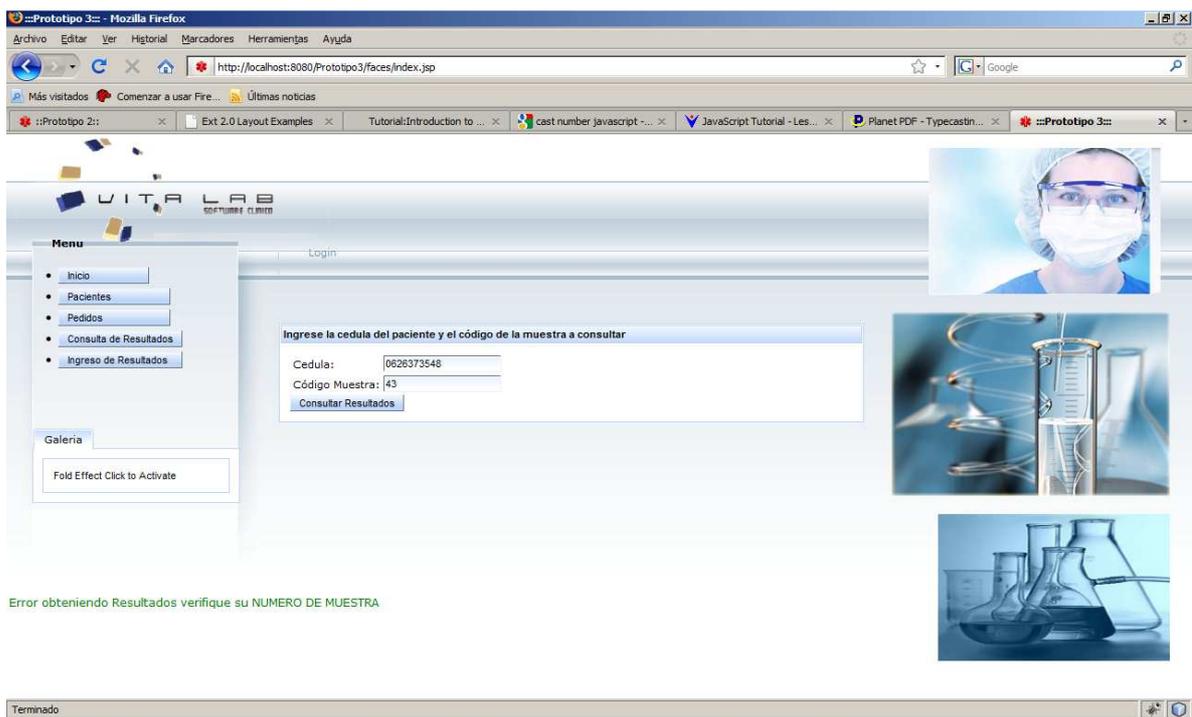
**FIGURA 3.28. PANTALLA DE CAPTURA DE DATOS DEL PACIENTE**

**Fuente:** El Autor



**FIGURA 3.29 PANTALLA DE DESPLIEGUE DE EXAMEN**

Fuente: El Autor



**FIGURA 3.30 PANTALLA DE MENSAJE "PACIENTE O MUESTRA INCORRECTA"**

Fuente: El Autor

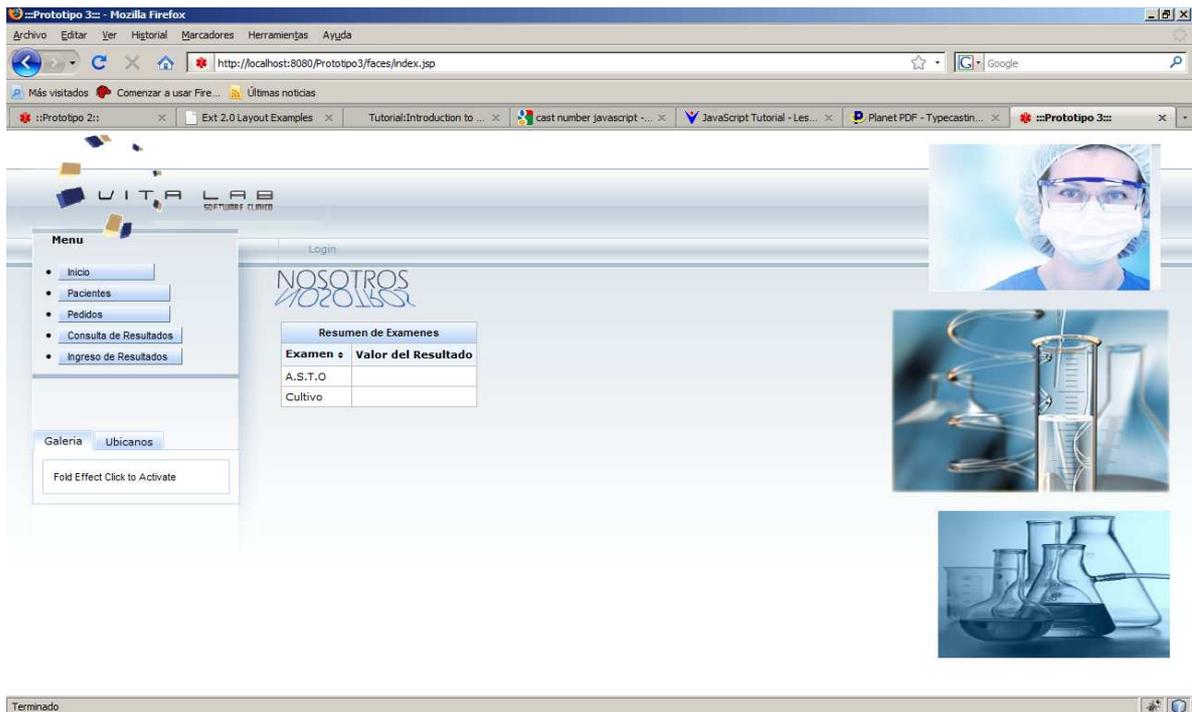


FIGURA 3.31 PANTALLA DE DESPLIEGUE DE LISTA DE RESULTADOS SIN VALORES

Fuente: El Autor

### 3.3 PROTOTIPO USANDO EXT JS

#### 3.3.1 REGISTRAR PEDIDO DE EXAMEN

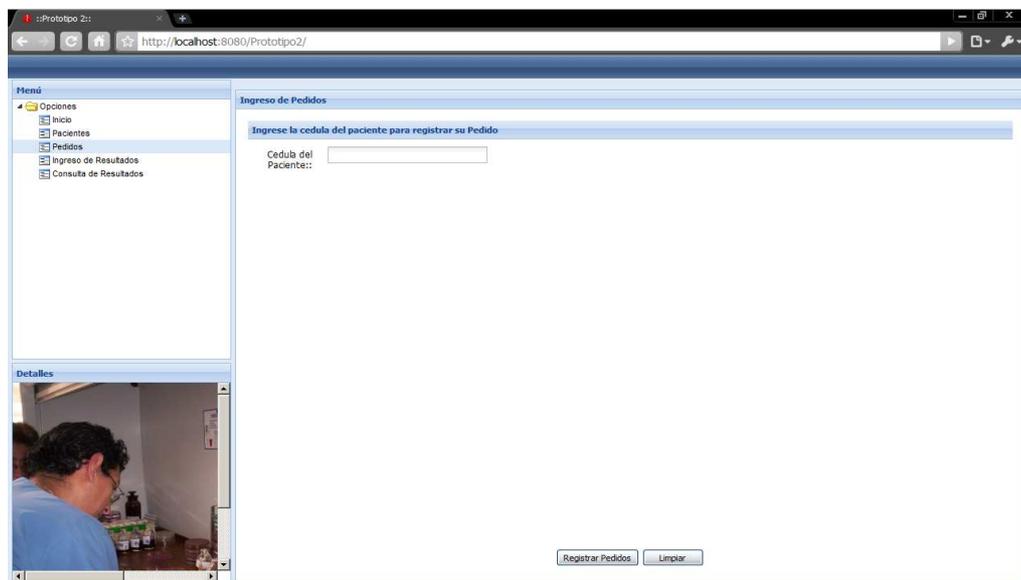


FIGURA 3.32 PANTALLA DE INGRESO DE CÉDULA DEL PACIENTE

Fuente: El Autor

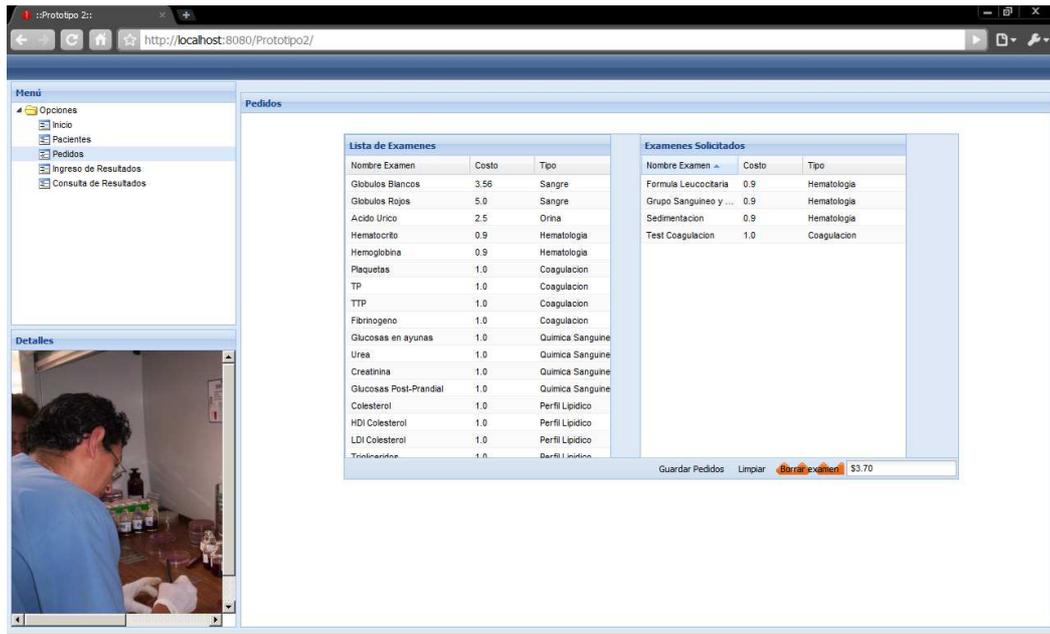


FIGURA 3.33 PANTALLA DE DESPLIEGUE DE LISTA DE EXÁMENES

Fuente: El Autor

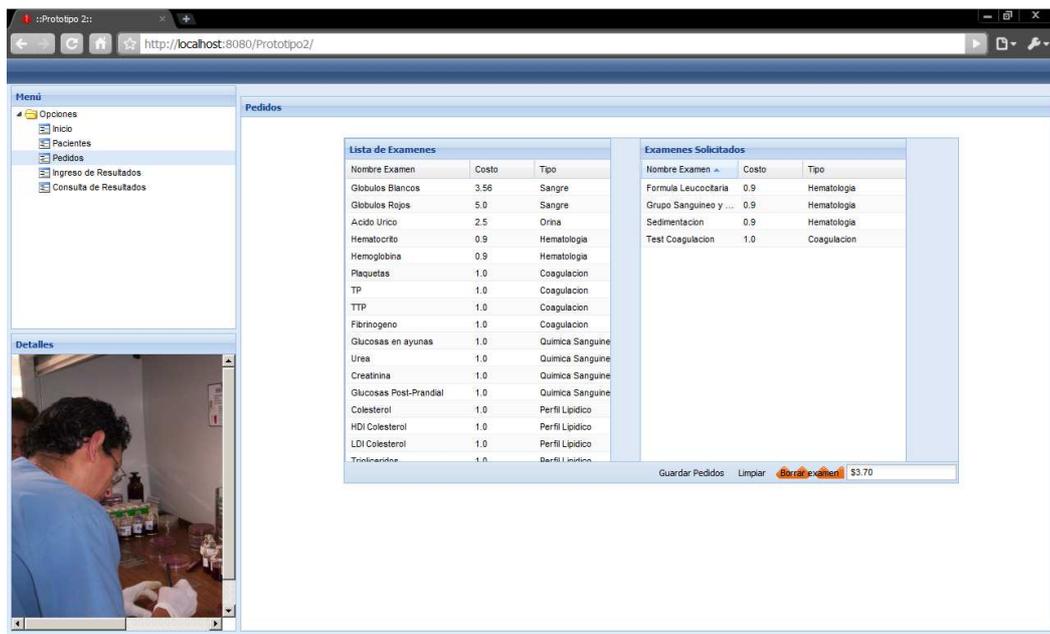
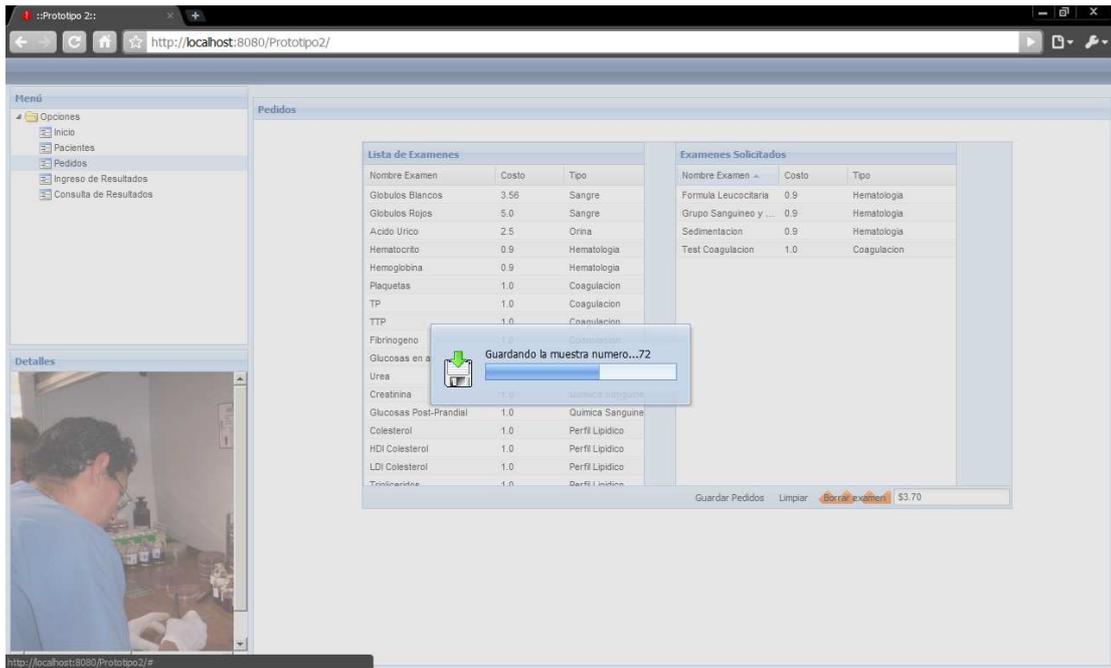


FIGURA 3.34 PANTALLA DE SELECCIÓN DE EXÁMENES SOLICITADOS

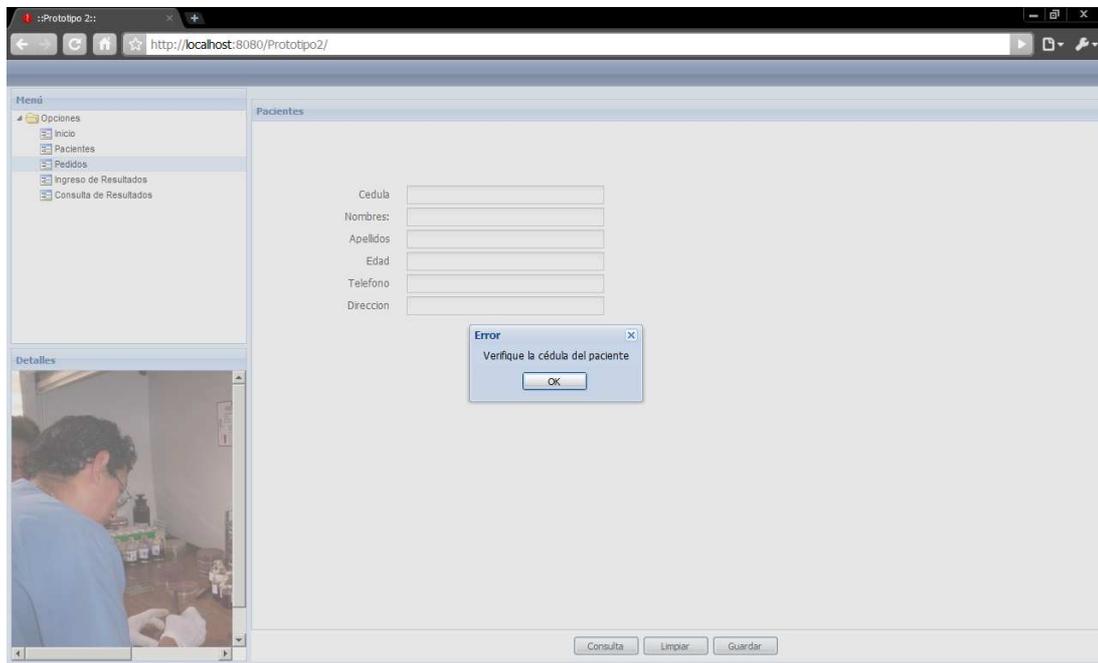
Fuente: El Autor



**FIGURA 3.35 PANTALLA DE PEDIDO HA SIDO GUARDADO**

**Fuente:** El Autor

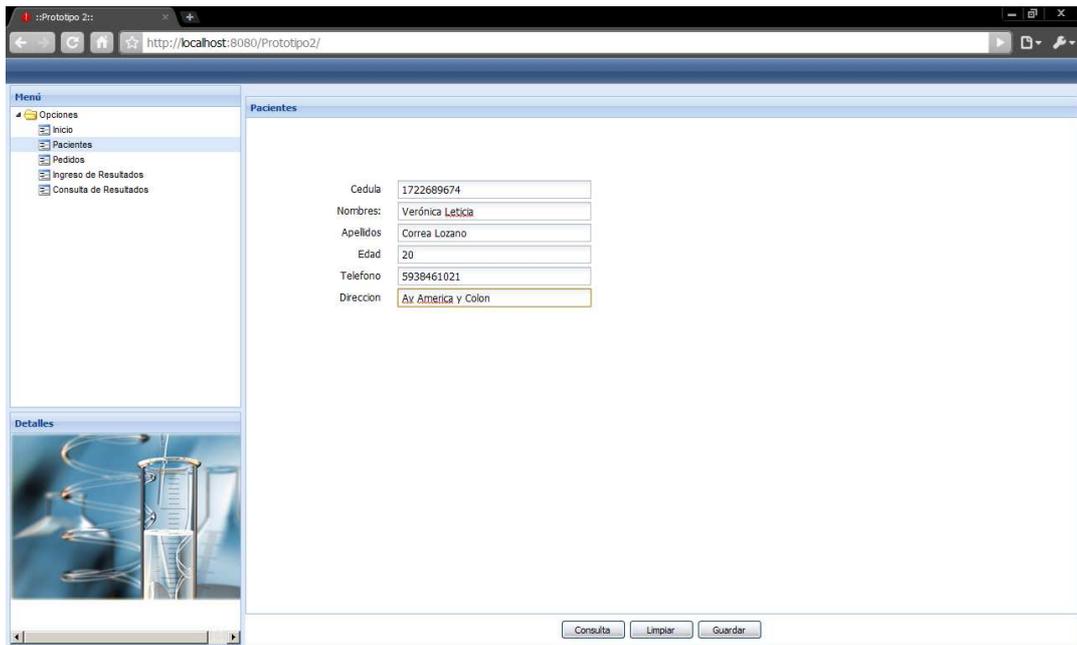
### Flujo de Excepción Paciente no Registrado



**FIGURA 3.36 PANTALLA DE EXCEPCIÓN PACIENTE NO HA SIDO REGISTRADO**

**Fuente:** El Autor

### 3.3.2 REGISTRAR PACIENTE



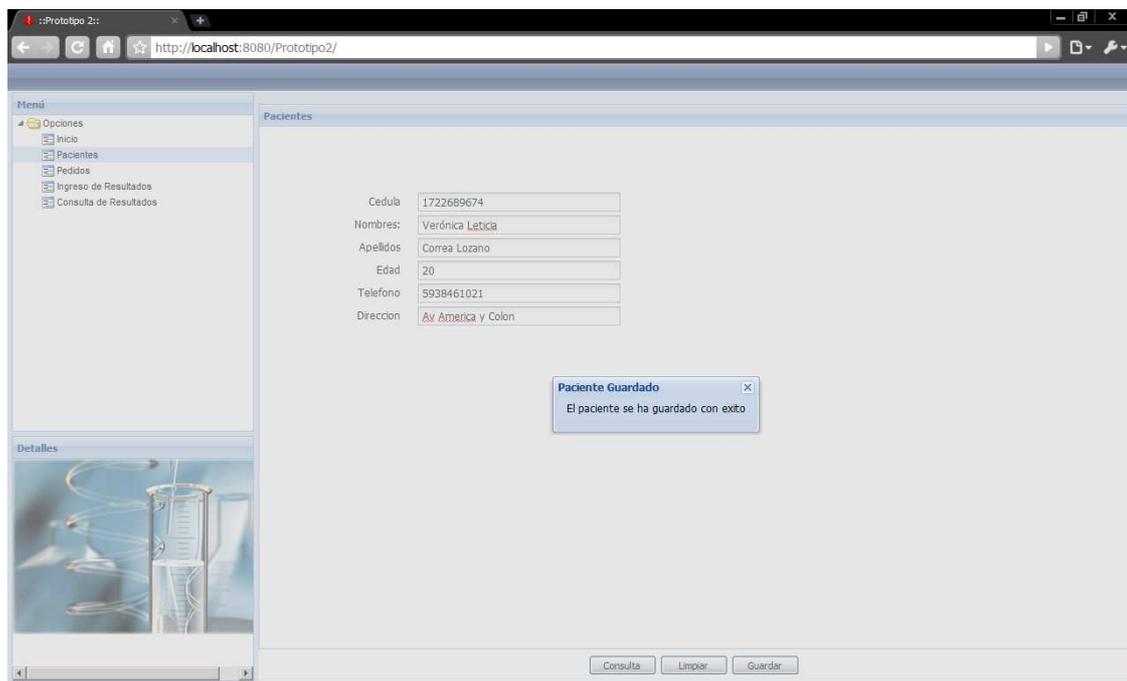
The screenshot shows a web browser window with the URL `http://localhost:8080/Prototipo2/`. The application interface includes a menu on the left with options: Inicio, Pacientes, Pedidos, Ingreso de Resultados, and Consulta de Resultados. The main area is titled "Pacientes" and contains a form with the following fields:

Cedula	1722689674
Nombres	Verónica Leticia
Apellidos	Correa Lozano
Edad	20
Telefono	5938461021
Direccion	Av America y Colon

At the bottom of the form, there are three buttons: "Consulta", "Limpiar", and "Guardar". A "Detalles" section at the bottom left shows a laboratory glass and test tubes.

FIGURA 3.37 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE

Fuente: El Autor



This screenshot shows the same patient registration form as Figure 3.37, but with a success message displayed in a dialog box. The message reads: "Paciente Guardado" and "El paciente se ha guardado con éxito". The form fields and buttons remain visible in the background.

FIGURA 3.38 PANTALLA DE PACIENTE GUARDADO

Fuente: El Autor

### 3.3.3 INGRESAR RESULTADOS DE EXAMEN

Menú

- Opciones
  - Inicio
  - Pacientes
  - Pedidos
  - Ingreso de Resultados
  - Consulta de Resultados

Detalles

funciona

Ingreso de Resultados

Ingrese la cédula y el número de muestra del paciente:

Cédula del Paciente:

Número de Muestras:

FIGURA 3.39 PANTALLA DE CAPTURA DE CÓDIGO DE MUESTRA

Fuente: El Autor

Menú

- Opciones
  - Inicio
  - Pacientes
  - Pedidos
  - Ingreso de Resultados
  - Consulta de Resultados

Detalles

funciona

Ingreso de Resultados

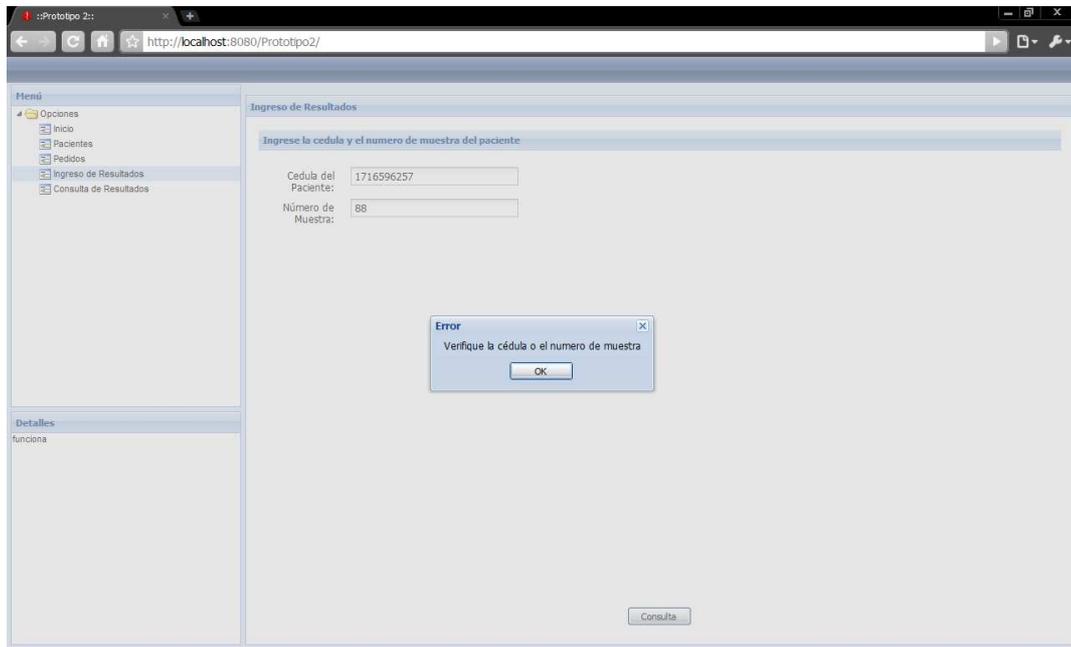
Edición de Resultados

Cód	Nombre Examen	Valor del Result...
84	Formúla Leucocitaria	123123
85	TGO	44444
86	TGP	5000
87	TP	Normal
88	Urea	Alta

FIGURA 3.40 PANTALLA DE LISTA DE EXÁMENES REGISTRADOS

Fuente: El Autor

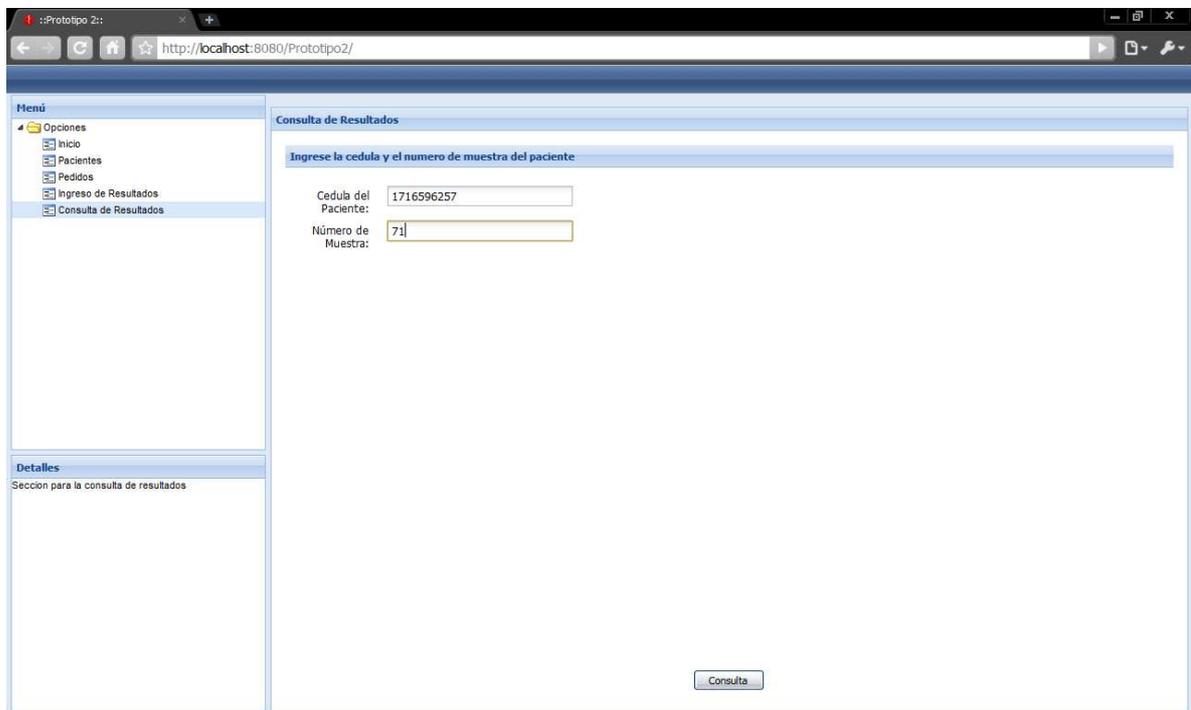
## Flujo Alterno muestra no registrada



**FIGURA 3.41 PANTALLA DEL FLUJO ALTERNO1 MUESTRA NO ESTÁ REGISTRADA**

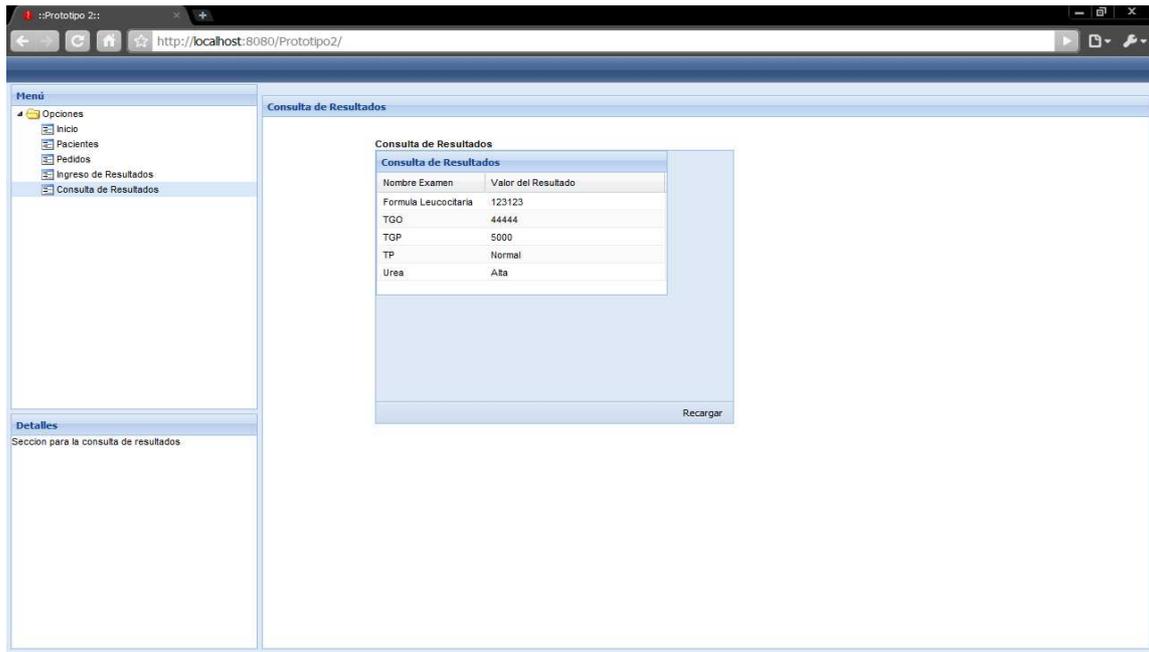
**Fuente:** El Autor

### 3.3.4 CONSULTAR RESULTADOS DE EXAMEN



**FIGURA 3.42 PANTALLA DE CAPTURA DE DATOS DEL PACIENTE Y NÚMERO DE MUESTRA**

**Fuente:** El Autor



**FIGURA 3.43 PANTALLA DE DELPLIEGUE DE RESULTADOS EXAMEN**

**Fuente:** El Autor

## CAPITULO 4. ANÁLISIS COMPARATIVO

### 4.1 EVALUACIÓN DE PROTOTIPOS

Para la evaluación de los prototipos desarrollados, se considerará que todos hacen uso de una única capa de backend desarrollada utilizando tecnología Java, la cual está expuesta como web services.

En este capítulo se dará una valoración definitiva a los criterios que no fueron valorados en el capítulo 2 y a su vez se dará un valor definitivo para los criterios cuya calificación fue preliminar.

#### 4.1.1 AMBIENTE PARA EL PROCESO DE DESARROLLO Y EVALUACIÓN

Durante el proceso de desarrollo de los prototipos se utilizó una máquina con las siguientes características

Sistema Operativo:	Microsoft Windows Vista
Procesador:	Intel Pentium Dual core 1,6 GHz
Memoria RAM:	3GB
Versión de JDK:	1.6 update 12.
Versión de flash player:	Flash Player 10.
IDE para Rich Faces	Netbeans 6.5
IDE para Adobe Flex	Flex Builder 3
IDE para ExtJs	Aptana Studio, Netbeans
Servidores web Flex	Apache Tomcat, JBoss Application Server 4.2.
Servidor de Aplicaciones	Glassfish

**TABLA 23. CARACTERÍSTICAS DE LA MÁQUINA DE DESARROLLO**

**Elaborado por:** El autor

Para la evaluación del desempeño de los prototipos adicionales a la máquina de desarrollo se utilizó una máquina de pruebas con las siguientes características:

Sistema Operativo:	Open Suse 10.3
Procesador:	Intel Pentium Dual core 1,6 GHz
Memoria RAM:	1.5 GB
Versión de JDK:	1.6.

**TABLA 24. CARACTERÍSTICAS DE LA MÁQUINA GNU/LINUX**

**Elaborado por:** El autor

#### **4.1.2 HERRAMIENTAS PARA LA EVALUACIÓN**

Con el fin de simular un ambiente de producción adecuado en el cual poner a prueba el rendimiento de los prototipos, en los aspectos referentes al consumo de canal de red y tiempo de inicialización se escogieron las herramientas que se describen a continuación.

##### **4.1.2.1 WAPT**

Se trata de una herramienta desarrollada por SoftLogica que permite realizar pruebas de carga, estrés y rendimiento sobre aplicaciones web, entre sus características encontramos:

- Costo de Licencia: entre \$350 a \$750 dólares.
- Capacidad de simular un número elevado de clientes conectándose simultáneamente o a intervalos a la aplicación.
- Capacidad de restringir el ancho de banda para simular conexiones ADSL (Asymmetric Digital Subscriber Line), o conexiones de mayor ancho de banda.
- Test basados en grabaciones del comportamiento de un usuario modelo sobre un navegador incorporado.
- Reportes gráficos y en formato HTML de los parámetros medidos.
- No funciona de manera adecuada con aplicaciones con contenido Flash, razón por la cual no se pudo utilizar para la valoración del prototipo construido con Flex.

##### **4.1.2.2 Net Limiter**

Se trata de una herramienta de monitoreo de red y administración del ancho de banda perteneciente a Locktime Software. Entre sus características tenemos:

- Costo de licencia: entre 17,34€ a 34,74€ euros.
- Firewall incorporado.
- Administración remota de equipos.

- Administración de ancho de banda para cada máquina de la red.
- Control de conexiones entrantes y salientes.
- Creación de filtros y reglas de acceso.
- Estadísticas de consumo de ancho de banda.

#### 4.1.2.3 Performance Monitor

Herramienta de monitoreo y auditoría integrada en el sistema operativo Windows y que basado en mediciones de contadores asignados a cada uno de los componentes del sistema permite determinar el estado del funcionamiento a nivel de servicios y hardware.

#### 4.1.3 AMBIENTE PARA LA EVALUACIÓN

A continuación se describe el diagrama de red utilizado para la evaluación de tiempo de inicialización y consumo de canal de red.



FIGURA 4.1 DIAGRAMA DE RED EVALUACIÓN

Fuente: El Autor

Se describe a continuación las máquinas que se utilizaron para la evaluación

Nombre de la Máquina	Sony-PC
Sistema Operativo:	Microsoft Windows Vista
Procesador:	Intel Pentium Dual core 1,6 GHz
Memoria RAM:	3GB
Velocidad tarjeta de Red	10/100 Mbps (Mega bits por segundo)

TABLA 25. CARACTERÍSTICAS DE LA MÁQUINA SERVIDOR

Elaborado por: El autor

Nombre de la Máquina	Casa
Sistema Operativo:	Microsoft Windows XP
Procesador:	Intel Pentium Dual Core 3,4 GHz
Memoria RAM:	1GB
Velocidad tarjeta de Red	10/100 Mbps

**TABLA 26. CARACTERÍSTICAS DE LA MÁQUINA CLIENTE**

**Elaborado por:** El autor

#### **4.1.3.1 Configuración de los Escenarios de pruebas con las herramientas**

##### **WAPT**

Las pruebas utilizando WAPT en la máquina cliente se configuraron con las siguientes características:

- Tiempo de medición 3 minutos.
- 20 usuarios con intervalos de un usuario cada 4 segundos.
- Velocidad de conexión simulada 256 Kbps. (Kilo bits por segundo).

##### **NetLimiter**

NetLimiter se utilizó de forma exclusiva para las mediciones del prototipo desarrollado con Adobe Flex, en este se limitó la velocidad de descarga a 32 KBps (KiloBytes por segundo) lo cual equivale a una conexión de 256 Kbps para la descarga.

La descripción de la configuración de las herramientas se encuentra descrita en el Anexo2.

##### **Performance Monitor**

Se programó mediciones para los contadores correspondientes a la interfaz de red, durante un lapso de 5 minutos, los cuales se almacenaron en un archivo de texto.

De manera particular para las mediciones en Flex, se vario el número de clientes entre uno y cuatro accediendo de forma simultánea.

La descripción de la configuración de las mediciones con Performance Monitor se encuentra descrita en el Anexo2.

## 4.1.4 EVALUACIÓN DEL PROTOTIPO DESARROLLADO CON ADOBE FLEX

### 4.1.4.1 Apariencia en los diferentes navegadores

Los componentes del prototipo desarrollado con Adobe Flex conservan las mismas características y dimensiones en los distintos navegadores considerados, el posicionamiento puede verse afectado por la cantidad de barras de herramientas del navegador y el espacio que estas ocupan, sin embargo si consideramos la posición en partir del espacio destinado para mostrar las páginas dentro del navegador, esta se mantiene.

Cabe señalar que si el navegador no cuenta con los complementos o plugins necesarios para la visualización de la aplicación, esta informa de este requerimiento y ofrece la opción de descargar e instalar el componente de Flash Player más adecuado para el navegador.

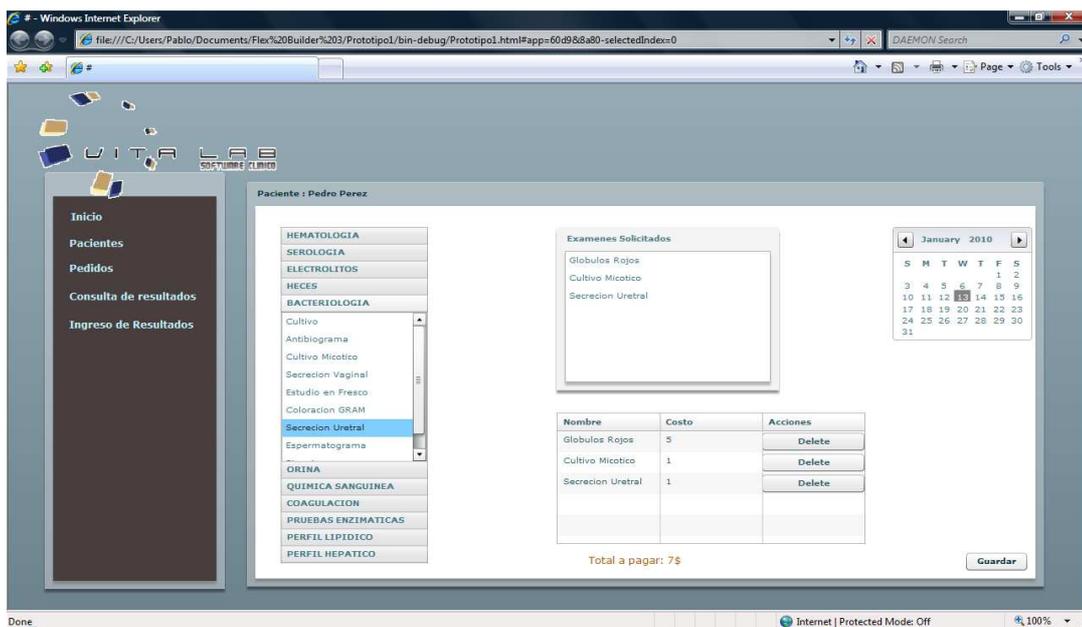


FIGURA 4.2 PROTOTIPO DE ADOBE FLEX EN INTERNET EXPLORER

Fuente: El Autor

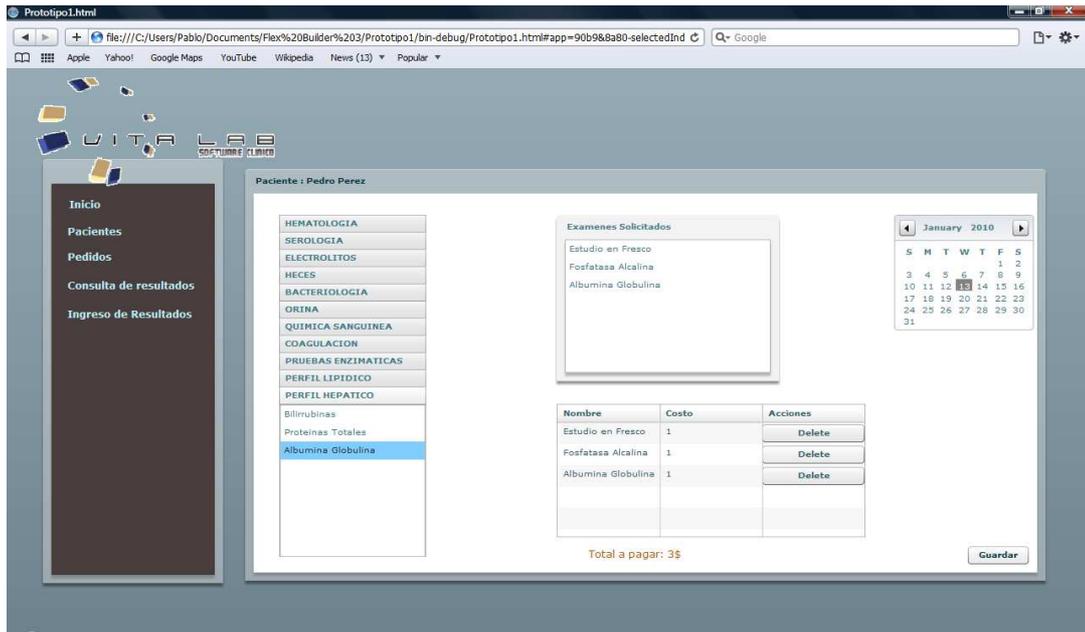


FIGURA 4.3 PROTOTIPO DE ADOBE FLEX EN SAFARI

Fuente: El Autor

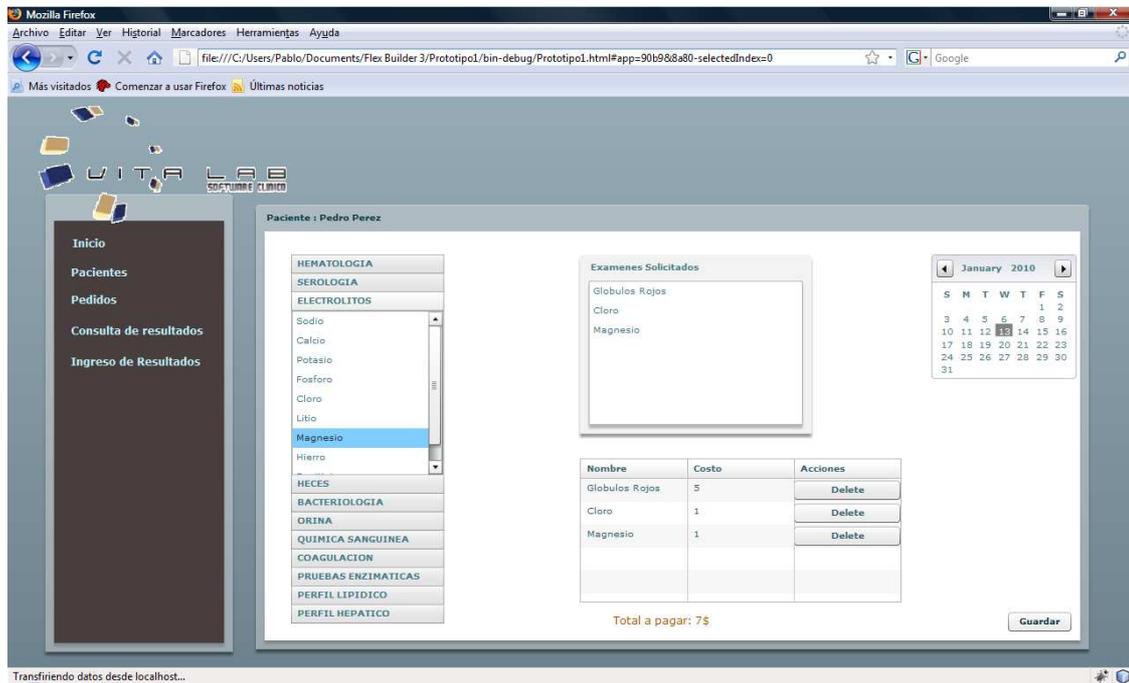


FIGURA 4.4 PROTOTIPO DE ADOBE FLEX EN MOZILLA FIREFOX

Fuente: El Autor

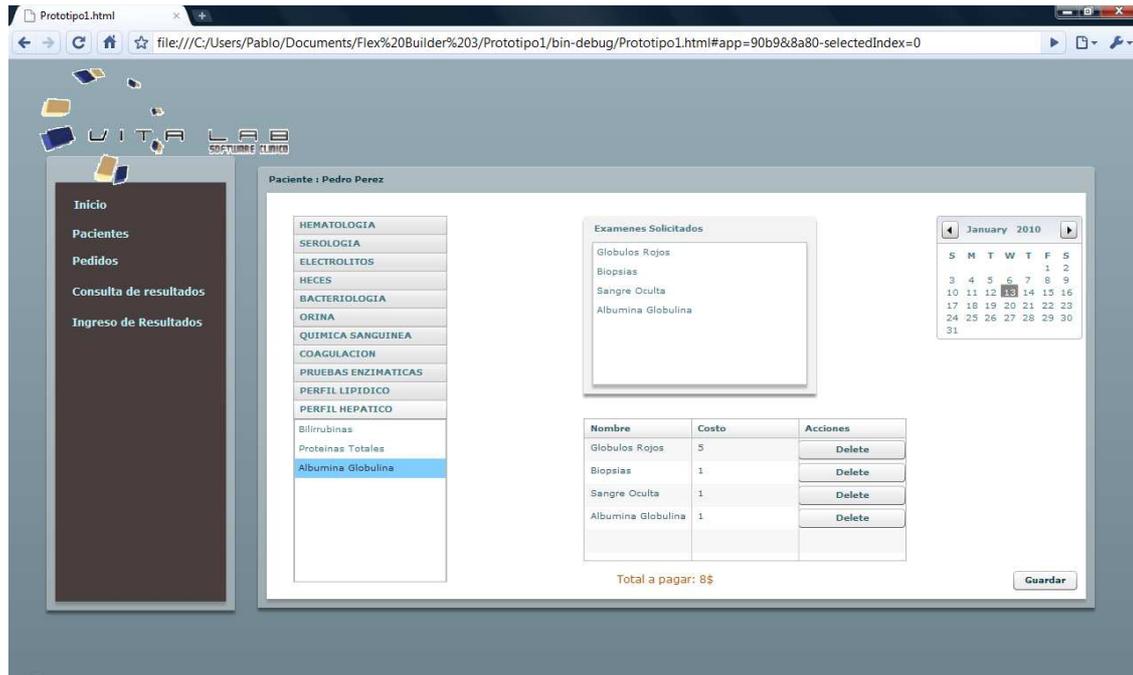


FIGURA 4.5 PROTOTIPO DE ADOBE FLEX EN GOOGLE CHROME

Fuente: El Autor

## Valoración final 4

### 4.1.4.2 Independencia de la resolución

El aspecto más importante y que permite conservar la apariencia de las aplicaciones, es la aplicación de constraints o anclas para mantener la relación de distancia entre controles, al igual que la distancia respecto a los bordes de los contenedores; esta característica permite que pese al cambio de resolución y el aumento o disminución de área de despliegue disponible, se conserve la relación de distancia, permitiendo a los controles crecer o cambiar de posición si es necesario.

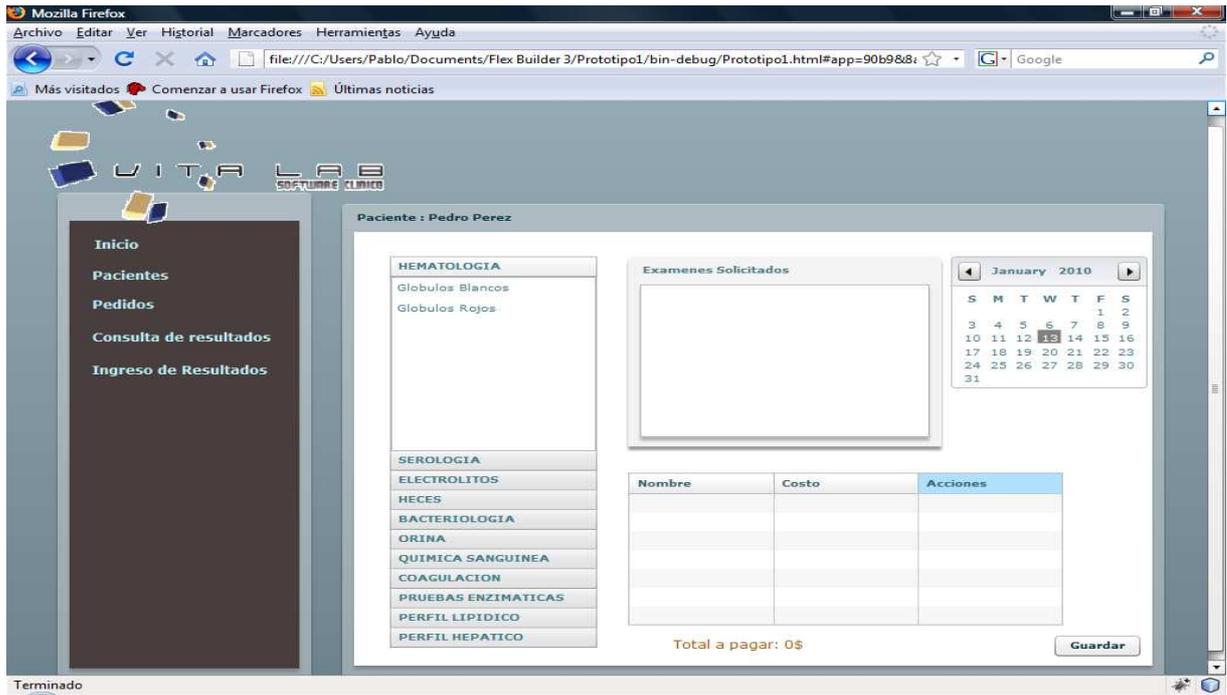


FIGURA 4.6 PROTOTIPO DE ADOBE FLEX A RESOLUCIÓN DE 1024X 768 PÍXELES

Fuente: El Autor

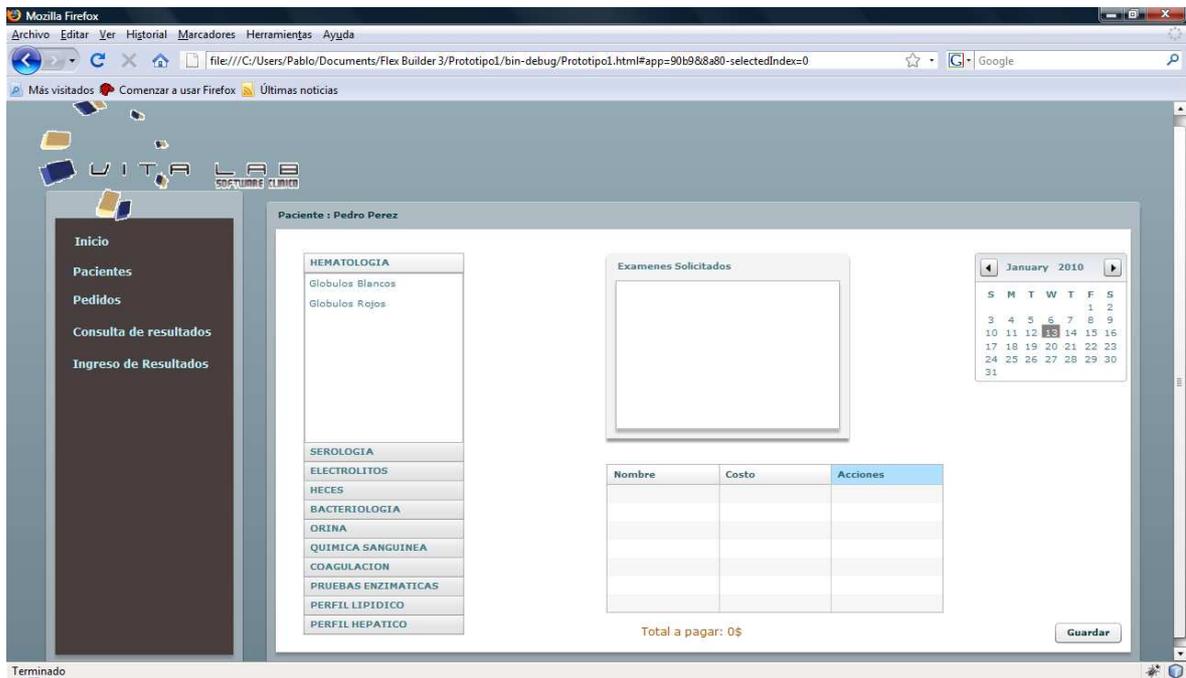


FIGURA 4.7 PROTOTIPO DE ADOBE FLEX A RESOLUCIÓN DE 1280 X 768 PÍXELES

Fuente: El Autor

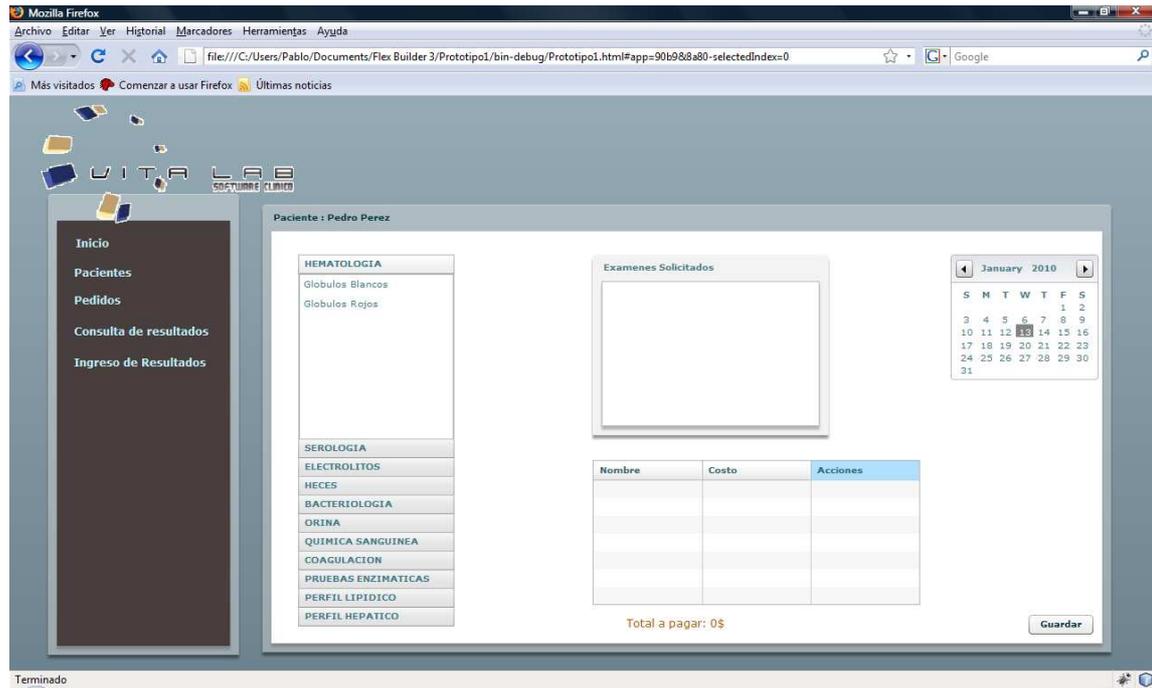


FIGURA 4.8 PROTOTIPO DE ADOBE FLEX A RESOLUCIÓN DE 1280 X 800 PÍXELES

Fuente: El Autor

## Valoración final: 2.

### 4.1.4.3 Tiempo de descarga o tiempo de inicialización

Una vez realizadas las mediciones se obtuvo un tiempo promedio de descarga de para la página principal de la aplicación de 10,119 segundos.

Los reportes de las mediciones obtenidas se presentan en el Anexo3.

## Valoración final: 2

### 4.1.4.4 Consumo de canal de red

A partir de los resultados obtenidos con performance monitor, se determino que el promedio de consumo de ancho de banda por solicitud de usuario es de 10,11 KBps.

Los reportes de las mediciones obtenidas se presentan en el Anexo3.

## Valoración final: 3

#### 4.1.4.5 Requerimientos de Hardware y Software para el desarrollo

La instalación y uso del framework no requirió de recursos adicionales a los descritos en la sección 4.1.1, los cuales encajan y en algunos casos superan los requerimientos mínimos que se mencionan en la sección 1.2.1.5.

El único requisito adicional y que no se menciona en la sección 1.2.1.5 es la necesidad de la versión “Debug” de Flash Player con el fin de realizar un seguimiento paso a paso de la ejecución del código; esto sin embargo no representa un requerimiento extraordinario, ya que Adobe permite la descarga de esta versión del plugin, la cual además es de simple instalación.

Además para comprobar la independencia de plataforma, se instaló las versiones de Flex Builder disponibles para GNU/Linux y Windows.

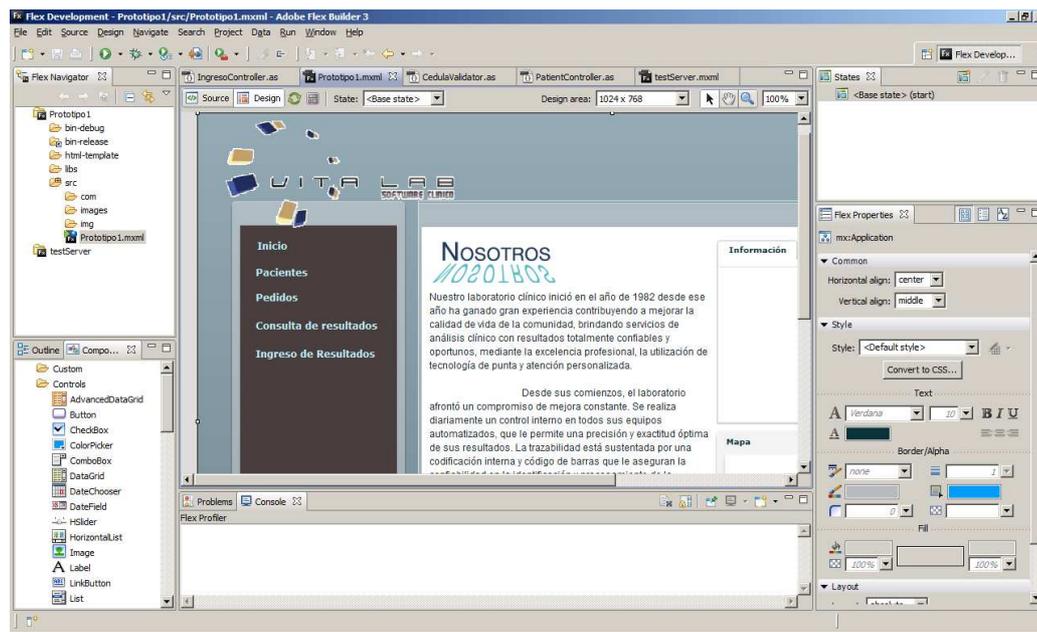


FIGURA 4.9 FLEX BUILDER INSTALADO EN UN AMBIENTE WINDOWS

Fuente: El Autor

Cabe destacar que las características de Flex Builder en un ambiente GNU/Linux difieren del que se tiene en un ambiente Windows, de manera especial en la creación de interfaces; la cual se hace de manera visual únicamente en Windows y en GNU/Linux se limita a la creación vía código.



FIGURA 4.10 INSTALACIÓN DE FLEX BUILDER EN UN AMBIENTE GNU/LINUX

Fuente: El Autor

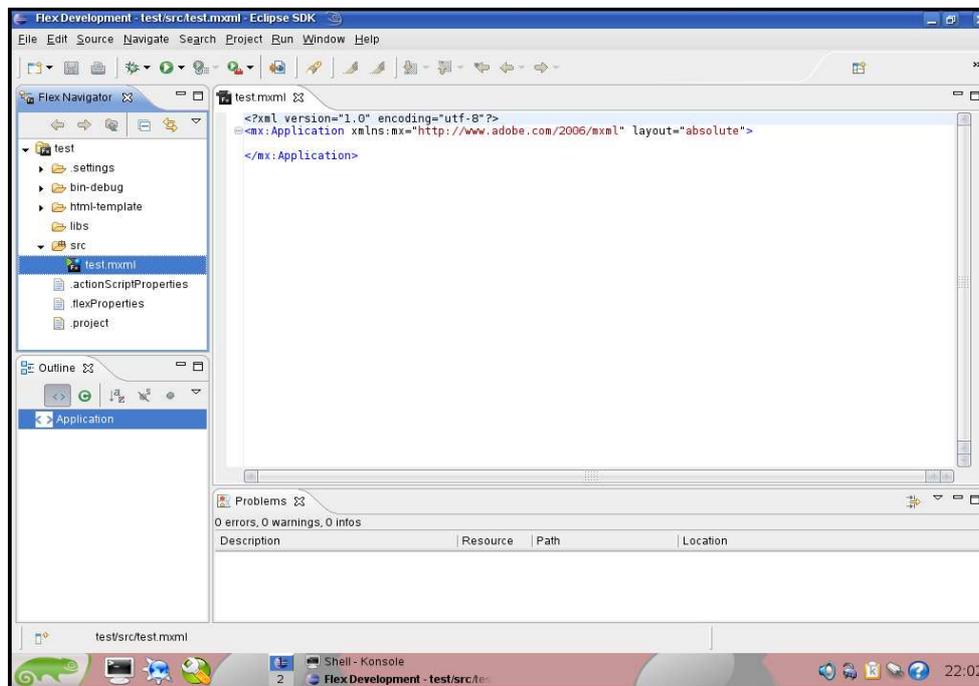


FIGURA 4.11 FLEX BUILDER INSTALADO EN UN AMBIENTE GNU/LINUX

Fuente: El Autor

Valoración final: 3

#### 4.1.4.6 Patrones de Diseño

Una vez analizada la microarquitectura Cairngorm e intentado implementarla, se pudo notar que se trataba de una arquitectura compleja, si se toma en cuenta la necesidad de la creación de un componente de escucha y difusión de eventos entre Vista y Controlador; por esta complejidad se adoptó el patrón de Diseño

MVC pero sin Cairngorm de por medio. Se logro realizar una implementación exitosa de este patrón valiéndose de la experiencia personal. Es así que se realizo la aplicación de la siguiente manera:

Vista: Prototipo1.MXML

PatientManager.as

Muestra.as

Exam.as

Session.as

User.as

CedulaValidator.as

Resultado.as

Controlador:

IndexController.as

IngresoController.as

PatientControler.as

PatientManger.as

PedidoCotroller.as

Modelo

Backend expuesto a través de webServices

Se comprobó la capacidad para implementar un patrón de diseño aceptado y con amplia documentación, a pesar de no mencionarse de manera formal por parte de Adobe.

La valoración se disminuyó debido a la complejidad para implementar y verificar el funcionamiento del patrón ad hoc propuesto por la comunidad.

**Valoración final: 2**

#### **4.1.4.7 Curva de aprendizaje**

La valoración final a este criterio se dará en base a la experiencia personal durante el desarrollo del prototipo. Se debe considerar que se trata de un usuario con alrededor de 3 años de experiencia en el desarrollo de aplicaciones web, con conocimientos medios de lenguajes como C# y Java, con un nivel bajo de conocimiento de lenguajes como ActionScript y JavaScript.

Con estos antecedentes, se tiene que la experiencia con Flex Builder resultó satisfactoria y en un tiempo aproximado de 3 semanas con 6 horas de trabajo por día se logro un nivel medio de conocimiento de componentes y uso del lenguaje. Gracias al apoyo de los recursos disponibles para el aprendizaje se logró desarrollar una aplicación haciendo uso de varios componentes que demuestran la riqueza gráfica del framework.

**Valoración final: 3**

#### **4.1.4.8 Herramientas de desarrollo**

La experiencia de trabajo con Flex Builder da como resultado un balance positivo, el diseño de interfaces en un entorno visual acelera considerablemente el tiempo de desarrollo, permite además la creación de componentes personalizados de una manera sencilla, así como el trabajo colaborativo permitiendo la interacción con servidores de control de versiones.

Flex Builder a nivel de análisis sintáctico y semántico de código cumple de manera satisfactoria su cometido, detectando errores y sugiriendo posibles correcciones.

Además de manera intuitiva y mediante asistentes gráficos permite el despliegue de la aplicación en varios servidores web y de aplicaciones.

Pese a que Flex Builder no permite la realización automática de pruebas unitarias, las herramientas de debug adicionales como Flash Player (versión Debug) integrada con la perspectiva de Debug del IDE permiten una realizar pruebas de cada uno de los proyectos ya sean librerías, componentes personalizados o aplicaciones con varias interfaces.

**Valoración final: 4**

#### **4.1.4.9 Facilidad de interacción con otras tecnologías**

Luego del trabajo se comprobó que Adobe Flex es un framework orientado a la presentación, el cual sin embargo proporciona grandes facilidades para interactuar con diferentes tecnologías mediante el concepto de widgets. Si se trata de interactuar con componentes de negocio basado en los conceptos de orientación a servicios Flex ofrece varias opciones, la manera más sencilla es mapear el servicio mediante un tag propio de MXML, en el cual se definen las

operaciones así como que función será la encargada de manejar los resultados y cual la encargada de los posibles fallos.

De esta forma se puede acceder de manera directa a los objetos o resultados devueltos por la invocación de los servicios y mapearlos a Value Objects o directamente a componentes como listas o grids.

**Valoración final: 4**

#### **4.1.4.10 Facilidad de puesta en producción**

Una aplicación desarrollada en Flex y que no requiere de interacción con web services u otro tipo de recursos, puede ser puesta en producción en cualquier servidor web o servidor de aplicaciones con soporte para aplicaciones web, en el caso del prototipo desarrollado, el cual requiere de acceso a web services, se pudo poner en producción a la aplicación en servidores como Apache, Apache Tomcat y JBoss Application Server, para lo cual basta con copiar la carpeta que contiene la solución compilada para distribución en el directorio de aplicaciones web y acceder a la página HTML con el nombre del proyecto

**Valoración final: 2**

### **4.1.5 EVALUACIÓN DEL PROTOTIPO DESARROLLADO CON RICH FACES**

#### **4.1.5.1 Apariencia en los diferentes navegadores**

Mediante el acceso al prototipo desarrollado usando Rich Faces desde varios navegadores, se observa que los componentes conservan el diseño, tamaño y relación entre ellos, debido al espacio disponible para la presentación de la aplicación, se notan diferencias mínimas de posición, sin embargo si se considera solo el área de despliegue no se notan diferencias.

En el caso particular del prototipo desarrollado, este muestra consistencia en 3 de los cuatro navegadores considerados en la evaluación, mas la apariencia se ve afectada de manera significativa en Internet Explorer v7 no en los componentes propios de Rich Faces sino en elementos adicionales como imágenes de background definidas en hojas de estilos

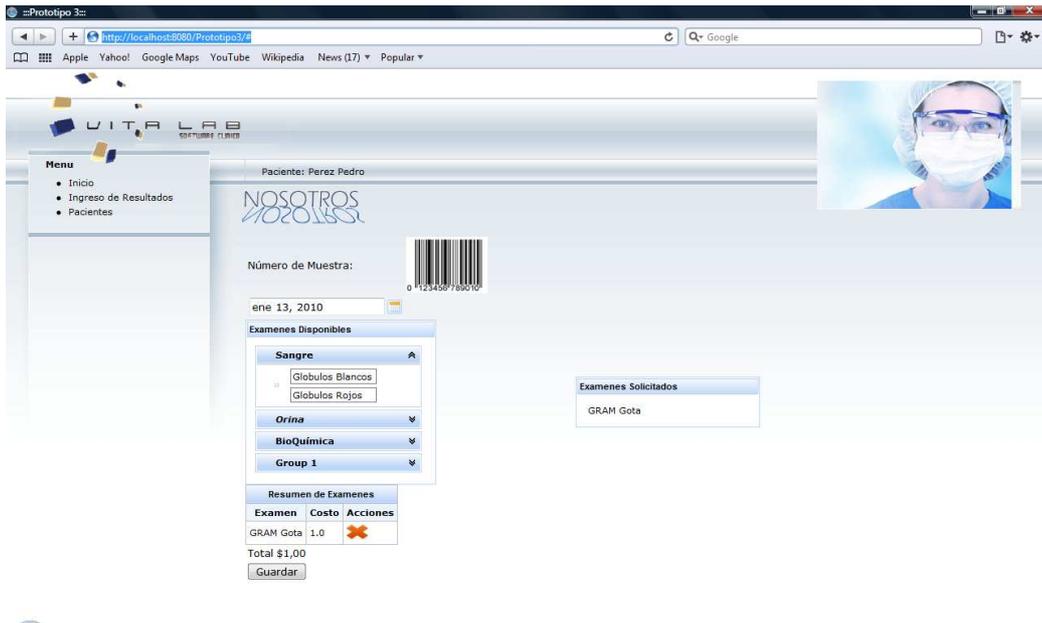


FIGURA 4.12 PROTOTIPO DE RICH FACES EN SAFARI

Fuente: El Autor

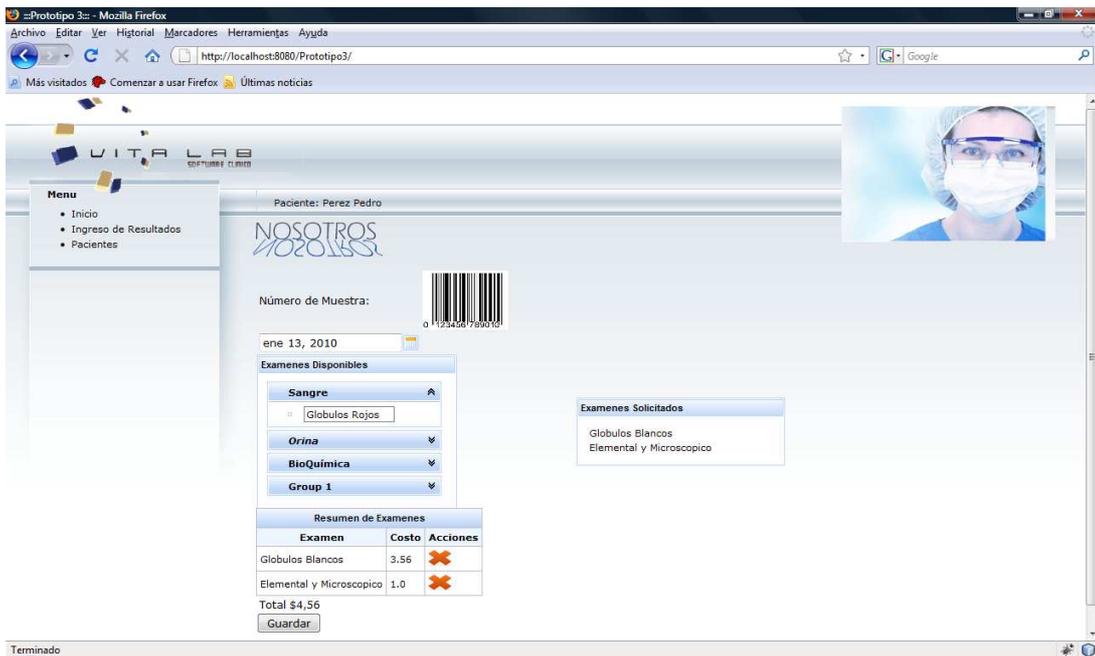


FIGURA 4.13 PROTOTIPO DE RICH FACES EN MOZILLA FIREFOX

Fuente: El Autor

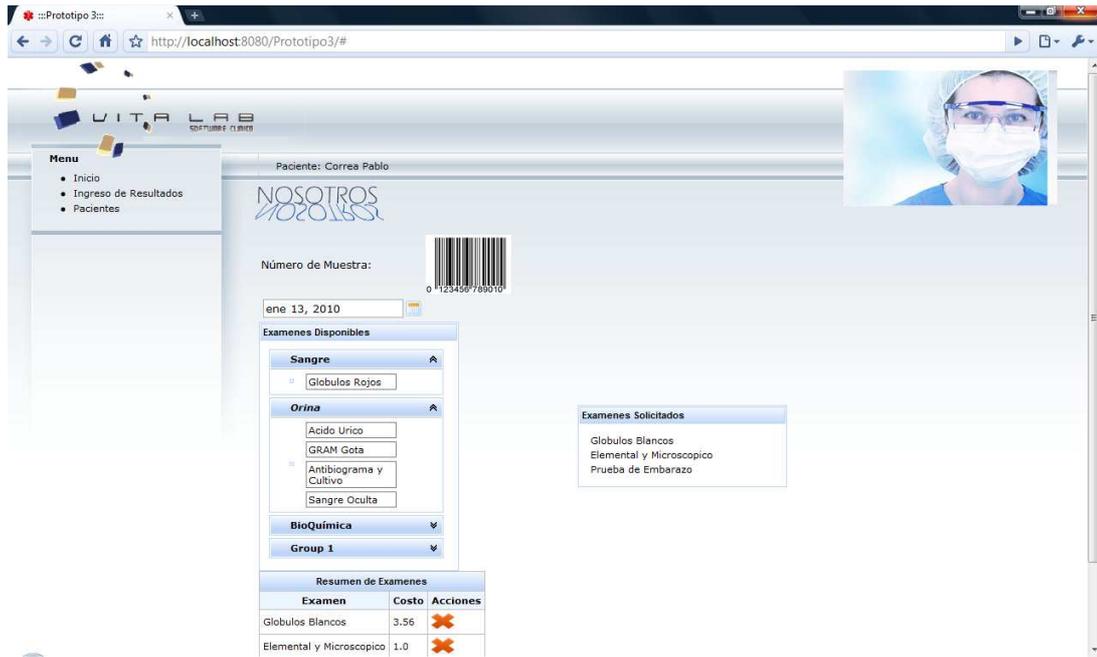


FIGURA 4.14 PROTOTIPO DE RICH FACES EN GOOGLE CHROME

Fuente: El Autor

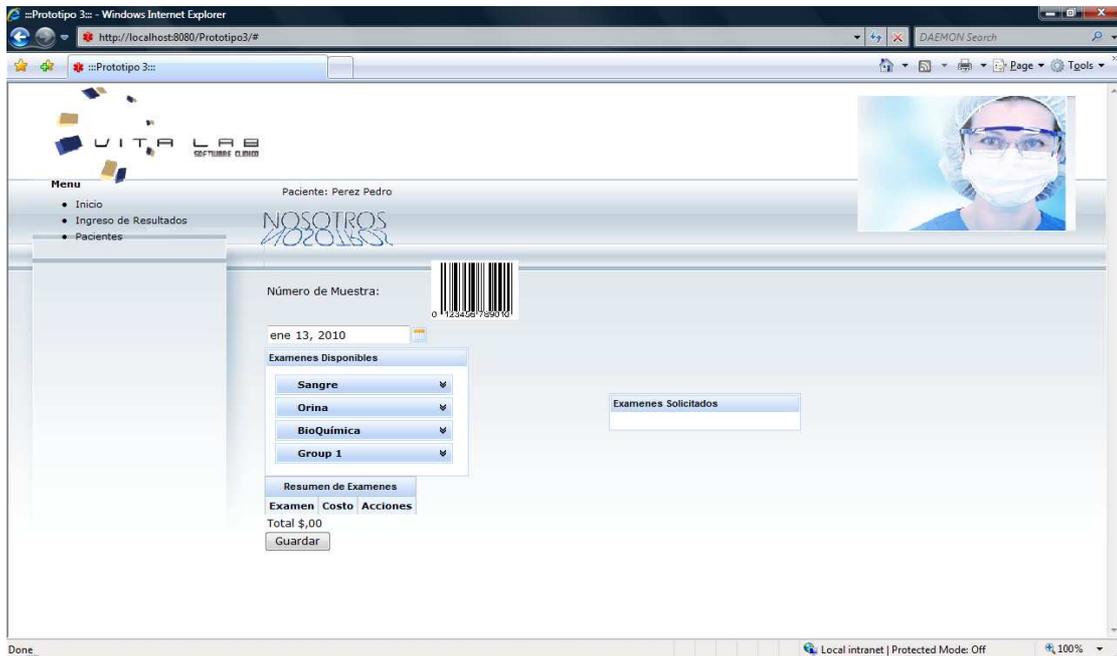


FIGURA 4.15 PROTOTIPO DE RICH FACES EN INTERNET EXPLORER V7

Fuente: El Autor

Valoración final: 4

#### 4.1.5.2 Independencia de la resolución

De las pruebas realizadas se pudo notar que los componentes propios de Rich Faces conservan su aspecto, para componentes como imágenes y componentes HTML en los cuales se ha definido tamaño en pixeles, este se mantiene constante razón por la cual distorsionan su imagen

**Valoración final: 2**

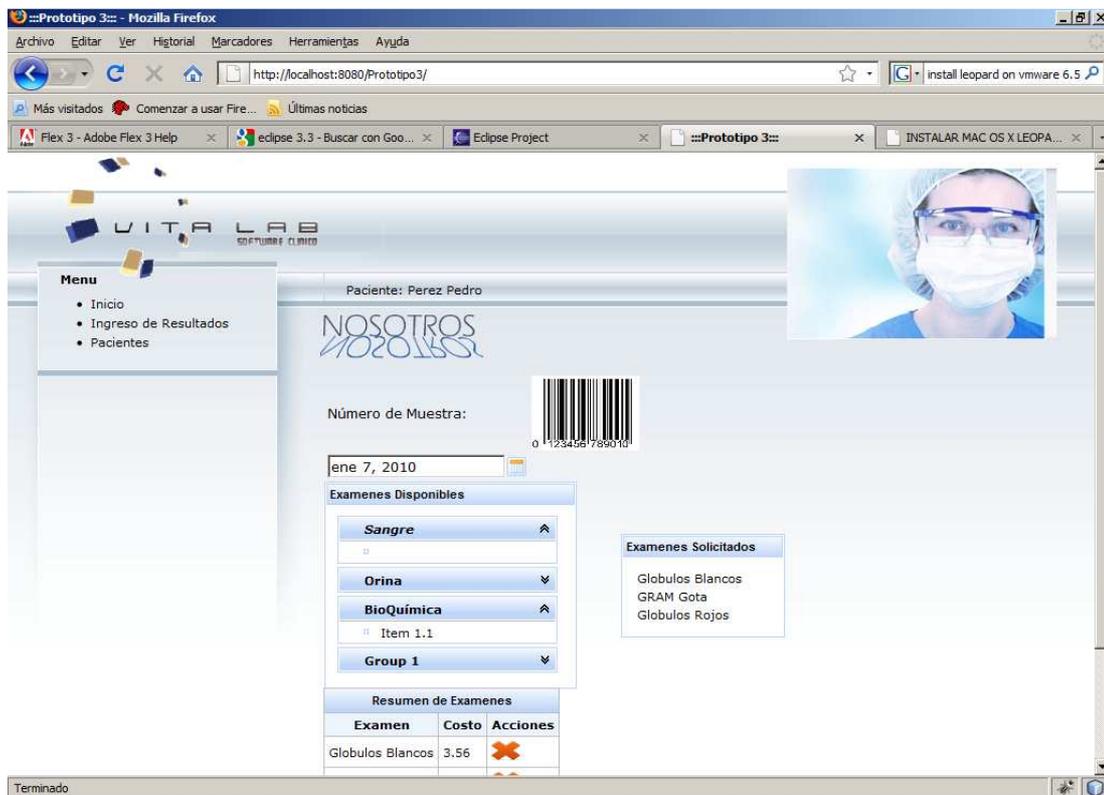


FIGURA 4.16 PROTOTIPO DE RICH FACES A RESOLUCIÓN DE 1024 X 768 PÍXELES

Fuente: El Autor

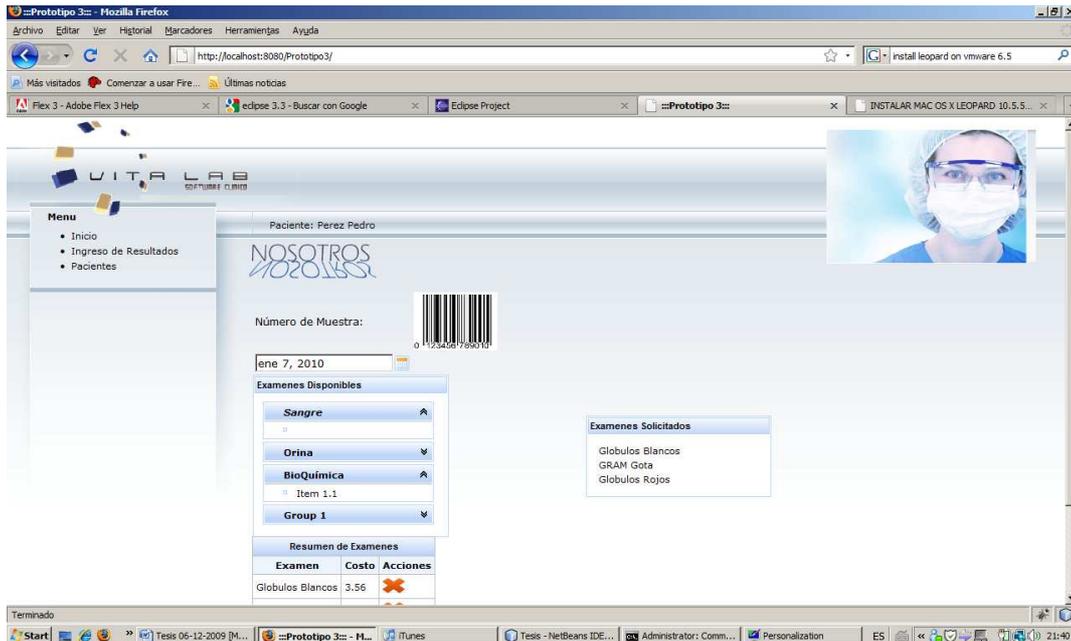


FIGURA 4.17 PROTOTIPO DE RICH FACES A RESOLUCION DE 1280 X 768 PÍXELES

Fuente: El Autor

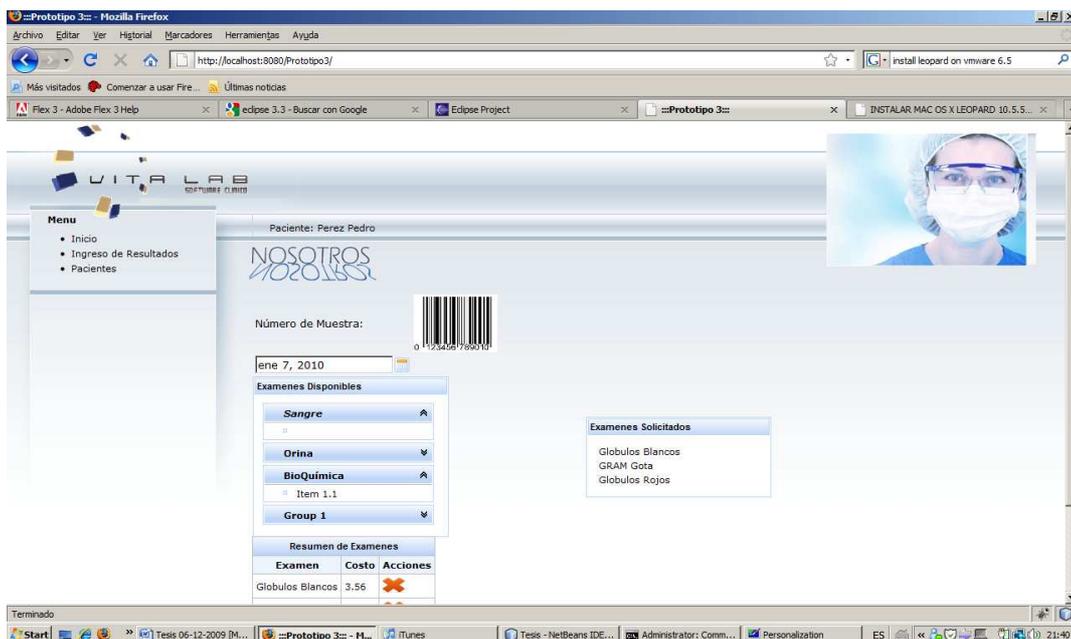


FIGURA 4.18 PROTOTIPO DE RICH FACES A RESOLUCIÓN DE 1280 X 800 PÍXELES

Fuente: El Autor

#### 4.1.5.3 Tiempo de descarga o tiempo de inicialización

Una vez realizadas las mediciones se obtuvo un tiempo promedio de descarga de para la página principal de la aplicación 1,57 segundos.

**Valoración final: 4**

Los reportes de las mediciones obtenidas se presentan en el Anexo3.

#### 4.1.5.4 Consumo de canal de red

El consumo de canal de red promedio obtenido por los reportes corresponde a 105 Kbps recibidos y 3,3 Kbps enviados, lo cual corresponde a 13,54 KBps.

$$\frac{1\text{Byte}}{8\text{bits}} \times \frac{108300\text{ bits}}{s} \times \frac{1\text{KB}}{1000\text{ bytes}} = 13,54 \frac{\text{KB}}{s}$$

#### Valoración final: 2

Los reportes de las mediciones obtenidas se presentan en el Anexo3.

#### 4.1.5.5 Requerimientos de Hardware y Software para el desarrollo

Como se menciona en la sección 2.3.3 Rich Faces requiere únicamente de las librerías propias del framework y la versión 1.5 del JDK de java como software para el desarrollo, la integración en un proyecto se da mediante ajustes especiales a determinados archivos de configuración, propios de un proyecto web de Java.

Considerando que el JDK es el único requerimiento de software específico para el desarrollo con Rich Faces, determina su total independencia de sistema operativo, al igual que el uso de Netbeans como IDE de desarrollo.

Respecto al hardware se utiliza el descrito en la sección 4.1, y como servidor de aplicaciones se utilizo Glassfish v2, el cual no requiere de ninguna configuración adicional a la predeterminada.

#### Valoración final 3

#### 4.1.5.6 Patrones de Diseño

Con Rich Faces se utilizo completamente el patrón MVC, tal como se propone en la especificación JSF, con lo cual se comprobó la capacidad para implementar este patrón de forma natural gracias a los requerimientos propios de JSF como el uso de backing beans y JSP a nivel de controlador y vista respectivamente, el modelo corresponde a un único proyecto que fue reutilizado para todos los prototipos.

#### Valoración final 4.

#### **4.1.5.7 Curva de aprendizaje**

El uso de características especiales de determinados componentes de Rich Faces como propiedades de drag and drop, efectos para paneles, etc requieren de mayor especialización, incluso con un conocimiento intermedio del lenguaje es necesario tomar en cuenta particularidades de los componentes. Lo cual refuerza la necesidad invertir un tiempo considerable para llegar a un nivel aceptable de experticia en el framework; el cual no es recomendable para usuarios sin conocimientos básicos en el desarrollo de aplicaciones JEE.

**Valoración final: 1**

#### **4.1.5.8 Herramientas para el desarrollo**

Si bien la integración de Rich Faces con Netbeans y Eclipse es sencilla y se hace mediante la incorporación de plugins, no se cuenta con un editor gráfico que acelere el proceso de desarrollo, lo cual disminuye considerablemente el tiempo de desarrollo.

La integración con Netbeans por su parte si facilita aspectos como configuración de algunos archivos propios de proyectos web con JSF, convirtiéndolo en un proceso transparente y ayudando al usuario a establecer su ambiente de desarrollo.

**Valoración final: 3**

#### **4.1.5.9 Facilidad de interacción con otras tecnologías**

Rich Faces como extensión a la especificación JSF, se integra de manera nativa con componentes desarrollados con Java, mas esto no es un impedimento ya que gracias a los avances del lenguaje y las arquitecturas de servicios puede fácilmente consumir web services desarrollados con otros lenguajes, esto a nivel de backend. A nivel de front end puede incluir código JavaScript y se da soporte a AJAX de forma nativa, el soporte multimedia y la inclusión de servicios de localización como Google maps son también soportados y de sencilla implementación.

**Valoración final: 4**

#### **4.1.5.10 Facilidad de puesta en producción**

Como cualquier aplicación web desarrollada con Java la puesta en producción es el tema que requiere mayor atención, la cantidad de servidores y la dependencia de archivos o de configuraciones específicas para cada uno de estos, determina un esfuerzo adicional para lograr el despliegue sin modificaciones en distintos servidores de aplicaciones.

Durante el despliegue del prototipo se escogió Glassfish por ser el servidor que cumple con las especificaciones propias de Sun, y que además se integra de forma nativa con Netbeans.

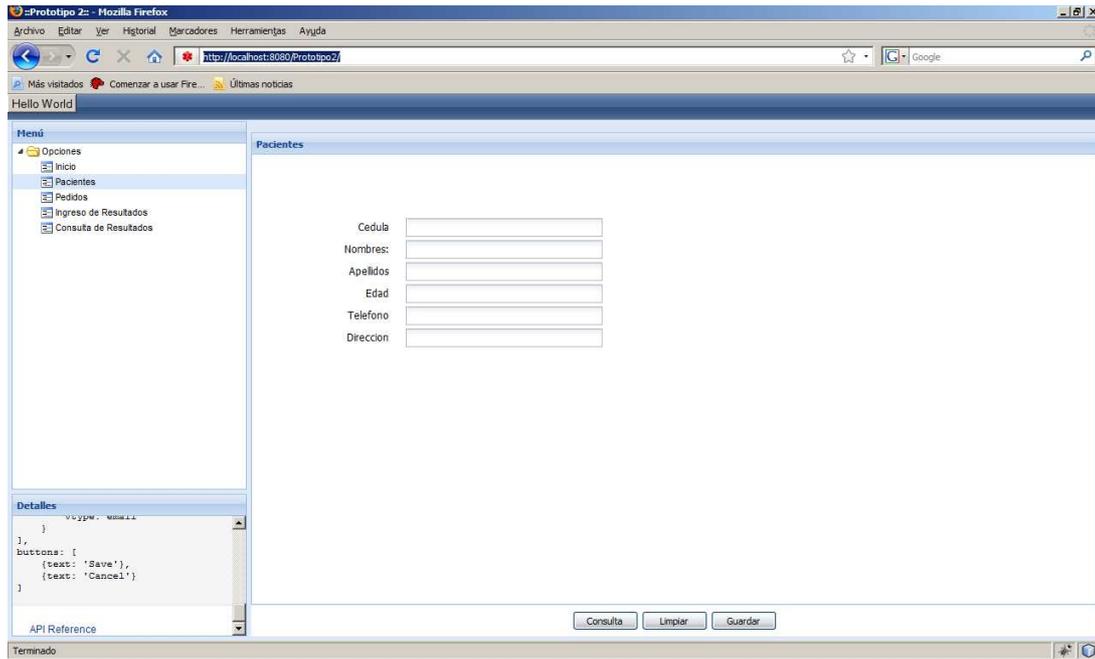
La estrecha relación de Netbeans con Glassfish ha determinado que el despliegue de cualquier tipo de aplicación se convierta en un proceso transparente para el desarrollador, lo cual sumado a la interfaz amigable de administración del servidor hacen que la puesta en producción de una aplicación realizada con Rich Faces no requiera de un esfuerzo mínimo.

**Valoración final: 1**

### **4.1.6 ANÁLISIS DE LAS CARACTERÍSTICAS DE EXT JS**

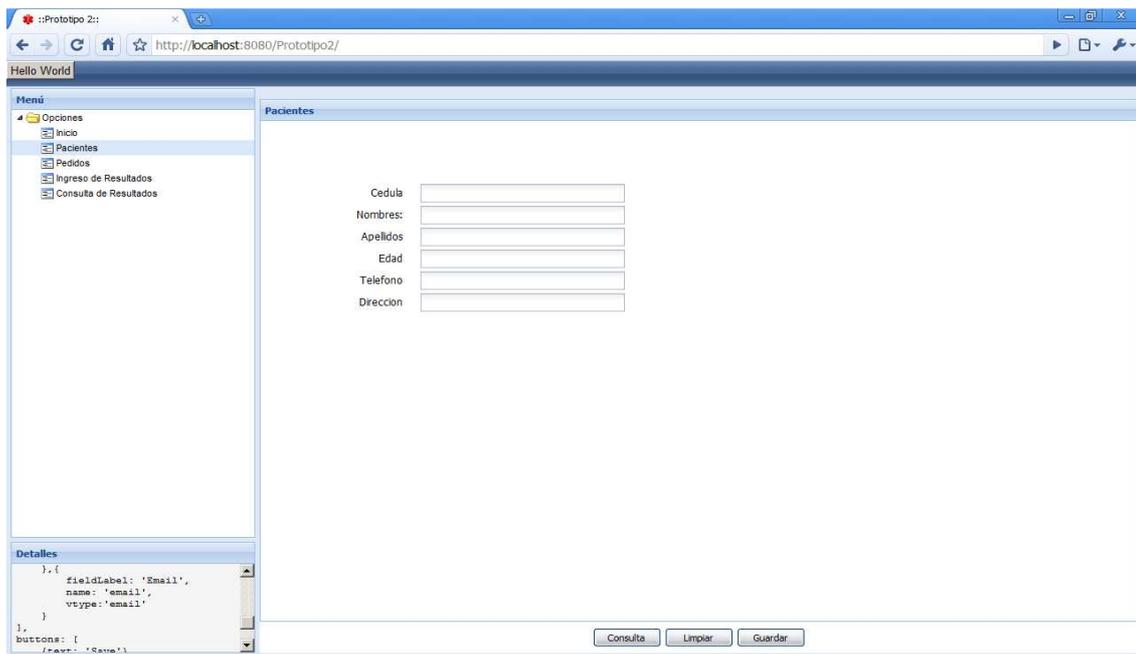
#### **4.1.6.1 Apariencia en los diferentes navegadores**

ExtJs conserva sus características en casi la totalidad de navegadores, sin embargo tiene como gran limitante que necesita permisos para poder ejecutar código javascript, en el caso de Internet Explorer V7 se no se tiene una presentación similar y esto es por las limitantes propias de este explorador.



**FIGURA 4.19** PROTOTIPO DE EXT JS EN MOZILLA FIREFOX

Fuente: El Autor



**FIGURA 4.20** PROTOTIPO DE EXT JS EN GOOGLE CHROME

Fuente: El Autor

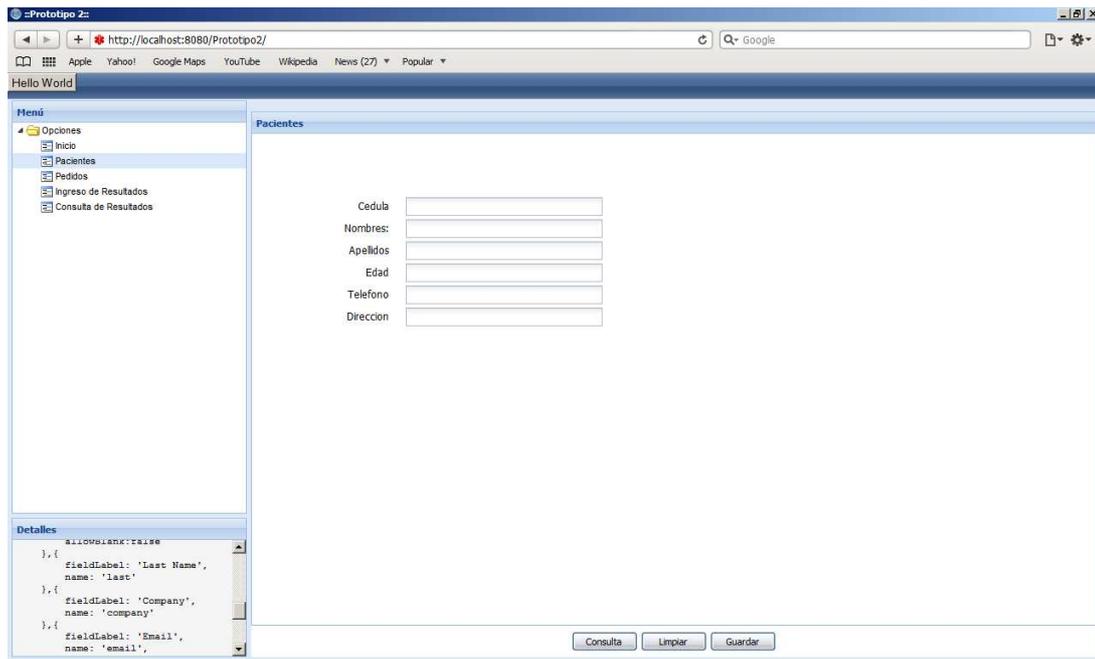


FIGURA 4.21 PROTOTIPO DE EXT JS EN SAFARI

Fuente: El Autor

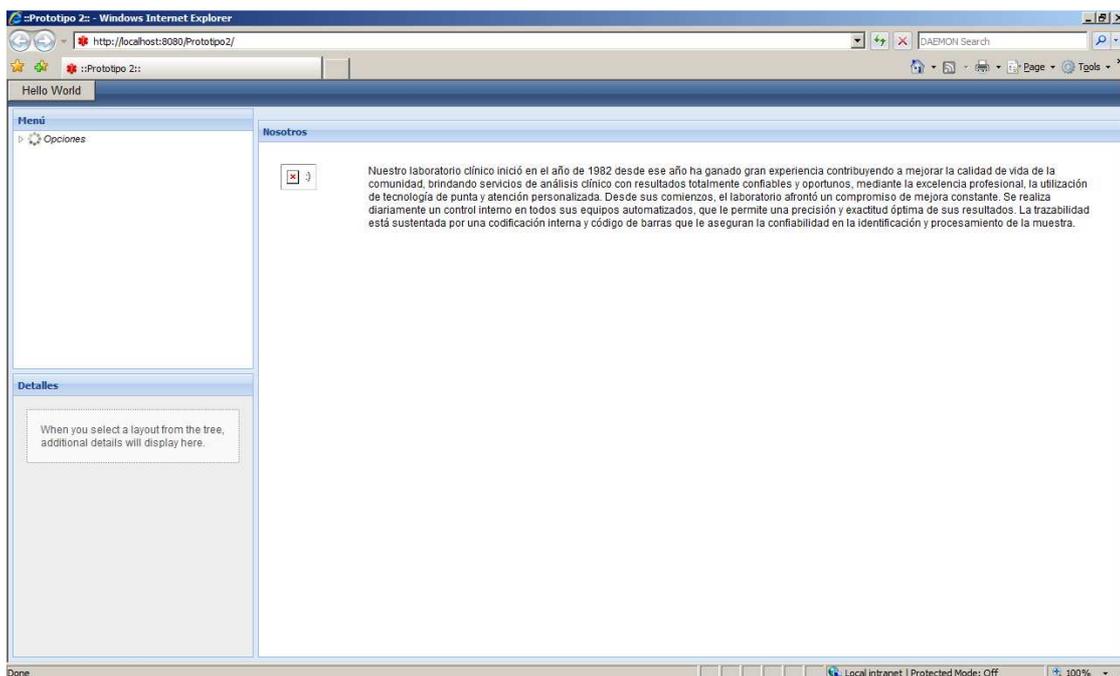


FIGURA 4.22 PROTOTIPO DE EXT JS EN SAFARI

Fuente: El Autor

## Valoración preliminar: 4

### 4.1.6.2 Independencia de la resolución

Ext Js es el framework que mejor conserva las características gráficas de sus componentes, sin importar el cambio de resolución los componentes conservan

exactamente la relación de tamaño y ubicación en el área disponible, a excepción de componentes en los cuales se haya establecido un atributo estático de ancho y alto.

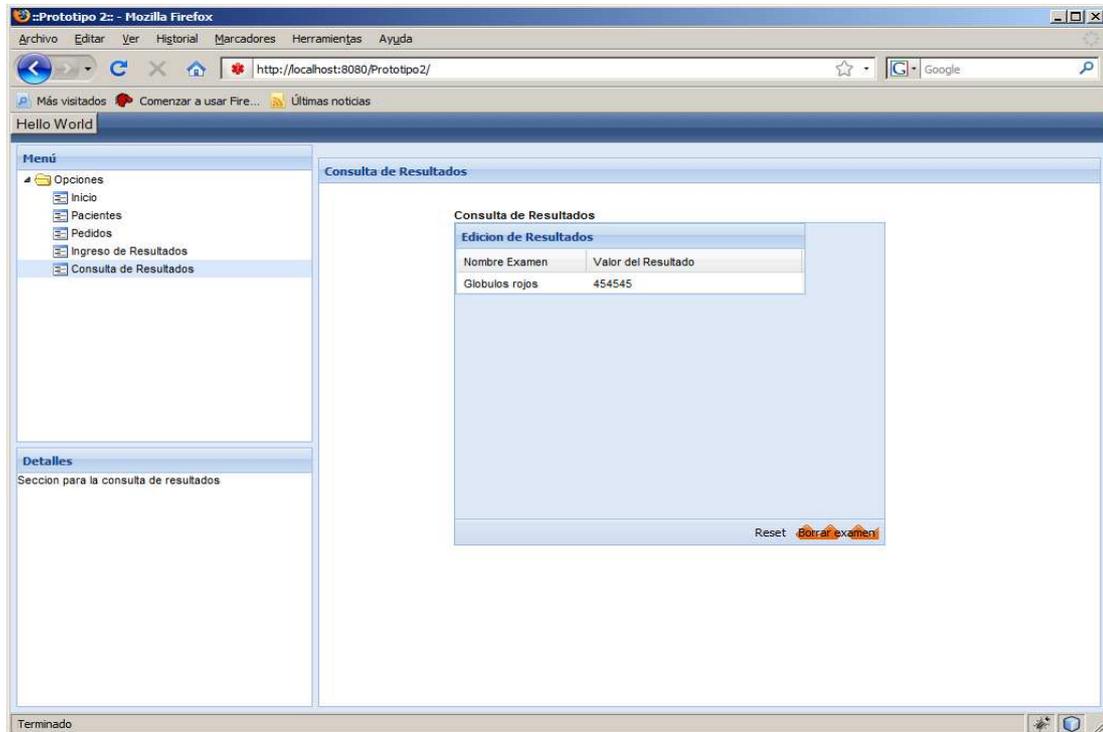


FIGURA 4.23 PROTOTIPO DE EXT JS A RESOLUCIÓN DE 1280 X 768

Fuente: El Autor

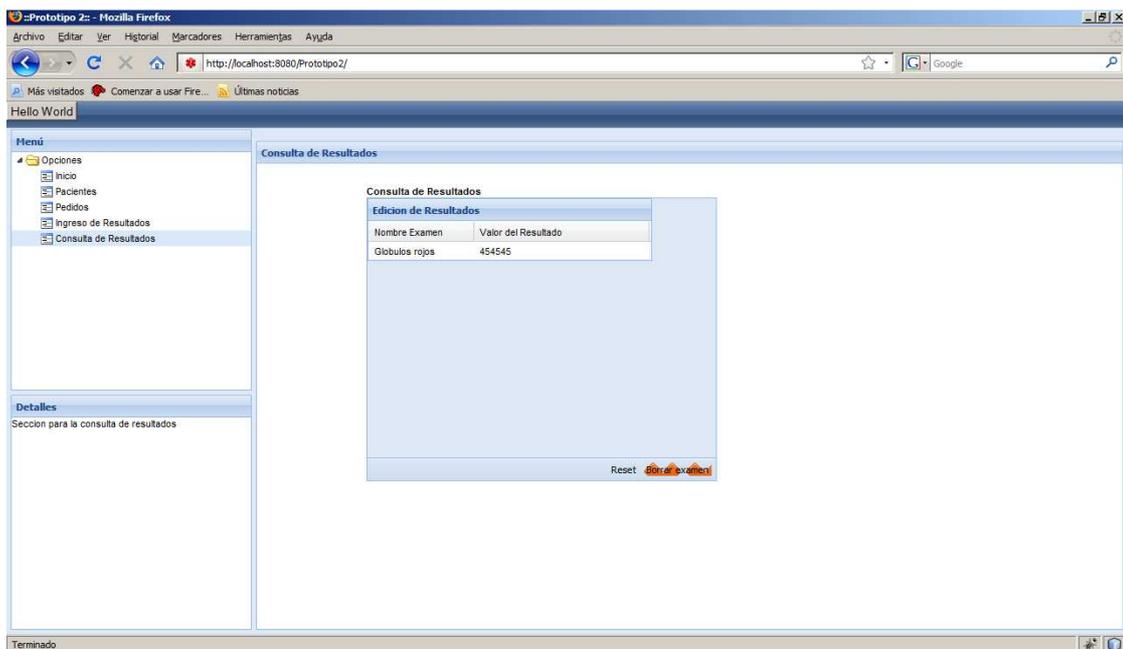


FIGURA 4.24 PROTOTIPO DE EXT JS A RESOLUCIÓN DE 1280 X 768

Fuente: El Autor

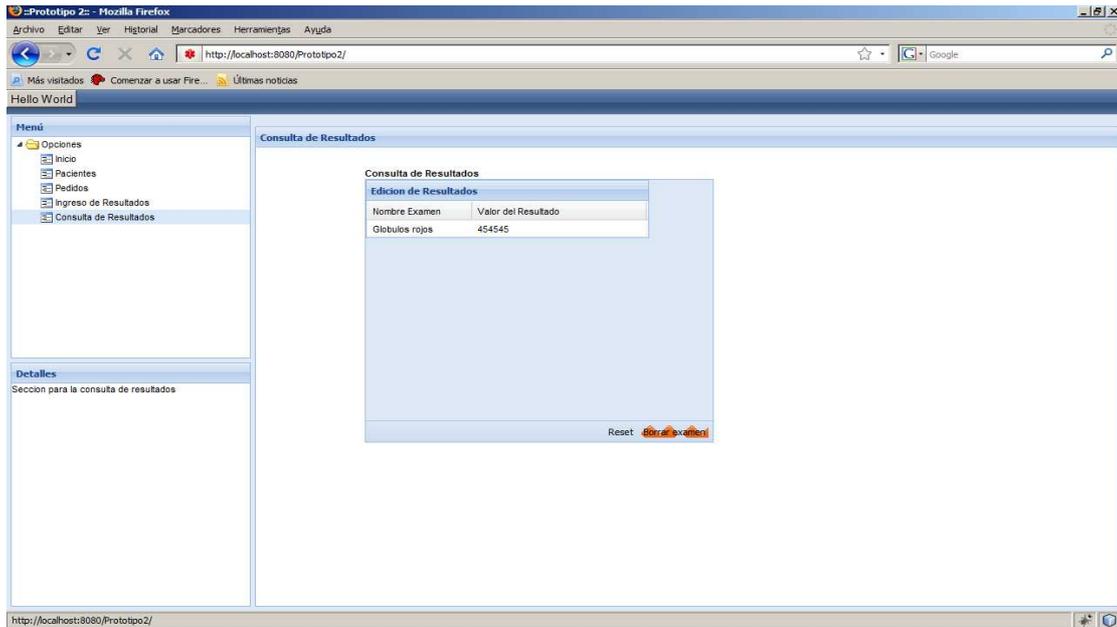


FIGURA 4.25 PROTOTIPO DE EXT JS A RESOLUCIÓN DE 1280 X 800

Fuente: El Autor

## Valoración final: 2

### 4.1.6.3 Tiempo de descarga o tiempo de inicialización

Una vez realizadas las mediciones se obtuvo un tiempo promedio de descarga de para la página principal de la aplicación 1,64 segundos.

## Valoración final: 3

Los reportes de las mediciones obtenidas se presentan en el Anexo3.

### 4.1.6.4 Consumo de canal de red

El consumo promedio de canal de red es de 53 kbps recibidos y 22,5 kbps, lo cual sumado y debidamente transformado equivale a 9,44 KBps.

$$\frac{1\text{Byte}}{8\text{bits}} \times \frac{75500\text{ bits}}{s} \times \frac{1\text{KB}}{1000\text{ bytes}} = 9.44 \frac{\text{KB}}{s}$$

## Valoración final: 4

Los reportes de las mediciones obtenidas se presentan en el Anexo3.

### 4.1.6.5 Requerimientos de Hardware y Software para el desarrollo

Ext Js no requirió de ningún tipo de software o hardware adicional al descrito en la sección 4.1.1, el cual se ajusta al requerido en la sección 1.2.3.4.

## Valoración final 4

#### **4.1.6.6 Patrones de Diseño**

Dependiendo del punto de vista el hecho de que los componentes nativos de Ext Js expuestos a nivel de front end no puedan ser mapeados y manipulados mediante clases intermedias o controladores puede considerarse un acierto o un error. Durante el desarrollo y por el enfoque del caso de aplicación lo ya expuesto se convirtió en un gran limitante, al intentar implantar un patrón de diseño, se intento una aproximación mediante la comunicación de la vista con los componentes controladores mediante conexiones HTTP del tipo post; lo cual determina la imposibilidad del framework para implementar patrones de diseño con sus componentes propios. Esto afecta de forma significativa a la valoración de este criterio.

**Valoración final: 1**

#### **4.1.6.7 Curva de aprendizaje**

La experiencia en durante el desarrollo del prototipo arroja un resultado negativo, si consideramos la falta de facilidades para seguir un patrón de diseño Ext Js, la incapacidad de consumo de web services que no sean del tipo JASON, la falta de experiencia con lenguajes como Java Script y la limitada ayuda de los IDE´s de desarrollo.

Este prototipo fue el que requirió mucho más tiempo para conseguir resultados aceptables, debido a la necesidad de crear componentes externos que permitan la integración con el backend común, la manipulación de las interfaces y componentes propios del framework.

**Valoración final: 1**

#### **4.1.6.8 Herramientas para el desarrollo**

Si bien en la evaluación previa de la sección 2.4.11 y la descripción del capítulo 1 se mencionan varios IDE´s para el soporte de ExtJs, ninguno permite de manera satisfactoria un desarrollo ágil de aplicaciones. La edición por lo menos al nivel de código no cumple por completo su cometido y a la vez no incorporan capacidad de ejecución de pruebas y depuración de código.

Firebug se convirtió en una gran ayuda para suplir las carencias propias del framework en el tema de depuración y pruebas.

Al momento de la realización del prototipo (últimos meses de 2009) se conoció de la existencia de un designer preview propio del desarrollador de ExtJS, el cual no permite la generación o edición de código sino simplemente es un preview de interfaces.

Sin embargo para la valoración del criterio y el desarrollo del prototipo se utilizó el framework puro con todas las limitantes ya expuestas.

**Valoración final: 1**

#### **4.1.6.9 Facilidad de interacción con varias tecnologías**

Ext Js se puede integrar con la mayoría de lenguajes destinados a la creación de aplicaciones web, extendiendo las capacidades visuales de estos, sin embargo si es usado totalmente como capa de front end puede interactuar de forma natural con servicios del tipo JSON o mediante conexión HTTP por POST o GET, este tipo de conexión si bien le da flexibilidad para la interacción también es un limitante con aplicaciones orientadas a servicios ya que requieren de una envoltura especial para lograr esta interacción.

De la documentación que se pudo revisar se notó que la mejor tecnología para el acoplamiento del framework puro es PHP (*PHP Hypertext Pre-processor*), con la cual se obtiene mejores resultados.

La opción más interesante y la que requirió de mayor atención es la de Coolite Toolkit, la cual es una extensión de componentes de ExtJs desarrollada de manera exclusiva para plataforma punto Net y que se integra nativamente con aplicaciones tipo ASP (Active Server Pages). Esta hace uso además de Visual Studio y todas sus ventajas como IDE de desarrollo. Esta es quizás la mejor opción para los desarrolladores.

**Valoración final: 2**

#### **4.1.6.10 Facilidad de puesta en producción**

El servidor elegido para el despliegue del prototipo de ExtJs es Glassfish, esto porque se utilizaron como componentes de enlace con la capa de Backend Servlets desarrollados en tecnología Java, la elección se hizo en base a la capacidad de integración con el Netbeans y las facilidades de administración.

Cabe señalar que de haber escogido otro tipo de tecnología se tendría un nivel igual o superior de dependencia del servidor de aplicaciones, situación que no se presentaría con una aplicación pura de Ext Js.

**Valoración final: 2**

## **4.2 COMPARACIÓN DE FRAMEWORKS**

Se utilizará a continuación la matriz de priorización para determinar cuál de los frameworks es el más adecuado para la aplicación considerada.

Se presenta en primer lugar la matriz de priorización considerando los valores sin multiplicarlos por su factor de preferencia, luego la matriz con los valores multiplicados por su valor de preferencia, la cual será la considerada de forma definitiva en la priorización.

## Matrices de Priorización

Opciones	CRITERIOS																TOTAL
	AP1	AP2	RE1	RE2	REC1	REC2	ARQ1	ARQ2	SOP1	SOP2	SOP3	DES1	DES2	DES3	DES4		
FLEX	4	4	2	3	3	3	2	2	4	4	4	3	4	4	4	50	
RICH FACES	4	4	4	2	3	2	2	4	4	4	4	1	3	4	2	47	
EXT JS	4	4	3	4	4	4	1	3	4	3	2	1	1	2	4	44	

**TABLA 27. MATRIZ DE PRIORIZACIÓN SIN VALORES DE PREFERENCIA**

Elaborado por: El autor

Matriz de priorización con los valores de preferencia ponderados

Opciones	CRITERIOS																TOTAL
	AP1	AP2	RE1	RE2	REC1	REC2	ARQ1	ARQ2	SOP1	SOP2	SOP3	DES1	DES2	DES3	DES4		
FLEX	0,15	0,06	0,18	0,32	0,14	0,16	0,19	0,13	0,27	0,12	0,21	0,31	0,31	0,34	0,28	3,18	
RICHFACES	0,15	0,06	0,36	0,21	0,14	0,10	0,19	0,26	0,27	0,12	0,21	0,10	0,24	0,34	0,14	2,91	
EXT JS	0,15	0,06	0,27	0,43	0,16	0,22	0,09	0,19	0,27	0,09	0,10	0,10	0,07	0,17	0,28	2,71	

**TABLA 28. MATRIZ DE PRORIZACIÓN CON VALORES DE PREFERENCIA**

Elaborado por: El autor

### 4.3 ANÁLISIS DE RESULTADOS

Realizadas las mediciones y tabulados los datos en la matriz de priorización se obtienen los siguientes análisis a partir de los resultados:

- El framework más adecuado para la aplicación propuesta (con mediana transaccionalidad y focalizada en un área de negocio específica) es Adobe Flex, no solo por su riqueza visual sino por las facilidades que brinda al desarrollador.
- Las características comunes de los frameworks son su independencia de navegador, independencia de la resolución, buena documentación y soporte amplio por sus fabricantes así como la comunidad de desarrolladores.
- Para el desarrollo de aplicaciones con Rich Faces se requiere un nivel elevado de conocimiento sobre la tecnología JEE ya que funciona de manera interrelacionada con componentes de esta arquitectura.
- El punto más fuerte para Rich Faces es el uso de patrones de diseño aceptados y definidos y que pueden ser identificados en sus componentes, lo cual a nivel arquitectónico le da gran valor.
- No es recomendable utilizar Ext JS en aplicaciones con gran soporte para la integración con servicios y alta interacción con back end; se requiere complementarlo con otras tecnologías que fortalezcan este tipo de interacciones.
- El uso de Ext JS es recomendable para realizar portales informativos con baja interacción con la capa de backend y permitiendo explotar sus características visuales.
- Ext JS conserva problemas heredados de JavaScript como la necesidad de autorización para ejecutarlo en el ambiente del cliente, y las limitaciones propias del lenguaje.



## CAPITULO 5. CONCLUSIONES Y RECOMENDACIONES

### 5.1 CONCLUSIONES

- Las aplicaciones RIA (Rich Internet Applications) integran las ventajas de las aplicaciones web como: facilidad de despliegue y mantenimiento, actualización centralizada, independencia de plataforma y una mejor experiencia para el usuario gracias a: riqueza multimedia y visual, familiaridad con controles y ausencia del efecto de refresco en las páginas.
- A fin de escoger una plataforma adecuada para la implementación de aplicaciones RIA es importante determinar sus beneficios y debilidades en base a criterios de evaluación. En este trabajo se han establecido 15 criterios referenciales sobre: apariencia, rendimiento, recursos, arquitectura, soporte y facilidades para el desarrollo, que servirán como base para las organizaciones a fin de seleccionar el framework más idóneo de acuerdo a sus necesidades específicas.
- Con la implementación de los prototipos fue posible validar las características de cada framework así como valorar el cumplimiento de los criterios descritos y analizados teóricamente. Por ejemplo en el caso de Ext JS, teóricamente se mencionaba el soporte para el desarrollo de varios IDE's al menos para el nivel de código, sin embargo en la práctica ninguno de los IDE's sugeridos cumplió las expectativas durante el desarrollo.
- La experiencia del desarrollo de prototipos permitió notar que es necesario aumentar el esfuerzo para la obtención de un IDE con entorno de desarrollo gráfico, así como soporte para una efectiva integración con la capa de back end por parte de todos los frameworks de desarrollo, mejorando de esta forma el tiempo de implementación y disminuyendo la curva de aprendizaje para desarrolladores novatos.
- De acuerdo al análisis de resultados de la evaluación, se pudo determinar:
  - Adobe Flex es el framework con mayores facilidades gráficas de desarrollo, ofreciendo capacidades para crear no solo aplicaciones

basadas en un navegador, sino también aplicaciones sobre escritorio, utilizando el mismo lenguaje y componentes. Entre sus mayores desventajas se cuentan la falta de patrones de diseño formales y el manejo de las sesiones de usuario.

- Rich Faces es el framework más sólido para el diseño arquitectónico gracias al respaldo de un conjunto amplio de especificaciones como JEE para la implementación de aplicaciones. Pese a carecer de un entorno totalmente gráfico para el desarrollo de aplicaciones es una alternativa atractiva si no se desea realizar una gran inversión en licenciamiento.
- Ext Js es un framework con una gran riqueza visual la cual no puede ser correctamente aprovechada con la aplicación desarrollada, a partir de la investigación e implementación se pudo notar que su tendencia es el desarrollo de aplicaciones con poca interacción con un back end complejo.

## **5.2 RECOMENDACIONES**

- Para complementar el estudio realizado resultará beneficioso extender el análisis a frameworks RIA que permiten el desarrollo de aplicaciones desktop based (aplicaciones RIA que no requieren de un navegador para su funcionamiento), con esto se conseguiría completar la visión del espectro que abarca la definición de RIA.
- Para complementar el estudio realizado se debería considerar el análisis de frameworks adicionales para RIA, como es el caso de Silverlight de Microsoft y Java Fx de Sun Oracle, los durante el periodo de desarrollo de este trabajo presentaron grandes avances y aceptación en el mercado.

- Para la selección de un framework de desarrollo de aplicaciones RIA en una organización se puede tomar como base los 15 criterios establecidos en este trabajo, y escoger los más adecuados para las necesidades particulares así como sus ponderaciones, para determinar un valor final se sugiere utilizar la matriz de comparación.
- Una vez seleccionado un framework de desarrollo resulta necesario establecer estándares y patrones de diseño que aseguren una arquitectura sólida y saquen provecho de dicho framework.

## REFERENCIAS

- [1] MORITZ Florian. **Rich Internet Applications (RIA): A Convergence of User Interface Paradigms of Web and Desktop Exemplified by JavaFX.** <http://www.flomedia.de/diploma/>.  
27/septiembre/2008
- [2] ADOBE. **Creating rich Internet applications with the Adobe technology platform.** White Paper obtenido de <http://www.adobe.com>,  
27/septiembre/2008
- [3] GROSSMAN Gary, HUANG Emmy. **ActionScript 3.0 overview.** [http://www.adobe.com/devnet/actionscript/articles/actionscript3\\_overview.html](http://www.adobe.com/devnet/actionscript/articles/actionscript3_overview.html).  
12/marzo/2009
- [4] Anónimo. **ECMA.** <http://es.wikipedia.org/wiki/ECMA>  
13/marzo/2009
- [5] HERNANDEZ Cristian. **Flex: Cairngorm Framework.** <http://www.cristianhernandez.org/blog/?p=78>.  
3/mayo/2009

- [6] BURLESON Thomas, Shuman Leo , Adobe Customer Training/Partner Enablement Group. **Introducing Cairngorm.**  
[http://www.adobe.com/devnet/flex/articles/introducing\\_cairngorm.html](http://www.adobe.com/devnet/flex/articles/introducing_cairngorm.html)  
3/mayo/2009
- [7] Anónimo. **About Cairngorm.**  
<http://opensource.adobe.com/wiki/display/cairngorm/About>  
7/mayo/2009
- [8] Adobe Systems Inc. **Adobe Store**  
[https://store1.adobe.com/cfusion/store/index.cfm?&store=OLS-US&catType=SUPPORT#store=OLS-US&view=ols\\_cat&catID=SUPPORT&loc=en\\_us](https://store1.adobe.com/cfusion/store/index.cfm?&store=OLS-US&catType=SUPPORT#store=OLS-US&view=ols_cat&catID=SUPPORT&loc=en_us)  
18/mayo/2009
- [9] Adobe Systems Inc. **Flex SDK**  
<http://opensource.adobe.com/wiki/display/flexsdk/Flex+SDK;jsessionid=49204D327066235307981ED603E848AB>  
3/mayo/2009
- [10] Anónimo. **Discovery-earth.** <http://dsc.discovery.com/discovery-earth-live>.  
23/mayo/2009
- [11] Anónimo. **Rich Faces Developer Guide.** [http://www.jboss.org/file-access/default/members/jbossRichFaces/freezone/docs/devguide/en/html\\_single/index.html](http://www.jboss.org/file-access/default/members/jbossRichFaces/freezone/docs/devguide/en/html_single/index.html)  
7/junio/2009
- [12] Anónimo, **Rich Faces.** [http://en.wikipedia.org/wiki/Rich\\_Faces](http://en.wikipedia.org/wiki/Rich_Faces)  
[13/junio/2009](http://en.wikipedia.org/wiki/Rich_Faces)
- [13] Anónimo. **RED Hat Store**  
<https://www.redhat.com/apps/store/developers/subscription.html>  
15/mayo/2009
- [14] Anónimo. **Rich Faces in Production**  
<http://www.jboss.org/community/wiki/RichFacesInProduction>  
18/mayo/2009

- [15] SHEA Frederick, RAMSAY Colin, BLADES Steve. **Learning Ext JS Build dynamic, desktop-style user interfaces for your data-driven web applications.** First Edition. Packt Publishing Ltd. Birmingham. 2008.
- [16] Anónimo. **Desing Paterns.**  
[http://en.wikipedia.org/wiki/Gang\\_of\\_Four\\_\(software\)](http://en.wikipedia.org/wiki/Gang_of_Four_(software))  
 14/junio/2009
- [17] Ext Js Inc. **Ext JS Cross-Browser Rich Internet Application Framework.** <http://www.extjs.com/products/extjs/>  
 1/junio/2009
- [18] Exadel. **Rich Faces Live Demo.** [http://livedemo.exadel.com/Rich\\_Faces-demo/RichFaces/actionparam.jsf?c=actionparam&tab=usage](http://livedemo.exadel.com/Rich_Faces-demo/RichFaces/actionparam.jsf?c=actionparam&tab=usage)  
 18/noviembre/2009
- [19] Adobe Systems Inc. **Adobe Live Docs.** <http://livedocs.adobe.com/flex/>  
 25/noviembre/2009
- [20] Softlogica. **WAPT.** [http://www.loadtestingtool.com/`](http://www.loadtestingtool.com/)  
 12/enero/2010
- [21] LockTime Software. **NetLimiter-Ultimate BandWith shaper**  
<http://www.netlimiter.com/>  
 13/enero/2010
- [22] Anónimo **Herramientas de priorización: Tomar decisiones entre distintas opciones** <http://www.ongconcalidad.org/priorizacion.pdf>  
 26/noviembre/2009

## **ANEXOS**

## Anexo 1

### Tabla de Utilización de los Navegadores a Julio de 2009

Month	Internet Explorer	Firefox	Safari	Chrome	Opera	Netscape	Other
<a href="#">September, 2008</a>	74.18%	19.07%	2.82%	1.10%	2.01%	0.41%	0.41%
<a href="#">October, 2008</a>	73.64%	19.60%	2.87%	1.00%	2.12%	0.39%	0.38%
<a href="#">November, 2008</a>	71.61%	21.21%	3.08%	1.11%	2.17%	0.44%	0.38%
<a href="#">December, 2008</a>	70.50%	21.69%	3.41%	1.40%	2.17%	0.42%	0.41%
<a href="#">January, 2009</a>	69.72%	22.11%	3.62%	1.52%	2.23%	0.43%	0.37%
<a href="#">February, 2009</a>	69.23%	22.58%	3.47%	1.54%	2.19%	0.62%	0.37%
<a href="#">March, 2009</a>	68.46%	23.30%	3.63%	1.62%	2.12%	0.50%	0.37%
<a href="#">April, 2009</a>	67.77%	23.84%	3.53%	1.79%	2.04%	0.64%	0.39%
<a href="#">May, 2009</a>	68.10%	22.75%	3.70%	2.18%	2.06%	0.81%	0.40%
<a href="#">June, 2009</a>	68.32%	22.43%	3.79%	2.40%	2.03%	0.60%	0.42%
<a href="#">July, 2009</a>	67.68%	22.47%	4.07%	2.59%	1.97%	0.67%	0.55%

Fuente <http://marketshare.hitslink.com/report.aspx?qprid=1>

## Anexo 2

### Configuración de WAPT para las pruebas de los Prototipos

#### 1. Tipo de Prueba

El tipo de prueba escogido fue prueba de rendimiento



FIGURA A2.1 CONFIGURACIÓN DEL ESCENARIO DE PRUEBAS WAPT PASO 1

Fuente: El Autor

#### 2. Nivel de carga y duración

EL número de usuarios se escoge en un rango de cero a veinte con intervalos de cuatro segundos y se establece el tiempo de la prueba a tres minutos.

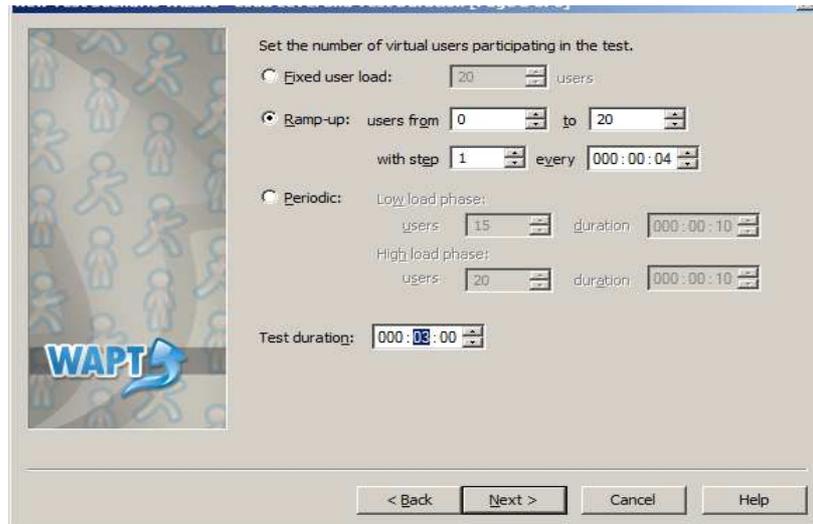


FIGURA A2.2 CONFIGURACIÓN DEL ESCENARIO DE PRUEBAS WAPT PASO 2

Fuente: El Autor

### 3. Inicialización de características de la conexión

La conexión escogida es la DSL de 256Kbps

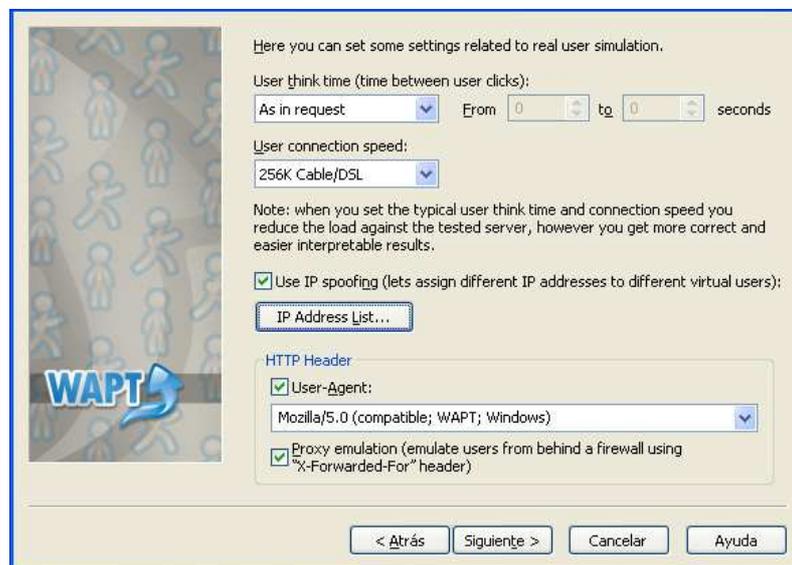


FIGURA A2.3 CONFIGURACIÓN DEL ESCENARIO DE PRUEBAS WAPT PASO 3

Fuente: El Autor

### 4. Finalización

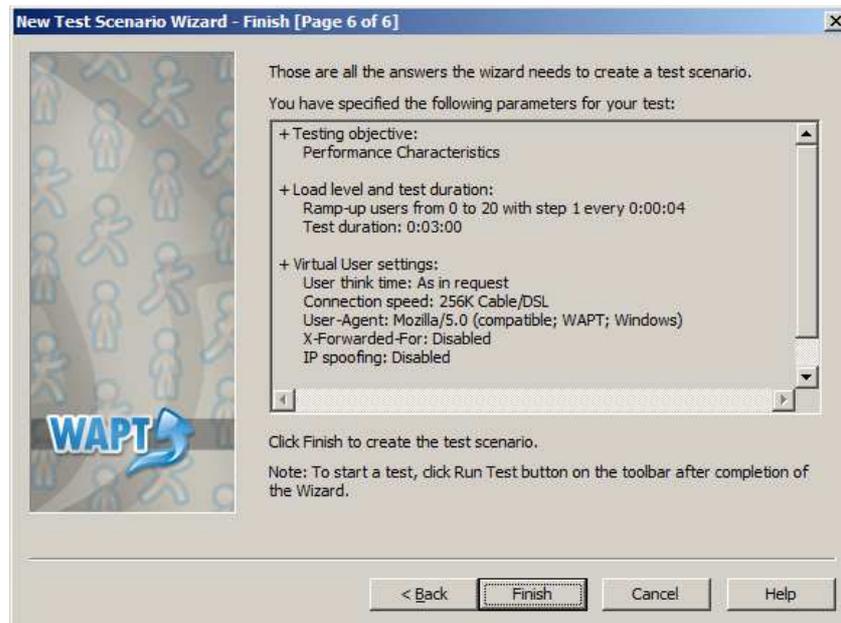


FIGURA A2.4 RESUMEN Y FINALIZACIÓN DE CONFIGURACIÓN DE LA PRUEBA

Fuente: El Autor

## Configuración de NetLimiter para las pruebas de los Prototipos

### 1. Selección de la Red

Dentro de la opción de Administración de Redes, seleccionamos la red local que queremos limitar.

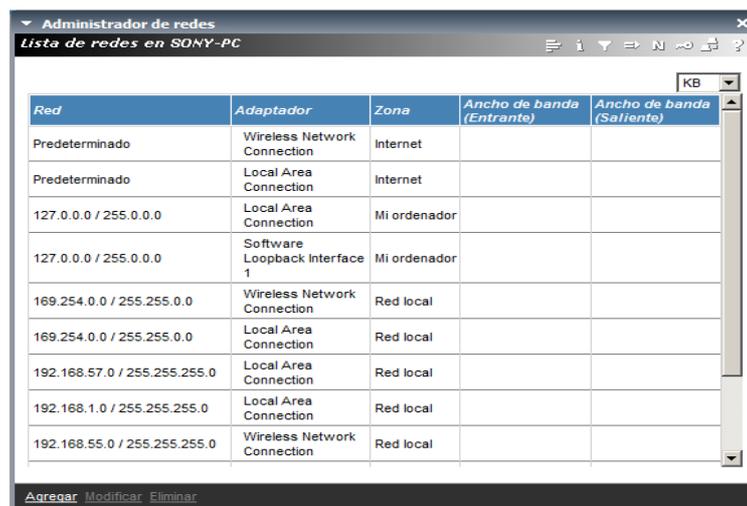


FIGURA A2.5 ADMINISTRADOR DE REDES DE NET LIMITER

Fuente: El Autor

### 2. Determinación de Ancho de Banda

Se procede luego a establecer los valores de ancho de banda entrante y saliente, para la red seleccionada



FIGURA A2.6 CONFIGURACIÓN DE ANCHO DE BANDA

Fuente: El Autor

## Configuración de Performance Monitor

### 1. Configuración del registro

En la pantalla principal de Performance Monitor se seleccionó la opción Nuevo Registro, en el árbol de Registros y alertas de rendimiento.

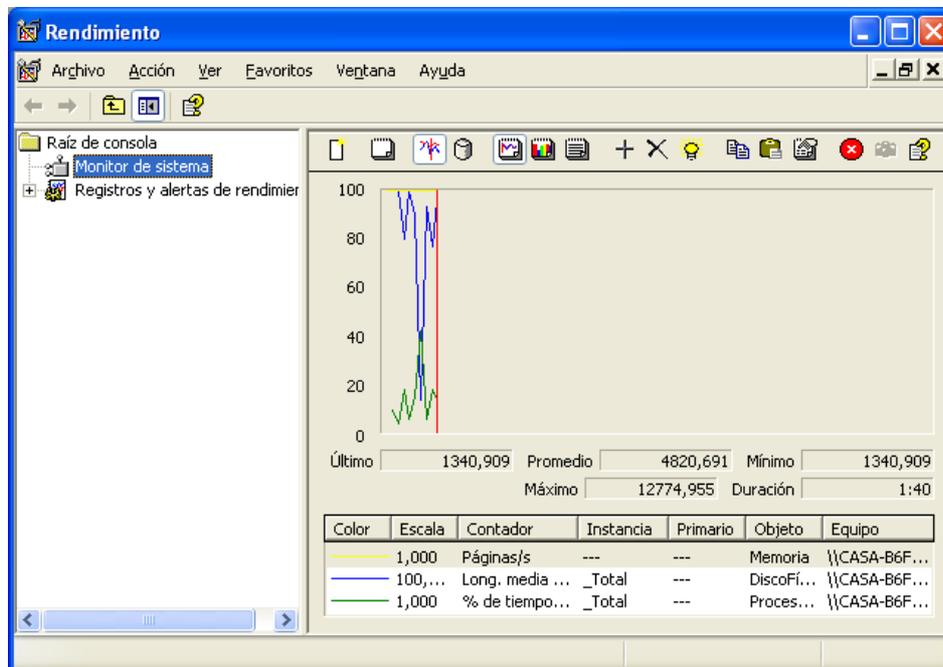


FIGURA A2.7 PANTALLA PRINCIPAL DE PERFORMANCE MONITOR

Fuente: El Autor

## 2. Selección de contadores

Una vez creado el nuevo registro se procede a asignarle los contadores para las mediciones respectivas, en este caso en particular el objeto es la interfaz de red, para la cual se seleccionan los contadores de paquetes recibidos por segundo, paquetes por segundo y total de bytes por segundo.

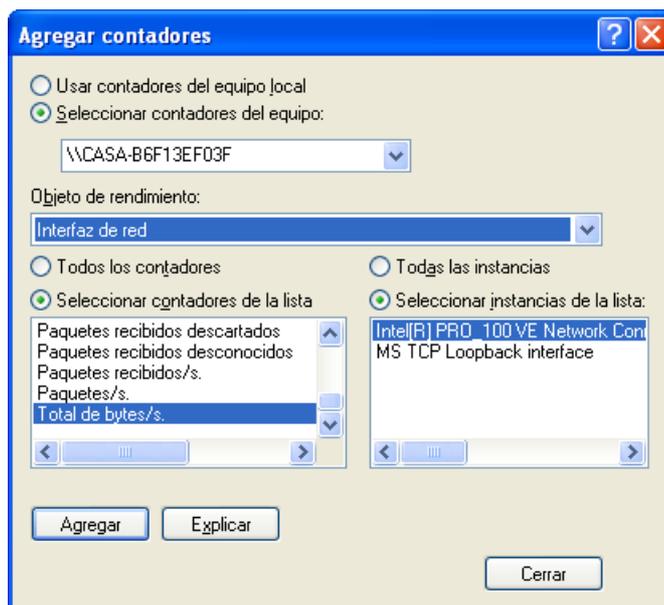


FIGURA A2.8 SELECCIÓN DE CONTADORES PARA LA INTERFAZ DE RED

Fuente: El Autor

## 3. Configuración de archivo de Registro

Agredados los contadores, se presenta el resumen de propiedades del registro en el cual es posible seleccionar el tipo de registro y la dirección donde se almacenarán los resultados.



FIGURA A2.9 SELECCIÓN DE CONTADORES PARA LA INTERFAZ DE RED

Fuente: El Autor

#### 4. Resumen del Registro

Finalmente se muestra el resumen del registro creado.

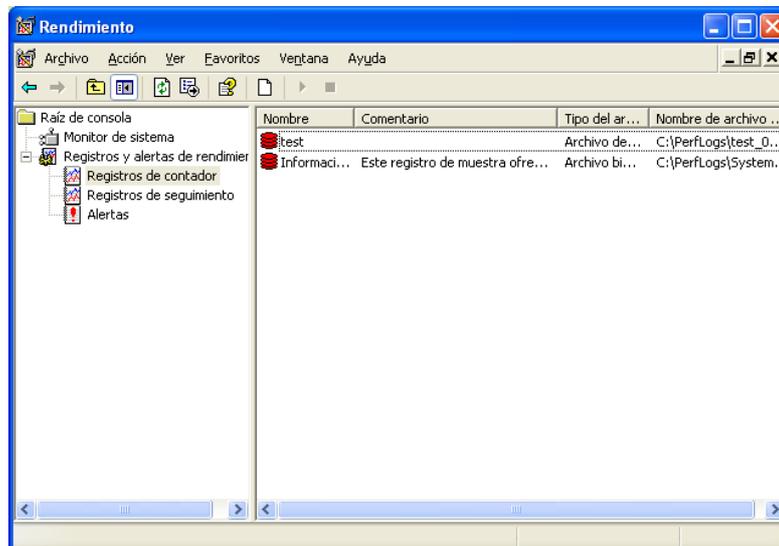


FIGURA A2.10 RESUMEN DEL REGISTRO CREADO

Fuente: El Autor

## Anexo 3

### Reporte del Prototipo de Ext JS

**Test execution parameters:**

Test status: finished  
 Test started at: 24/01/2010 0:29:37  
 Test finished at: 24/01/2010 0:32:42  
 Scenario name: prototipo2.wps  
 Test run comment:  
 Test executed by: Casa  
 Test executed on: CASA-B6F13EF03F  
 Test duration: 0:03:00  
 Virtual users: 0 - 20

**Summary**

Profile	Sessions performed	Sessions with errors	Pages performed	Pages with errors	Hits performed	Hits with errors	Total KBytes sent	Total KBytes received
Prototipo2	645	0	1.290	0	1.935	645	506	14.929
Total	645	0	1.290	0	1.935	645	506	14.929

**Number of active users**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00
Prototipo2	4	9	12	17	20	20	20	20	20	20
Total	4	9	12	17	20	20	20	20	20	20

**Sessions**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	44	73	36	71	70	69	77	41	71	93	645
Total	44	73	36	71	70	69	77	41	71	93	645

**Sessions per second**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	2,2	3,65	3,6	3,55	3,5	3,45	3,85	4,1	3,55	4,65	3,58
Total	2,2	3,65	3,6	3,55	3,5	3,45	3,85	4,1	3,55	4,65	3,58



**Hits with errors**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	44	73	36	71	70	69	77	41	71	93	645
Total	44	73	36	71	70	69	77	41	71	93	645

**KBytes sent**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	20,8	34,6	17,1	33,6	33,2	32,9	36,7	19,5	33,8	44,3	307
Total	20,8	34,6	17,1	33,6	33,2	32,9	36,7	19,5	33,8	44,3	307

**KBytes received**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	979	1.598	788	1.554	1.532	1.510	1.686	897	1.554	2.036	14.135
Total	979	1.598	788	1.554	1.532	1.510	1.686	897	1.554	2.036	14.135

**Response time, sec**

Name	Time	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00
Prototipo2	Min	0,19	0,35	0,7	0,69	1,28	1,09	0,93	1,39	1,07	1,55
	Avg	0,39	0,78	1,19	1,53	1,95	1,98	2,09	2,23	2,11	2,18
	Avg90	0,6	1,5	1,71	2,84	3,26	3,21	3,25	3,14	3,22	3,43
	Max	0,6	1,5	1,71	2,84	3,26	3,21	3,25	3,14	3,22	3,43





**Sending speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	13,8	22,9	22,6	22,2	21,9	21,7	24,2	25,8	22,3	29,2	22,5
Total	13,8	22,9	22,6	22,2	21,9	21,7	24,2	25,8	22,3	29,2	22,5

**Receiving per user speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	125	85,9	55,5	42,8	33,2	31,9	35,6	37,9	32,8	43,0	53,0
Total	125	85,9	55,5	42,8	33,2	31,9	35,6	37,9	32,8	43,0	53,0

**Sending per user speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	4,14	2,91	1,88	1,45	1,12	1,08	1,21	1,29	1,11	1,46	1,79
Total	4,14	2,91	1,88	1,45	1,12	1,08	1,21	1,29	1,11	1,46	1,79

**Total errors %**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo2	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3
Total	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3	33,3

**URLs****Profile "Prototipo2"**

Name	Server/Port	Page	User think time
Prototipo2.page_1: http://192.168.1.3:8080/Prototipo2	http://192.168.1.3:8080	<a href="#">/Prototipo2</a>	0-0

## Reporte del Prototipo de Rich Faces

### Test execution parameters:

Test status: finished

Test started at: 24/01/2010 0:06:04

Test finished at: 24/01/2010 0:09:23

Scenario name: prototipo3.wps

Test run comment:

Test executed by: Casa

Test executed on: CASA-B6F13EF03F

Test duration: 0:03:00

Virtual users: 0 - 20

### Summary

Profile	Sessions performed	Sessions with errors	Pages performed	Pages with errors	Hits performed	Hits with errors	Total KBytes sent	Total KBytes received
Prototipo3	65	10	307	10	1.252	10	1.279	39.470
Total	65	10	307	10	1.252	10	1.279	39.470

### Number of active users

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00
Prototipo3	4	9	12	17	20	20	20	20	20	20
Total	4	9	12	17	20	20	20	20	20	20

### Sessions

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	2	0	4	6	9	11	7	8	18	65
Total	0	2	0	4	6	9	11	7	8	18	65

### Sessions per second

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0,1	0	0,2	0,3	0,45	0,55	0,7	0,4	0,9	0,36
Total	0	0,1	0	0,2	0,3	0,45	0,55	0,7	0,4	0,9	0,36

**Sessions per second**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0,1	0	0,2	0,3	0,45	0,55	0,7	0,4	0,9	0,36
Total	0	0,1	0	0,2	0,3	0,45	0,55	0,7	0,4	0,9	0,36

**Pages**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	10	18	10	22	31	40	48	20	39	59	297
Total	10	18	10	22	31	40	48	20	39	59	297

**Hits**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	24	60	38	64	108	222	202	76	172	276	1.242
Total	24	60	38	64	108	222	202	76	172	276	1.242

**Pages per second**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0,5	0,9	1	1,1	1,55	2	2,4	2	1,95	2,95	1,65
Total	0,5	0,9	1	1,1	1,55	2	2,4	2	1,95	2,95	1,65

**Hits per second**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	1,2	3	3,8	3,2	5,4	11,1	10,1	7,6	8,6	13,8	6,9
Total	1,2	3	3,8	3,2	5,4	11,1	10,1	7,6	8,6	13,8	6,9

**Pages with errors**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0	0	0	0	0	0	0	0	0	0
Total	0	0	0	0	0	0	0	0	0	0	0

**Hits with errors**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0	0	0	0	0	0	0	0	0	0
Total	0	0	0	0	0	0	0	0	0	0	0

**KBytes sent**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	14,5	43,5	27,0	44,4	82,3	181	160	58,9	134	223	968
Total	14,5	43,5	27,0	44,4	82,3	181	160	58,9	134	223	968

**KBytes received**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	438	861	528	1.005	1.499	2.781	3.096	1.227	2.497	4.013	17.945
Total	438	861	528	1.005	1.499	2.781	3.096	1.227	2.497	4.013	17.945

**Response time, sec**

Name	Time	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00
Prototipo3.page_1: http://192.168.1.3:8080/Prototipo3	Min	1,09	1,11	1,15	1,15	1,23	1,16	1,14	1,15	1,15	1,08
	Avg	1,14	1,25	1,2	2,74	3,51	1,18	1,16	1,17	1,21	1,17
	Avg90	1,18	1,34	1,27	4,99	7,54	1,23	1,21	1,2	1,52	1,26
	Max	1,18	1,34	1,27	4,99	7,54	1,23	1,21	1,2	1,52	1,26
Prototipo3.page_3: http://192.168.1.3:8080/Prototipo3/faces/pacientes.jsp	Min	2,96	3,03	2,33	2,57	7,86	1,48	2,39	2,39	1,48	1,48
	Avg	2,97	3,13	3,25	3,03	9,79	5,54	2,39	2,39	2,39	2,39
	Avg90	2,97	3,3	4,04	3,33	11,3	12,2	2,39	2,39	2,4	2,4
	Max	2,97	3,34	4,04	3,33	11,3	16,4	2,39	2,39	3,31	3,31
Prototipo3.page_4: http://192.168.1.3:8080/Prototipo3/faces/pedidos.jsp	Min	0	4,27	0	3,38	8,07	2,43	1,52	2,43	2,43	1,6
	Avg	0	4,27	0	4,39	10,4	5,32	2,43	2,43	2,43	2,44
	Avg90	0	4,28	0	5,41	12,8	11,0	2,43	2,43	2,43	2,46
	Max	0	4,28	0	5,41	13,0	13,7	3,35	2,43	2,43	3,34

**HTTP reply codes**

Code	Request	Number
Prototipo3	-	1.252
200 OK	Prototipo3.All	1.161
200 OK	Prototipo3.page_1: http://192.168.1.3:8080/Prototipo3 (1)	81
	Prototipo3.page_3: http://192.168.1.3:8080/Prototipo3/faces/pacientes.jsp (2)	640
	Prototipo3.page_4: http://192.168.1.3:8080/Prototipo3/faces/pedidos.jsp (3)	440
302 Found	Prototipo3.page_1: http://192.168.1.3:8080/Prototipo3 (1)	81
Network error	Prototipo3.page_4: http://192.168.1.3:8080/Prototipo3/faces/pedidos.jsp (3)	10

**HTTP errors %**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0	0	0	0	0	0	0	0	0	0
Total	0	0	0	0	0	0	0	0	0	0	0

**Socket errors %**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0	0	3,03	0	0	0	3,8	1,15	1,08	0,8
Total	0	0	0	3,03	0	0	0	3,8	1,15	1,08	0,8

**Timeouts %**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0	0	0	0	0	0	0	0	0	0
Total	0	0	0	0	0	0	0	0	0	0	0

**Receiving speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	304	739	961	788	1.351	2.806	2.614	1.983	2.192	3.521	1.754
Total	304	739	961	788	1.351	2.806	2.614	1.983	2.192	3.521	1.754

**Sending speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	7,67	22,9	29,0	23,3	43,0	96,2	84,4	61,9	71,0	118	56,9
Total	7,67	22,9	29,0	23,3	43,0	96,2	84,4	61,9	71,0	118	56,9

**Receiving per user speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	83,1	90,4	80,1	52,5	68,6	140	131	99,2	110	176	105
Total	83,1	90,4	80,1	52,5	68,6	140	131	99,2	110	176	105

**Sending per user speed (kbits/sec)**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	1,96	2,78	2,41	1,56	2,18	4,81	4,22	3,09	3,55	5,89	3,3
Total	1,96	2,78	2,41	1,56	2,18	4,81	4,22	3,09	3,55	5,89	3,3

**Total errors %**

Profile	0:00:00-0:00:18	0:00:18-0:00:36	0:00:36-0:00:54	0:00:54-0:01:12	0:01:12-0:01:30	0:01:30-0:01:48	0:01:48-0:02:06	0:02:06-0:02:24	0:02:24-0:02:42	0:02:42-0:03:00	Total
Prototipo3	0	0	0	3,03	0	0	0	3,8	1,15	1,08	0,8
Total	0	0	0	3,03	0	0	0	3,8	1,15	1,08	0,8

**URLs****Profile "Prototipo3"**

Name	Server/Port	Page	User think time
Prototipo3.page_1: http://192.168.1.3:8080/Prototipo3	http://192.168.1.3:8080	<a href="#">/Prototipo3</a>	0-0
Prototipo3.page_3: http://192.168.1.3:8080/Prototipo3/faces/pacientes.jsp	http://192.168.1.3:8080	<a href="#">/Prototipo3/faces/pacientes.jsp</a>	0-43
Prototipo3.page_4: http://192.168.1.3:8080/Prototipo3/faces/pedidos.jsp	http://192.168.1.3:8080	<a href="#">/Prototipo3/faces/pedidos.jsp</a>	0-0

### Reporte de Performance Monitor para el Prototipo de Adobe Flex

Hora	Paquetes recibidos	Paquetes/s."	Total de bytes/s
01/24/2010 00:58:16.859	19.050.713.688.575.500	3.229.435.966.452.760	26.650
01/24/2010 00:58:20.859	3.325.389.261.787.910	53.506.263.309.970.900	45.572
01/24/2010 00:58:24.859	48.417.008.238.579.700	74.372.517.809.777.100	68.426
01/24/2010 00:58:28.859	5.334.676.081.751.910	82.399.456.849.595.200	76.024
01/24/2010 00:58:32.859	22.427.112.772.352.600	59.805.634.059.607.000	3.141
01/24/2010 00:58:36.859	19.985.128.785.172.200	39.970.257.570.344.400	1.950
01/24/2010 00:58:40.859	10.028.028.687.632.800	15.042.043.031.449.200	544
01/24/2010 00:58:44.859	66.598.595.866.615.900	10.575.956.796.795.900	93.067
01/24/2010 00:58:48.859	4.838.789.539.351.490	74.712.916.203.458.200	68.709
01/24/2010 00:58:52.859	24.736.223.275.942.200	39.478.012.905.039.100	34.837
01/24/2010 00:58:56.859	0.99968382228967001	12.496.047.778.620.800	529
01/24/2010 00:59:00.859	36.211.904.228.092.900	58.188.784.035.487.300	50.781
01/24/2010 00:59:04.859	52.944.341.947.330.400	82.163.625.003.168.400	76.057
01/24/2010 00:59:08.859	49.056.213.870.867.400	77.088.336.082.791.600	69.296
01/24/2010 00:59:12.859	10.009.035.627.466.200	25.022.589.068.665.500	1.132
01/24/2010 00:59:16.859	0.249816494787477	0.999265979149908	61.855
01/24/2010 00:59:20.859	0.74824592718500538	12.470.765.453.083.400	498
01/24/2010 00:59:28.859	75.122.736.948.190.300	14.523.729.143.316.700	10.110
01/24/2010 00:59:32.859	66.451.436.132.000.000	1.024.251.459.177.440	92.681
01/24/2010 00:59:36.859	61.835.639.644.614.400	95.632.446.737.824.800	85.851
01/24/2010 00:59:40.859	59.961.577.667.352.700	11.242.795.812.628.600	9.401
01/24/2010 00:59:44.859	17.491.527.350.603.200	37.481.844.322.721.100	2.352
01/24/2010 00:59:48.859	0.99863797470205018	17.476.164.557.285.800	1.100
01/24/2010 00:59:52.859	54.562.655.924.241.200	87.850.881.786.278.400	76.177

01/24/2010 00:59:56.859	52.470.385.489.942.800	80.954.309.041.626.100	74.507
01/24/2010 01:00:00.859	3.197.420.670.507.290	51.208.690.426.093.300	45.854
01/24/2010 01:00:04.859	0.75049453225876106	17.511.539.086.037.700	1.090
01/24/2010 01:00:08.859	0.99968262426730947	1.499.523.936.400.960	51.059
01/24/2010 01:00:12.859	37.051.528.439.650.000	6.008.355.963.186.480	51.007
01/24/2010 01:00:16.859	53.969.545.901.373.400	83.952.626.957.691.900	76.918
01/24/2010 01:00:20.859	47.711.572.173.777.300	74.440.044.543.380.400	67.932
01/24/2010 01:00:24.859	0.75146435114815013	22.543.930.534.444.500	1.118
01/24/2010 01:00:28.859	0.74879545156705851	17.471.893.869.898.000	1.088
01/24/2010 01:00:32.859	0.24963301120184639	0.49926602240369278	28
01/24/2010 01:00:36.859	67.605.854.516.289.000	10.766.858.311.853.400	93.843
01/24/2010 01:00:40.859	60.201.441.400.607.700	93.924.240.525.429.500	85.644
01/24/2010 01:00:44.859	13.237.012.451.491.400	22.727.700.624.258.900	17.969
01/24/2010 01:00:48.859	12.534.920.919.061.200	30.083.810.205.746.900	352
01/24/2010 01:00:52.859	35.721.054.754.681.400	58.952.230.224.509.200	48.815
01/24/2010 01:00:56.859	44.996.825.138.493.100	71.244.973.135.947.500	62.806
01/24/2010 01:01:00.859	56.537.243.872.609.300	88.308.172.951.465.000	77.763
01/24/2010 01:01:04.859	12.216.267.420.129.500	21.191.484.300.224.800	18.064
01/24/2010 01:01:08.859	3.498.878.407.152.790	57.481.573.831.795.900	4.066
01/24/2010 01:01:12.859	42.621.811.167.764.400	75.214.960.884.290.200	5.969
01/24/2010 01:01:16.859	0.2494335739061441	0.49886714781228819	28
01/24/2010 01:01:20.859	32.521.389.173.633.700	52.034.222.677.814.000	45.404
01/24/2010 01:01:24.859	49.411.958.870.954.500	77.112.602.480.429.000	68.912
01/24/2010 01:01:28.859	55.143.975.738.441.800	86.726.434.570.458.500	76.890
01/24/2010 01:01:32.859	3.250.549.952.876.070	75.012.691.220.217.100	3.713

01/24/2010 01:01:36.859	44.996.681.942.865.600	99.992.626.539.701.500	5.532
01/24/2010 01:01:40.859	12.740.577.480.861.300	21.733.926.290.881.100	14.160
01/24/2010 01:01:44.859	0.24984281177169534	0.49968562354339069	28
01/24/2010 01:01:48.859	66.430.041.794.600.400	10.588.848.767.259.600	93.513
01/24/2010 01:01:52.859	47.996.506.753.532.500	7.499.454.180.239.460	68.505
01/24/2010 01:01:56.859	24.510.136.757.978.800	39.516.342.936.333.200	35.280
01/24/2010 01:02:00.859	15.028.266.259.486.500	30.056.532.518.973.000	1.751
01/24/2010 01:02:08.859	0.74903073468570625	12.483.845.578.095.100	69
01/24/2010 01:02:12.859	0.24992049897502694	0.49984099795005388	28
01/24/2010 01:02:16.859	0.25016414064288189	0.50032828128576379	29
01/24/2010 01:02:28.859	0.50045099180377195	10.009.019.836.075.400	57
01/24/2010 01:02:36.859	0.49935378696648475	0.7490306804497272	40
01/24/2010 01:02:44.859	37.208.307.895.708.700	60.682.005.494.343.800	50.923
01/24/2010 01:02:48.859	53.654.087.997.524.000	83.101.447.921.746.500	76.419
01/24/2010 01:02:52.859	4.825.816.317.664.360	76.012.858.060.620.000	68.679
01/24/2010 01:02:56.859	2.250.939.956.996	57.524.021.123.231.100	2.883
01/24/2010 01:03:00.859	0.50118736153637955	0.75178104230456932	471
01/24/2010 01:03:04.859	0.24951145316437939	0.49902290632875879	28
01/24/2010 01:03:08.859	0.50066174121971541	0.75099261182957311	41
01/24/2010 01:03:12.859	0.74921171712464008	19.978.979.123.323.700	675
01/24/2010 01:03:16.859	50.024.167.598.322.300	8.754.229.329.706.410	64.656

Total de Bytes por cada solicitud, eliminando valores extraños

**KBPS por solicitud**

**Hora**

**26,65048889** 01/24/2010 00:58:16.859

**11,39308365** 01/24/2010 00:58:20.859

**17,10642781** 01/24/2010 00:58:24.859

---

19,00603581	01/24/2010 00:58:28.859
3,141041739	01/24/2010 00:58:32.859
1,949799127	01/24/2010 00:58:36.859
0,544020556	01/24/2010 00:58:40.859
23,2667308	01/24/2010 00:58:44.859
17,17726411	01/24/2010 00:58:48.859
8,709336875	01/24/2010 00:58:52.859
0,528832742	01/24/2010 00:58:56.859
12,69520684	01/24/2010 00:59:00.859
19,01413628	01/24/2010 00:59:04.859
17,3239141	01/24/2010 00:59:08.859
1,132272155	01/24/2010 00:59:12.859
15,46364103	01/24/2010 00:59:16.859
0,497583542	01/24/2010 00:59:20.859
10,11001794	01/24/2010 00:59:28.859
23,17025427	01/24/2010 00:59:32.859
21,46272493	01/24/2010 00:59:36.859
9,400726179	01/24/2010 00:59:40.859
2,351610913	01/24/2010 00:59:44.859
1,100499048	01/24/2010 00:59:48.859
19,0441815	01/24/2010 00:59:52.859
18,62673699	01/24/2010 00:59:56.859
11,46344005	01/24/2010 01:00:00.859
1,089968226	01/24/2010 01:00:04.859
12,76469751	01/24/2010 01:00:08.859
12,75167155	01/24/2010 01:00:12.859
19,22939918	01/24/2010 01:00:16.859
16,98288416	01/24/2010 01:00:20.859
1,118429443	01/24/2010 01:00:24.859
1,087500594	01/24/2010 01:00:28.859
0,028458163	01/24/2010 01:00:32.859
23,46067127	01/24/2010 01:00:36.859
21,41097986	01/24/2010 01:00:40.859
17,96862002	01/24/2010 01:00:44.859
0,351729881	01/24/2010 01:00:48.859
12,20373615	01/24/2010 01:00:52.859
15,70145465	01/24/2010 01:00:56.859
19,44068154	01/24/2010 01:01:00.859
18,06362122	01/24/2010 01:01:04.859
4,066196549	01/24/2010 01:01:08.859
5,969310012	01/24/2010 01:01:12.859
0,028435427	01/24/2010 01:01:16.859
11,35109056	01/24/2010 01:01:20.859
17,22805344	01/24/2010 01:01:24.859
19,22256331	01/24/2010 01:01:28.859

---

3,712878173	01/24/2010 01:01:32.859
5,532092063	01/24/2010 01:01:36.859
14,1600277	01/24/2010 01:01:40.859
0,028482081	01/24/2010 01:01:44.859
23,37831754	01/24/2010 01:01:48.859
17,12625353	01/24/2010 01:01:52.859
8,819960207	01/24/2010 01:01:56.859
1,750542548	01/24/2010 01:02:00.859
0,068910828	01/24/2010 01:02:08.859
0,028490937	01/24/2010 01:02:12.859
0,028518712	01/24/2010 01:02:16.859
0,057051413	01/24/2010 01:02:28.859
0,040447657	01/24/2010 01:02:36.859
12,73079757	01/24/2010 01:02:44.859
19,10472341	01/24/2010 01:02:48.859
17,16977938	01/24/2010 01:02:52.859
2,882703772	01/24/2010 01:02:56.859
0,471366714	01/24/2010 01:03:00.859
0,028444306	01/24/2010 01:03:04.859
0,040553601	01/24/2010 01:03:08.859
0,675289494	01/24/2010 01:03:12.859
16,16405916	01/24/2010 01:03:16.859

### Medidas de Tiempo de Inicialización

Para la toma de estas medidas se utilizaron 4 clientes simultáneos.

Tiempo de Inicialización
10
14
12
10
8
7
3
8
14
15
11
8
9
12
17

---

12
10
8
2
10
8
12
10
3
10
14
17
10
8
12
10
10,12903226

---