



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

**ANÁLISIS COMPARATIVO ENTRE MODELOS DE HOSTING PARA CLOUD
COMPUTING, CASO PRÁCTICO: SISTEMA DE GESTOR DE PROYECTOS
EN LÍNEA**

TESIS DE GRADO

**Previa la obtención del título de
INGENIERO EN SISTEMAS INFORMÁTICO**

Presentado por:

**ANA CRISTINA GUAÑO ÁLVAREZ
MARÍA GABRIELA ALTAMIRANO HERNÁNDEZ**

RIOBAMBA – ECUADOR

2011

ÍNDICE DE ABREVIATURAS

AMI:	Amazon Machine Image
API:	Application programming interface
APP:	Application software
AWS:	Amazon Web Service
EC2:	Elastic Compute Cloud
HTML:	hypertext markup language.
HTTP:	hypertext transport protocol.
IAAS:	Infrastructure as a Service
PAAS:	Platform as a Service
PDA:	personal digital assistant.
PDF:	portable document format.
SAAS:	Software as a Service
TI:	Tecnologías de la Información
URL:	Uniform Resource Locator
WWW:	World Wide Web.
XAAS:	Anything as a service
XML:	Xtensible Markup Language.

ÍNDICE

INTRODUCCIÓN	19
CAPITULO I	21
MARCO REFERENCIAL.....	21
1.1. TÍTULO DEL PROYECTO DE TESIS:.....	21
1.2. LUGAR DE REALIZACIÓN.....	21
1.3. ANTECEDENTES.....	21
1.4. JUSTIFICACIÓN.....	24
1.4.1. Justificación Teórica.....	24
1.4.2. Justificación Aplicativa	25
1.5. OBJETIVOS.....	26
1.5.1. Objetivo General	26
1.5.2. Objetivos Específicos.....	26
1.6. HIPÓTESIS.....	26
CAPITULO II	27
MARCO TEÓRICO	27
2.1. LOS CIMIENTOS DE CLOUD COMPUTING.....	27
2.1.1. Qué es Cloud Computing.....	27
2.1.2. Evolución de Cloud Computing.....	27
2.1.3. Cloud Computing y Xaas (Saas Iaas, Paas, Etc.).....	28
2.1.3.1. Software as a Service (SaaS).....	28
2.1.3.2. Infrastructure as a Service (IaaS).....	29
2.1.3.3. Platform as a Service (PaaS).....	29
2.1.4. Características de los principales servicios.....	29
2.1.4.1. Agilidad	29
2.1.4.2. Auto Reparable	30
2.1.4.3. Costo.....	30
2.1.4.4. Independencia de ubicación	30
2.1.4.5. Multi-arrendamiento	30
2.1.4.6. Fiabilidad.....	30
2.1.4.7. Escalabilidad.....	31
2.1.4.8. Seguridad.....	31
2.1.4.9. Mantenimiento	31
2.1.4.10. Medición.....	31
2.1.5. Aspectos Técnicos.....	31

2.1.5.1. Plataformas de Desarrollo.....	31
2.1.5.1.1. Windows Server	32
2.1.5.1.2. Linux.....	33
2.1.5.2. Principales Modelos de hosting Cloud Computing.....	35
2.1.5.2.1. Amazon Web Services	35
2.1.5.2.2. Google App Engine	36
2.1.5.2.3. Microsoft Azure.....	36
2.1.5.2.4. Force.Com.....	36
CAPITULO III	37
ANÁLISIS COMPARATIVO.....	37
3.1. INTRODUCCIÓN	37
3.2. DETERMINAR LAS MODELOS DE HOSTING A COMPARAR	37
3.3. ANÁLISIS DE LOS MODELOS SELECCIONADOS	39
3.3.1. Amazon Ec2.....	39
3.3.1.1. Funcionalidad.....	39
3.3.1.2. Características	40
3.3.1.3. Restricciones	42
3.3.1.4. Instancias / Cuotas	43
3.3.1.4.1. Norma de instancias.....	43
3.3.1.4.2. Instancias con Gran Capacidad de Memoria.....	44
3.3.1.4.3. Instancias con gran capacidad de CPU.....	44
3.3.1.5. Sistema Operativo	45
3.3.1.6. Software.....	45
3.3.1.7. Precios.....	46
3.3.1.7.1. Instancias bajo demanda.....	46
3.3.1.7.2. Instancias Reservadas	47
3.3.1.7.3. Instancias Spot.....	49
3.3.1.7.4. Transferencia de datos.....	49
3.3.1.7.5. Disponibilidad de transferencia de datos por zona	50
3.3.1.7.6. Transferencia de datos Regionales.....	50
3.3.1.7.7. Amazon Elastic Block Store	51
3.3.1.7.8. Direcciones IP elásticas	51
3.3.1.7.9. Amazon CloudWatch.....	51
3.3.1.7.10. Auto Escalabilidad.....	51
3.3.1.7.11. Equilibrio de carga elástica	52
3.3.2. Google App Engine.....	52

3.3.2.1. Funcionalidad.....	53
3.3.2.2. Características	53
3.3.2.3. Restricciones	55
3.3.2.4. Instancias / Cuotas	56
3.3.2.4.1. Cuotas Facturables y Fijas	56
3.3.2.4.2. Renovación de recursos.....	57
3.3.2.4.3. Cuotas por minuto	57
3.3.2.4.4. Reducción de un recurso.....	58
3.3.2.4.5. Recursos	59
3.3.2.5. Sistema Operativo	69
3.3.2.6. Software.....	69
3.3.2.7. Precios.....	70
3.4. DETERMINACIÓN DE LOS PARÁMETROS DE COMPARACIÓN....	70
3.4.1. Lenguajes de Programación	71
3.4.2. Sistema Operativo	73
3.4.3. Bases de Datos.....	73
3.4.4. Ambientes de Desarrollo.....	74
3.4.4.1. Eclipse	74
3.4.4.1.1. Características.....	74
3.4.4.1.2. Plugins.....	75
3.4.4.2. Netbeans.....	79
3.4.4.2.1. Características.....	79
3.4.4.2.2. Plugins.....	80
3.4.4.3. Comparación entre Eclipse y Netbeans.....	81
3.4.5. Servidor de Aplicaciones.....	82
3.4.6. Costo.....	83
3.4.7. Configuración.....	84
3.4.7.1. Amazon EC2.....	84
Ingresar a la página oficial de Amazon EC2 http://aws.amazon.com/	84
Se crea una cuenta en Amazon Web Services.....	85
En la siguiente pantalla se muestran detalladamente los costos por la creación de instancias o proyectos. En la misma se debe ingresar la forma de pago....	88
3.4.7.2. Google App Engine	96
Ingresar a la página oficial de google http://www.google.com	96
3.5. Análisis Comparativo	100
3.5.1. Lenguajes de Programación	102

3.5.1.1. Determinación de Variables	102
3.5.1.2. Valoraciones	102
3.5.1.3. Interpretación	102
3.5.1.4. Calificación.....	103
3.5.2. Sistema Operativo	103
3.5.2.1. Determinación de Variables	103
3.5.2.2. Valoraciones	103
3.5.2.3. Interpretación	104
3.5.2.4. Calificación.....	104
3.5.3. Bases de Datos.....	105
3.5.3.1. Determinación de Variables	105
3.5.3.2. Valoraciones	105
3.5.3.3. Interpretación	105
3.5.3.4. Calificación.....	106
3.5.4. Ambientes de Desarrollo.....	106
3.5.4.1. Determinación de Variables	106
3.5.4.2. Valoraciones	106
3.5.4.3. Interpretación	107
3.5.4.4. Calificación.....	107
3.5.5. Servidor de Aplicaciones.....	107
3.5.5.1. Determinación de Variables	108
3.5.5.2. Valoraciones	108
3.5.5.3. Interpretación	108
3.5.5.4. Calificación.....	109
3.5.6. Costos.....	109
3.5.6.1. Determinación de Variables	109
3.5.6.2. Valoraciones	109
3.5.6.3. Interpretación	111
3.5.6.4. Calificación.....	111
3.5.7. Configuración.....	112
3.5.7.1. Determinación de Variables	112
3.5.7.2. Valoraciones	112
3.5.7.3. Interpretación	113
3.5.7.4. Calificación.....	113
3.6. RESUMEN COMPARATIVO.....	114
3.7. COMPROBACIÓN DE LA HIPÓTESIS.....	114

CAPITULO IV	129
DESARROLLO DE LA APLICACIÓN WEB, GESTOR DE PROYECTOS EN LÍNEA.....	129
4.1. PARTE TEÓRICA	129
4.1.1. Introducción	129
4.1.2. Desarrollo.....	129
4.1.2.1. Herramientas utilizadas.....	134
4.2. PARTE PRÁCTICA.....	135
4.2.1. Introducción	135
4.2.2. Desarrollo.....	136
4.2.2.1. Fase 1: Definición del Ámbito	136
4.2.2.1.1 Equipo de Trabajo	136
4.2.2.1.2 Servicios.....	136
4.2.2.1.3 Funciones.....	137
4.2.2.1.4 Descripción del Sistema Actual	137
4.2.2.1.5 Identificación del Problema	138
4.2.2.2. Fase 2: Modelamiento del negocio	138
4.2.2.2.1 Definir los límites del proceso.....	138
Descripción de los Procesos.....	138
4.2.2.3. Fase 3: Estudio de Factibilidad.....	148
4.2.2.3.1 Factibilidad Operativa.....	148
4.2.2.3.2 Factibilidad Técnica.....	149
4.2.2.3.3 Factibilidad Económica.....	150
4.2.2.4. Fase 4: Exploración	150
4.2.2.4.1 Historia de Usuario	150
Riesgos.....	152
4.2.2.5. Fase 5: Iteración	157
4.2.2.5.4 Análisis de resultados.....	158
4.2.2.6. Fase 6: Diseño.....	158
CONCLUSIONES	163
RECOMENDACIONES	165
RESUMEN.....	166
Anexo 1. Requerimientos.....	169
Anexo 2. Diagramas de Caso de Uso	171
Anexo 3. Encuestas Google App Engine y Amazon EC2	173
GLOSARIO	179

BIBLIOGRAFÍA..... 182

ÍNDICE DE FIGURAS

Figura III - 1	Página Oficial de Amazon EC2.....	84
Figura III - 2	Create an Amazon Web Service Account. Amazon Web Service Sing In.....	85
Figura III - 3	Create an Amazon Web Service Account. Registration.....	85
Figura III - 4	Formulario de datos y Contrato de cliente de AWS	86
Figura III - 5	Envío de Notificación al e-mail.....	86
Figura III - 6	Acceso a e-mail	87
Figura III - 7	Mail de confirmación de la cuenta.....	87
Figura III - 8	Amazon Web Service Sing In	88
Figura III - 9	Información de Costos en Amazon.....	88
Figura III - 10	Consola de Gestión de AWS	89
Figura III - 11	Lanzamiento de Nueva Instancia.....	89
Figura III - 12	Requisitos de la Instancia. Choose an AMI	90
Figura III - 13	Cummunity AMIs.....	90
Figura III - 14	Requisitos de la Instancia. Instance Details.....	90
Figura III - 15	Requisitos de la Instancia. Create Key Pair	91
Figura III - 16	Requisitos de la Instancia. Configure Firewall	92
Figura III - 17	Requisitos de la Instancia. Review	92
Figura III - 18	Navigation. Security Groups	93
Figura III - 19	Security Groups.....	93
Figura III - 20	PuTTY Key Generator.....	94
Figura III - 21	PuTTY Configuration.....	94
Figura III - 22	Ejecución de Comandos	95
Figura III - 23	Página Oficial de Google	96
Figura III - 24	Creación de una nueva cuenta Gmail.....	96
Figura III - 25	Aceptación de la nueva cuenta.....	97
Figura III - 26	Welcome to Google App Engine. Crear Aplicación.....	97
Figura III - 27	Verify Your Account by SMS.....	98
Figura III - 28	An Authentication Code Has Been Sent to... ..	98
Figura III - 29	Create Application.....	98

Figura III - 30 Application Registered Successfully	99
Figura III - 31 My Aplications.....	99
Figura III - 32 Panel de Administración	100
Figura III - 33 Interpretación del parámetro - Lenguaje de Programación.....	103
Figura III - 34 Interpretación del parámetro - Sistema Operativo.	104
Figura III - 35 Interpretación del parámetro - Base de Datos.	106
Figura III - 36 Interpretación del parámetro - Ambientes de Desarrollo.....	107
Figura III - 37 Interpretación del parámetro - Servidor de Aplicaciones.	109
Figura III - 38 Interpretación del parámetro - Costos.....	112
Figura III - 39 Interpretación de Parámetros - Configuración	114
Figura III - 40 Encuesta Google App Engine - Primera Pregunta.....	116
Figura III - 41 Encuesta Google App Engine - Segunda Pregunta.....	117
Figura III - 42 Encuesta Google App Engine - Tercera Pregunta.....	117
Figura III - 43 Encuesta Google App Engine - Cuarta Pregunta	117
Figura III - 44 Encuesta Google App Engine - Quinta Pregunta.....	118
Figura III - 45 Encuesta Google App Engine - Sexta Pregunta	118
Figura III - 46 Encuesta Google App Engine - Séptima Pregunta	118
Figura III - 47 Encuesta Google App Engine - Octava Pregunta	119
Figura III - 48 Encuesta Google App Engine - Novena Pregunta	119
Figura III - 49 Encuesta Google App Engine - Décima Pregunta	119
Figura III - 50 Encuesta Amazon EC2 - Primera Pregunta.....	121
Figura III - 51 Encuesta Amazon EC2 - Segunda Pregunta.....	121
Figura III - 52 Encuesta Amazon EC2 - Tercera Pregunta.....	121
Figura III - 53 Encuesta Amazon EC2 - Cuarta Pregunta	122
Figura III - 54 Encuesta Amazon EC2 - Quinta Pregunta.....	122
Figura III - 55 Encuesta Amazon EC2 - Sexta Pregunta.....	122
Figura III - 56 Encuesta Amazon EC2 - Séptima Pregunta	123
Figura III - 57 Encuesta Amazon EC2 - Octava Pregunta.....	123
Figura III - 58 Encuesta Amazon EC2 - Novena Pregunta.....	123
Figura III - 59 Encuesta Amazon EC2 - Décima Pregunta	124
Figura III - 60 Diagrama de Chi cuadrado	128
Figura IV - I Metodología XP.....	130

Figura IV - 2 Prototipo	157
Figura IV - 3 Página Principal.....	159
Figura IV - 4 Arquitectura Lógica.....	160
Figura IV - 5. Diagrama de componentes	161
Figura IV - 6 Arquitectura física de la aplicación	161
Figura IV - 7. Diagrama de Base de Datos	162

ÍNDICE DE TABLAS

Tabla III - I Norma de Instancias	43
Tabla III - II Instancias con Gran Capacidad de Memoria	44
Tabla III - III Instancias con gran capacidad de CPU	45
Tabla III - IV Sistema Operativo	45
Tabla III - V Base de Datos disponibles en Amazon EC2	46
Tabla III - VI Entornos de Desarrollo en Amazon EC2	46
Tabla III - VII Precios en Amazon EC2.....	47
Tabla III - VIII Precio de Instancias reservadas en Amazon EC2.....	48
Tabla III - IX Precio de Instancias Spot en Amazon EC2	49
Tabla III - X Precio por Transferencia de Datos en Amazon EC2	50
Tabla III - XI Precio de Amazon Elastic Block Store.....	51
Tabla III - XII Ancho de banda de entrada seguro Google App Engine.....	62
Tabla III - XIII Tiempo de CPU en el almacenamiento de datos de Google App Engine	64
Tabla III - XIV Datos de archivo adjunto enviados en Google App Engine	66
Tabla III - XV Datos de extracción de URL recibidos en Google App Engine ..	67
Tabla III - XVI Transformaciones ejecutadas de Imagenes en Google App Engine.....	68
Tabla III - XVII Datos recibidos del API en Google App Engine	69
Tabla III - XVIII Sistema Operativo de Google App Engine	69
Tabla III - XIX Software Google App Engine	70
Tabla III - XX Precios Google App Engine	70
Tabla III - XXI Comparación entre Python y Java	72
Tabla III - XXII Comparación entre Eclipse y Netbeans.	81
Tabla III - XXIII Escala Cualitativa	100
Tabla III - XXIV Escala Cuantitativa	101
Tabla III - XXV Variedad de Lenguajes de Programación.....	102
Tabla III - XXVI Variedad de Sistemas Operativos.....	104
Tabla III - XXVII Variedad de Base de Datos	105
Tabla III - XXVIII Variedad de Ambientes de Desarrollo	107

Tabla III - XXIX Variedad de Servidor de Aplicaciones	108
Tabla III - XXX Valoración de Costos.....	110
Tabla III - XXXI Valoración de Configuración.....	113
Tabla III - XXXII Resumen Comparativo	114
Tabla III - XXXIII Presentación de resultados de la Encuesta de Google App Engine.....	115
Tabla III - XXXIV Presentación de resultados de la Encuesta de Amazon EC2	120
Tabla III - XXXV Indicadores.....	125
Tabla III - XXXVI Datos Observados.....	125
Tabla III - XXXVII Datos Esperados.....	126
Tabla III - XXXVIII Cálculo de Chi Cuadrado.....	126
Tabla IV - I Caso de Uso Autenticar.....	138
Tabla IV - II Caso de Uso Cerrar Sesión.....	139
Tabla IV - III Caso de Uso Agregar Proyecto	140
Tabla IV - IV Caso de Uso Agregar Usuarios.....	141
Tabla IV - V Caso de Uso Agregar Requerimiento.....	142
Tabla IV - VI Caso de Uso Agregar Tareas.....	143
Tabla IV - VII Caso de Uso Asignar Tareas	144
Tabla IV - VIII Caso de Uso Subir Tarea.....	145
Tabla IV - IX Caso de Uso Aprobar Tarea	146
Tabla IV - X Caso de Uso Aprobar Tarea	147
Tabla IV - XI Proceso Generar Reportes	148
Tabla IV - XII Costos Tangibles del proyecto	150
Tabla IV - XIII Historia de Usuario 1	151
Tabla IV - XIV Historia de Usuario 2	151
Tabla IV - XV Historia de Usuario 3	152
Tabla IV - XVI Historia de Usuario 4	152
Tabla IV - XVII Historia de Usuario 5	152
Tabla IV - XVIII Identificación de Riesgos	153
Tabla IV - XIX Criterios de valoración de la probabilidad de riesgos.....	153
Tabla IV - XX Criterio de valoración del impacto de riesgos	153

Tabla IV - XXI Determinación de exposición al riesgo	154
Tabla IV - XXII Valoración de determinación de exposición al riesgo	154
Tabla IV - XXIII Impacto Esperados	154
Tabla IV - XXIV Priorización de los Riesgos	154
Tabla IV - XXV Hoja de Gestión de Riesgo 1	155
Tabla IV - XXVI Hoja de Gestión de Riesgo 2.....	156

Mi tesis la dedico con todo cariño:

A Dios que me dio la oportunidad de vivir y de regalarme una familia maravillosa.

A mis padres Fernando y Elizabeth que me dieron la vida y siempre me han acompañado en todo momento, Gracias a mis padres por su entrega, dedicación y por inculcarme a seguir mis metas y una de las más importantes es de culminar mi carrera profesional.

A mi hermana Mary que me apoyado siempre en todas mis decisiones, gracias por tus consejos, por tu paciencia y por alegrarme los días. Te quiero mucho.

A todos mis amigos, muchas gracias por estar conmigo en todo este tiempo, donde hemos pasado momentos felices y tristes en nuestra vida politécnica. Siempre les llevare en mi corazón.

Anita Guaño A.

Agradezco a Dios por permitirme llegar hasta donde estoy y por bendecirme con el don de la vida y con una familia tan maravillosa.

Dedico mi tesis a mis padres José y Olga por su infinito e incondicional amor. Gracias Mamita por tu ayuda, tu paciencia, por siempre estar a mi lado y escucharme cuando lo he necesitado. Papito gracias por tu entrega y dedicación para brindarme siempre lo mejor. Gracias a los dos por su admirable abnegación, porque siempre han estado a mi lado apoyándome en los buenos y malos momentos de mi vida, por inculcarme el valor de la vida, de la verdad, el respeto y la fortaleza para siempre luchar por mis propósitos. Este logro se los debo a Ustedes.

A mi hija Mikaela que ha llegado a ser el motor de mi vida y la razón para seguir adelante.

A mi esposo por su apoyo incondicional, por creer en mí y por darme fuerzas para seguir luchando por nuestra familia.

A mis hermanos Vero, Mauri y Cesar; a mis sobrinitas Nicole y Camila, por su motivación, paciencia y apoyo incondicional; y a mis amigos por su amistad sincera.

A todos mis profesores que colaboraron en mi formación académica. Sus lecciones permanecerán siempre conmigo.

Gabriela Altamirano H.

NOMBRE

FIRMA

FECHA

Ing. Iván Menes

**DECANO DE LA FACULTAD
DE INFORMÁTICA Y
ELECTRÓNICA**

.....

.....

Ing. Raúl Rosero

**DIRECTOR DE LA
ESCUELA DE INGENIERÍA
EN SISTEMAS.**

.....

.....

Ing. Wladimir Castro

DIRECTOR DE TESIS

.....

.....

Ing. Gloria Arcos

MIEMBRO DEL TRIBUNAL

.....

.....

Lcdo. Carlos Rodríguez

**DIRECTOR DPTO.
DOCUMENTACIÓN**

.....

.....

NOTA DE LA TESIS

.....

“Nosotras, **Ana Cristina Guaño Alvarez y María Gabriela Altamirano Hernández** somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”

.....

.....

Ana Cristina Guaño Alvarez

María Gabriela Altamirano Hernández

INTRODUCCIÓN

Prácticamente desde el inicio de la computación existe el modelo de trabajo en red donde "terminales informales" establecen conexión con un servidor de trabajo que es donde todas las aplicaciones y recursos se encuentran.

Ahora bien, el modelo ha ido evolucionando hasta poder utilizar Internet como medio para acceder a múltiples recursos y aplicaciones web desde cualquier ordenador usando únicamente un navegador Web.

Esa Red es la nube, los terminales que se conectan son los clientes y el trabajo se realiza a través de uno o más servidores de trabajo.

Este concepto es cada vez más conocido, ya que se trata de un modelo de computación que permite a las organizaciones ahorrarse inversiones en aplicaciones, sistemas e incluso personal.

La idea principal es sencilla: que los clientes puedan acceder a recursos software y hardware ofrecidos como servicios a través de Internet. Así pues, es en la Red es donde queda almacenada toda nuestra información.

Estos modelos de computación presentan muchas ventajas en cuanto a ahorro de costes, pero también introducen nuevos riesgos en el funcionamiento cotidiano de las organizaciones, especialmente en lo que se refiere a la seguridad de la información.

Los sistemas basados en modelos de hosting están ayudando paulatinamente a que esta necesidad sea satisfecha. Estos modelos permiten independizarnos del hardware de tal forma que podemos disponer de un conjunto de servidores físicos que ofrezcan capacidad de cómputo y almacenamiento, y crear

servidores virtuales que funcionen sobre ellas y varíen su tamaño o su número a lo largo del tiempo, conforme las necesidades cambien.

En el primer capítulo se hizo una breve descripción de la información general del presente proyecto de tesis, en el segundo capítulo está todo el fundamento teórico necesario para el estudio, el tercer capítulo es el más importante ya que contiene la información de los modelos de hosting de cloud computing a comparar, los parámetros de medición, el análisis y la comprobación de la hipótesis. El cuarto capítulo tiene el desarrollo del Sistema Gestor de Proyectos en línea.

CAPITULO I

MARCO REFERENCIAL

1.1. TÍTULO DEL PROYECTO DE TESIS:

ANÁLISIS COMPARATIVO ENTRE MODELOS DE HOSTING PARA CLOUD COMPUTING, CASO PRÁCTICO: SISTEMA DE GESTOR DE PROYECTOS EN LÍNEA.

1.2. LUGAR DE REALIZACIÓN

Publicación en la Web, accesible a todo público.

1.3. ANTECEDENTES

Desde ya algún tiempo viene hablándose con creciente insistencia sobre “la internet de las nubes”, donde están flotando nuestros datos, correos electrónicos, servicios, aplicaciones, etcétera. Como desde sus inicios, no deja de ser una de las tantas metáforas que se utilizan para representar a Internet.

Por eso nos interesa desde Tendencias Digitales hacer un breve y simple recorrido por los factores claves de este nuevo momento que hace converger nuevos servicios a nivel de conectividad y software con cambios culturales a la

hora de nuestra actividad en la red. Este fenómeno se conoce hoy como "Cloud Computing".

A fines prácticos se puede dividir los servicios de "la nube" en tres grandes categorías: en tanto oferta de infraestructura (hardware y procesamiento); software (aplicaciones) y entornos de desarrollo (Application Engines). De cualquier manera, todas estas variantes comparte una única fuente: la de contar con recursos (ya sea infraestructura, software, o plataformas) de manera remota y deslocalizada a través de una conexión a Internet.

Esa idea no es nueva, se viene discutiendo en el medio desde hace muchos años con diferentes nombres que tal vez ahora queden claros para el lector: "Utility Computing", computación en demanda, computación elástica, "Grid Computing".

Una cosa es tener clara la idea y otra llevarla a la práctica de forma exitosa, por ello y a pesar de muchos casos aislados medianamente satisfactorios, hasta ahora no se había cruzado el nivel de madurez necesario para ser ampliamente aceptado y usado por el mercado, sin embargo cada día esto evoluciona a mayor velocidad y es claro que ello va a ocurrir con todas sus consecuencias más temprano que tarde.

Viniendo de un pasado construido en torno a aplicaciones instaladas en ordenadores de sobremesa, brutalmente sobredimensionadas y poner en manos de cada empleado una licencia con un coste de varios cientos de dólares para que pueda maquetar documentos de la manera más sofisticada posible, cuando lo único que necesitamos es que escriba y comparta fácilmente lo que escribe con quienes trabajan con él. Las empresas que prueban aplicaciones con los modelos de Cloud Computing se encuentran de repente con niveles de productividad y satisfacción sorprendentes, y con esquemas de trabajo que pasan a tener mucha más lógica cuanto más se utilizan.

El "Cloud Computing" parte de la premisa de que la información debe estar en los servidores, los programas que actualmente funcionan y están instalados en el ordenador, funcionen a través de internet y que la ejecución de estas "Cloud Applications" sea totalmente online, pudiendo acceder a la aplicación y a la información desde cualquier terminal con acceso a internet.

En nuestra actualidad aun existen empresas que trabajan con proyectos en equipo, los mismos que están destinados a trabajar bajo limitantes de tiempo, con riesgos de fallos en conexión y restringidos a una red específica, retrasando de esta manera el proceso de desarrollo y quitando valor a la empresa. Los modelos de hosting para Cloud Computing brindan a los desarrolladores mejorar su productividad en trabajos a nivel de grupo denominadas gestores de proyectos, las mismas que han sido implementadas a partir de la necesidad de administrar su contenido en línea, muchas de estas empresas han convertido sus aplicaciones de escritorio a la web. Dichas empresas proveen soluciones completas para todas las etapas de desarrollo de un proyecto software pero están dirigidas a grupos o empresas desarrolladoras de gran escala ya que contratar este tipo de servicio tiene un costo elevado para un grupo pequeño de consumidores, además en el internet existen un conjunto de aplicaciones que son un complemento al momento de gestionar proyectos pero no tienen todos los requerimientos en un solo sitio o poseen limitantes de tiempo o funcionalidad ya que son versiones de prueba.

Hoy en día existen alrededor de 10 modelos de hosting para Cloud Computing que proveen distintos servicios para soluciones de este tipo. Entre los principales modelos tenemos a Amazon EC2, Google App Engine, Windows Azure Platform, CloudSwitch y Rackspace, todos estos nos ofrecen servicios de tecnología y que actúan de una forma de desarrollar aplicaciones y a su vez guardar la información que se procesa en Internet. Gracias a estas plataformas las empresas pueden ir creciendo a medida que lo necesiten y al paso en que incrementen sus capacidades.

1.4. JUSTIFICACIÓN

1.4.1. Justificación Teórica

La principal finalidad será mezclar y gestionar las aplicaciones de manera inteligente: un software que detecte en tiempo real la respuesta de una máquina para gestionar la energía, saber cuándo la demanda cae para optimizarla y ganar valor para la empresa.

Uno de los principales beneficios del Cloud Computing es que no hay necesidad de conocer la infraestructura detrás de esta, pasa a ser “una nube” donde las aplicaciones y servicios pueden fácilmente escalar, funcionar rápido y casi nunca fallan, sin conocer los detalles del funcionamiento de esta “nube”.

Otra de las características que nos brindan los modelos de hosting es que en caso de fallo, el último backup de la aplicación pasa a ser automáticamente la copia primaria y se genera uno nuevo, es decir, es Auto Reparable. Todo el sistema/arquitectura es predecible y eficiente. Si un servidor maneja 1000 transacciones, 2 servidores manejarán 2000 transacciones, volviéndose así un sistema escalable.

Las aplicaciones son independientes del hardware en el que corran, incluso varias aplicaciones pueden correr en una misma máquina o una aplicación puede usar varias máquinas a la vez, todo esto implementado en un entorno virtualizado. Además el sistema está creado con Multipropósito, de tal forma que permite a diferentes clientes compartir la infraestructura sin preocuparse de ello y sin comprometer su seguridad y privacidad.

Las infraestructuras de Cloud Computing proporcionan mayor capacidad de adaptación, recuperación de desastres completa y reducción al mínimo de los tiempos de inactividad, brindando una prestación de servicios de talla mundial.

No necesita instalar ningún tipo de hardware o software: una infraestructura 100% de Cloud Computing. La cualidad de la tecnología de Cloud Computing es su simplicidad debido al hecho de que su implementación es mucho más rápida y con menos riesgos, además requiere una menor inversión de tal manera que podrá empezar a trabajar muy rápidamente gracias a una infraestructura de Cloud Computing.

Sus aplicaciones en tecnología de Cloud Computing estarán disponibles en cuestión de semanas o meses, incluso con un considerable nivel de personalización o integración.

La infraestructura de Cloud Computing no sólo proporciona útiles funciones de personalización y configuración de aplicaciones, sino que también conserva las personalizaciones incluso después de las actualizaciones. Y aún mejor, la tecnología de Cloud Computing es ideal para el desarrollo de aplicaciones que estén en consonancia con las crecientes necesidades de su organización.

1.4.2. Justificación Aplicativa

Una vez elegido el proveedor, a través de el desarrollo de un sistema de Gestor de Proyectos en Línea se podrá coordinar mejor y más rápidamente las distintas tareas distribuidas mediante la asignación, proporcionando un calendario obligatorio.

La implementación de la un sistema de Gestor de Proyectos en Línea, independiente de la infraestructura de la ESPOCH permitirá un acceso sin restricciones de ubicación debido a que en otros tipos de plataformas implementadas en la institución su limitante principal fue que los miembros del equipo de desarrollo obligadamente debían estar unidos a la red de la misma.

Gracias al control de versiones del Almacén de información, ningún dato se pierde y puede ser accedida en cualquier momento. Todas las modificaciones

de esos datos estarán inmediatamente disponibles para todo el mundo. Esto reduce la duplicación de datos y su mantenimiento.

1.5. OBJETIVOS

1.5.1. Objetivo General

Realizar un análisis comparativo de la usabilidad de modelos hosting para Cloud Computing. Caso Práctico: desarrollo de un Sistema de Gestor de Proyectos en Línea.

1.5.2. Objetivos Específicos

- Estudiar las características de los modelos de hosting para Cloud Computing.
- Establecer los parámetros a comparar entre los modelos de hosting Google App Engine y Amazon EC2.
- Realizar las pruebas y análisis comparativo de la usabilidad de cada uno de los modelos de hosting para Cloud Computing.
- Implementar un Sistema de Gestor de Proyectos en Línea determinando sus módulos para poder generar estadísticas y reportes.

1.6. HIPÓTESIS

El modelo Amazon EC2 es la mejor opción con respecto a la usabilidad de Cloud Computing para implementar un Gestor de Proyectos en Línea.

CAPITULO II

MARCO TEÓRICO

2.1. LOS CIMIENTOS DE CLOUD COMPUTING

2.1.1. Qué es Cloud Computing

Las compañías que dominen “la nube” podrían llegar a ser los verdaderos actores del futuro, con esquemas de concentración muy importantes debido a la misma naturaleza de la actividad.

La computación en nube es un concepto que incorpora el software como servicio, como en la Web 2.0 y otros conceptos recientes, también conocidos como tendencias tecnológicas, que tienen en común el que confían en Internet para satisfacer las necesidades de cómputo de los usuarios.

2.1.2. Evolución de Cloud Computing

El concepto de la computación en la nube empezó en proveedores de servicio de Internet a gran escala, como Google, Amazon AWS y otros que construyeron su propia infraestructura. De entre todos ellos emergió una arquitectura: un sistema de recursos distribuidos horizontalmente, introducidos como servicios virtuales de IT escalados masivamente y manejados como recursos configurados y mancomunados de manera continua.

Tanto las nubes como los grid están diseñados para escalar horizontalmente de forma muy eficiente. Ambos están contruidos para resistir fallos de los elementos o nodos individuales. Ambos se cobran “por-uso”; pero mientras que los grids normalmente procesan los trabajos en lotes (“batchs”), con un punto definido de inicio y final, los servicios en nube pueden ser continuos. Lo que es más, las nubes expanden los tipos de recursos disponibles, almacenamiento de archivos, bases de datos, y servicios Web y extienden la aplicabilidad a la Web y a las aplicaciones de la empresa.

Al mismo tiempo, el concepto de capacidad de cómputo como servicio llegó a ser el foco de diseño y operaciones de TI. La infraestructura de los servicios de cómputo empezaba a ser comparable con el desarrollo de la electricidad como servicio público.

2.1.3. Cloud Computing y XaaS (Saas laas, Paas, Etc.)

XaaS es la esencia de la computación en nube, se refiere a un número cada vez mayor de servicios que se ofrecen por Internet, y no siempre a nivel local o en el lugar¹. Los ejemplos más comunes del XaaS son:

2.1.3.1. Software as a Service (SaaS)

En español *Software como Servicio*. Modelo de distribución de software donde una empresa brinda servicios de mantenimiento, soporte y operación que usará el cliente durante el tiempo que haya contratado el servicio. El cliente usará el sistema alojado por esa empresa, la cual mantendrá la información del cliente en sus sistemas y proveerá los recursos necesarios para explotar esa información. Ejemplos: Salesforce, Basecamp.

¹ <http://www.nubeblog.com/2008/10/15/saas-iaas-y-paas-las-tres-clases-de-cloud-computing/>

2.1.3.2. Infrastructure as a Service (IaaS)

En español *Infraestructura como Servicio*. Modelo de distribución de infraestructura de computación como un servicio, normalmente mediante una plataforma de virtualización. En vez de adquirir servidores, espacio en un centro de datos o equipamiento de redes, los clientes compran todos estos recursos a un proveedor de servicios externo. Una diferencia fundamental con el hosting virtual es que el provisionamiento de estos servicios se hace de manera integral a través de la web. Ejemplos: Amazon Web Services EC2 y GoGrid.

2.1.3.3. Platform as a Service (PaaS)

En español *Plataforma como Servicio*. Aunque suele identificarse como una evolución de SaaS, es más bien un modelo en el que se ofrece todo lo necesario para soportar el ciclo de vida completo de construcción y puesta en marcha de aplicaciones y servicios web completamente disponibles en la Internet. Otra característica importante es que no hay descarga de software que instalar en los equipos de los desarrolladores. PaaS ofrece múltiples servicios, pero todos provisionados como una solución integral en la web. Aunque algunos servicios de Amazon Web Services como SimpleDB y SQS yo los considero PaaS, esta afirmación puede ser discutida. Otro ejemplo es Google App Engine.

2.1.4. Características de los principales servicios

2.1.4.1. Agilidad

Permite desarrollar de manera más rápida y eficiente aplicaciones de nueva generación. Incrementa la flexibilidad de la creación de servicios y ofrece ciclos de desarrollo y pruebas rápidos, reduciendo dramáticamente el tiempo a mercado. Esto significa que nuevos negocios pueden estar desplegados en horas en vez de días, semanas o incluso meses.

2.1.4.2. Auto Reparable

En caso de fallo, el último backup de la aplicación pasa a ser automáticamente la copia primaria y se genera uno nuevo.

2.1.4.3. Costo

Se afirma que es muy reducido y los gastos de capital se convierte en gastos de funcionamiento. Esto reduce drásticamente las barreras de entrada, como la infraestructura.

2.1.4.4. Independencia de ubicación

Permiten a los usuarios acceder a los sistemas mediante un navegador web sin importar su ubicación o del dispositivo que está utilizando (por ejemplo, PC, móvil). Como la infraestructura es fuera de sitio (normalmente proporcionados por un tercero) y se accede a través de Internet, los usuarios pueden conectarse desde cualquier lugar.

2.1.4.5. Multi-arrendamiento

Permite el uso compartido de recursos y costes a través de un gran número de usuarios lo que permite la centralización de la infraestructura en lugares con menores costes (como el inmobiliario, la electricidad, etc)

2.1.4.6. Fiabilidad

Mejora mediante el uso de varios sitios redundantes, lo que hace a Cloud Computing adecuado para la continuidad del negocio y recuperación ante desastres.

2.1.4.7. Escalabilidad

Hace crecer de forma flexible a las necesidades de las demandas del cliente, ya sea por capacidad de almacenaje o por cualquier otro requerimiento.

2.1.4.8. Seguridad

La seguridad es a menudo tan buena o mejor que en los sistemas tradicionales, en parte porque los modelos de hosting son capaces de dedicar recursos a la solución de los problemas de seguridad que muchos clientes no pueden permitirse. Además, la complejidad de la seguridad es mucho mayor cuando los datos se distribuyen sobre un área más amplia y / o el número de dispositivos.

2.1.4.9. Mantenimiento

Aplicaciones de Cloud Computing son más fáciles de mantener, ya que no tiene que estar instalado en el equipo de cada usuario. Son más fáciles para el soporte y mejora, ya que los cambios alcanzan a los clientes al instante.

2.1.4.10. Medición

Los recursos de uso deben ser mensurables y deben ser medidos por el cliente y la aplicación, diario, semanal, mensual y anual. Esto permitirá a los clientes sobre la elección del modelo de hosting para Cloud Computing en el coste y la fiabilidad (QoS).

2.1.5. Aspectos Técnicos

2.1.5.1. Plataformas de Desarrollo

Las principales plataformas en las que funciona la nube informática son Windows y Linux

2.1.5.1.1. Windows Server

Es una marca que abarca una línea de productos de Microsoft Corporation. El nombre se ha utilizado en las ediciones de Microsoft Windows para servidores como las siguientes²:

- Windows 2000 Server, Está destinada a ser el servidor de archivos, impresión, web, FTP de una empresa pequeña a mediana. Su antecesor es Windows NT 4.0 Server. Es ideal para cuando no se requiere de un servidor dedicado a cada tarea o departamento, logrando de esta manera mantener todo centralizado en un solo servidor.
- Windows Server 2003, se podría considerar como un Windows XP modificado para labores empresariales, no con menos funciones, sino que estas están deshabilitadas por defecto para obtener un mejor rendimiento y para centrar el uso de procesador en las características de servidor; por ejemplo, la interfaz gráfica denominada Luna de Windows XP viene desactivada por lo que sólo se utiliza la interfaz clásica de Windows.
- Windows Server 2008, gestiona el sistema hasta el punto de que se puede llegar a controlar el hardware de forma más efectiva, se puede controlar mucho mejor de forma remota y cambiar de forma radical la política de seguridad, está diseñado para ofrecer a las organizaciones la plataforma más productiva para virtualización de cargas de trabajo, creación de aplicaciones eficaces y protección de redes. Ofrece una plataforma segura y de fácil administración, para el desarrollo y alojamiento confiable de aplicaciones y servicios web, también proporciona a los profesionales de TI más control sobre sus servidores e infraestructura de red y les permite centrarse en las necesidades críticas del negocio
- Windows Server 2008 R2, Las mejoras en el rendimiento fueron un área de desarrollo importante en esta versión; Microsoft indico que realizaron

² http://es.wikipedia.org/wiki/Windows_Server

trabajos para disminuir el tiempo de arranque, mejorar la eficiencia de operaciones E/S a la vez reducir la potencia de procesamiento y mejoras generales de velocidad en dispositivos de almacenamiento, especialmente en SCSI.

- Windows Small Business Server, sistema operativo basado en Windows Server con integración de software Microsoft Servers, diseñado para el funcionamiento de la infraestructura de la red (tanto la intranet de gestión y acceso a Internet), de las pequeñas y medianas empresas que no tengan más de 75 estaciones de trabajo o usuarios.
- Windows Essential Business Server, producto similar a Small Business Server, pero para empresas de tamaño medio, utiliza única consola de administración y gestión, a través de los cuales la recaudación de los servidores se pueden configurar. Software de terceros que también pueden utilizar la misma consola para presentar su interfaz de administración.
- Windows Home Server, sistema operativo para hogares, diseñado para compartición de archivos, transmisión multimedia, copias de seguridad automatizadas y acceso remoto

2.1.5.1.2. Linux

Es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux, que es usado con herramientas de sistema GNU. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera bajo los términos de la GPL (Licencia Pública General de GNU) y otra serie de licencias libres, entre los más utilizados tenemos³:

³ http://es.wikipedia.org/wiki/Fedora_%28distribuci%C3%B3n_Linux%29

- Centos (Community Enterprise Operating System), es un clon a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente liberado por Red Hat. Red Hat Enterprise Linux se compone de software libre y código abierto, pero se publica en formato binario usable (CD-ROM o DVD-ROM) solamente a suscriptores pagados. Como es requerido, Red Hat libera todo el código fuente del producto de forma pública bajo los términos de la Licencia pública general de GNU y otras licencias. Los desarrolladores de CentOS usan ese código fuente para crear un producto final que es muy similar al Red Hat Enterprise Linux y está libremente disponible para ser bajado y usado por el público, pero no es mantenido ni asistido por Red Hat. Existen otras distribuciones también derivadas de las fuentes de Red Hat.
- Fedora, es una distribución LinuxEs un sistema operativo cliente/servidor para propósitos generales basada en RPM, que se mantiene gracias a una comunidad internacional de ingenieros, diseñadores gráficos y usuarios que informan de fallos y prueban nuevas tecnologías. Cuenta con el respaldo y la promoción de Red Hat.
- Ubuntu, es una distribución Linux basada en Debian GNU/Linux que proporciona un sistema operativo actualizado, con un fuerte enfoque en la facilidad de uso y de instalación del sistema. Al igual que otras distribuciones se compone de múltiples paquetes de software normalmente distribuidos bajo una licencia libre o de código abierto. Estadísticas web sugieren que el porcentaje de mercado de Ubuntu dentro de las distribuciones Linux es de aproximadamente 50%, y con una tendencia a subir como servidor web.

2.1.5.2. Principales Modelos de hosting Cloud Computing

2.1.5.2.1. Amazon Web Services

Amazon Elastic compute cloud (Amazon E2) es un servicio web ofrecido por la empresa Amazon, que consiste en proporcionar capacidad y herramientas de computación en nube bajo un régimen variable. Tiene como objetivo permitir escalar los distintos servicios y necesidades de procesamiento de una manera fácil y ágil.

Posee una sencilla interfaz de servicios web, que permite al usuario obtener y configurar el servicio con mínimos problemas. Proporciona un control completo de los recursos y reduce el tiempo de arranque de sus servidores, lo que permite escalar recursos rápidamente según las necesidades del usuario. De ahí su nombre de "elástico".

Pues permite ampliar o reducir recursos con facilidad, según se modifiquen las necesidades. Provee herramientas de recuperación de datos y fuerte aislamiento frente a otros procesos realizados en sus máquinas.

Se apoya en las tecnologías de virtualización, permitiendo utilizar gran variedad de sistemas operativos a través de sus interfaces de servicios web, personalizarlos, gestionar permisos de acceso a la red y ejecutar tantos sistemas como desee.

El Amazon Web Services (AWS) es una colección de servicios informáticos a distancia (también llamada de servicios web) que forman parte de un Cloud Computing, que ofrece a través de Internet Amazon.com.

Amazon Web Services ofrece servicios en línea para los sitios web o aplicaciones de cliente. La mayoría de estos servicios no están expuestos directamente a los usuarios finales, sino que ofrecen una funcionalidad que otros desarrolladores pueden utilizar.

2.1.5.2.2. Google App Engine

Tiene configuración gratuita que permite a los usuarios ejecutar su aplicación web en la infraestructura de Google. Google App Engine facilita la escala del tráfico y almacenamiento de datos según la necesidad de crecimiento de la aplicación. Pretende convertir el Cloud Computing en un producto de uso comercial. Es decir, ofrece hospedaje a aplicaciones web para que los programadores no se tengan que preocupar de contratar más o menos servidores, configurar firewalls, switches, replicación, backups, etc.⁴

Este motor de aplicación no requiere servidores para mantener las aplicaciones y permite a los usuarios subir inmediatamente su aplicación a la nube.

Google App Engine ofrece un entorno completamente integrado, esto da una facilidad enorme.

2.1.5.2.3. Microsoft Azure

Windows Azure TM es un sistema operativo de servicios en la nube que sirve como el desarrollo, servicio de alojamiento y el medio ambiente de gestión de servicios para la plataforma Windows Azure. Windows Azure proporciona a los desarrolladores, almacenamiento de datos, y gestión de aplicaciones web en Internet a través de centros de datos de Microsoft.

2.1.5.2.4. Force.Com

Este modelo de hosting para Cloud Computing proporciona una de las plataformas que permite más rápido construir y desplegar aplicaciones empresariales. Force.com permite la construcción de aplicaciones empresariales básicas en la demanda como de planificación de recursos empresariales (ERP), gestión de recursos humanos (HRM) y Supply Chain Management (SCM).

⁴ http://en.wikipedia.org/wiki/Google_App_Engine

CAPITULO III

ANÁLISIS COMPARATIVO

3.1. INTRODUCCIÓN

La determinación de un modelo de hosting para Cloud Computing que se va a utilizar para la implementación de un sistema informático, debe ser una de las decisiones más importantes que deben tomar los desarrolladores, la misma que debe estar basada en un profundo y minucioso análisis, de acuerdo a criterios de comparación o parámetros, además que debe cumplir con los requerimientos para satisfacer las necesidades de un sistema de gestión.

3.2. DETERMINAR LOS MODELOS DE HOSTING A COMPARAR

En la actualidad existen en el mercado una gran cantidad de modelos de hosting para Cloud Computing que son actuales, bastante populares y de gran importancia para los desarrolladores de las cuales hemos seleccionado dos modelos por las razones que explican a continuación:

- En la siguiente página⁵ se describe: “Varias son las grandes empresas que se han dedicado a ofrecer estos servicios, promoviendo el fácil acceso a nuestra información, los bajos costos, la escalabilidad y muchas

⁵ <http://www.maestrosdelweb.com/editorial/cloud-computing-nueva-era-de-desarrollo/>

características que nos hace pensar en la comodidad que nos brindan, entre ellas podemos mencionar:”

- Google Apps
 - Amazon Web Services
 - Azure de Microsoft
 - Rackspace
-
- En el siguiente portal⁶ se destaca lo siguiente: “El reciente lanzamiento del servicio App Engine de Google y la fuerte apuesta Blue Cloud de IBM se unen a experiencias ya en marcha de compañías como Amazon y Salesforce.com que refuerzan el carácter “rompedor” con que el modelo Cloud Computing promete irrumpir masivamente en el escenario de la entrega de estos servicios”

 - En el siguiente portal⁷, define a los siguientes como los principales tipos de Cloud Computing:
 - salesforce.com
 - Google App Engine
 - Amazon Web Services

De acuerdo a estas páginas, encuestas y a otras consultadas y no mencionadas se han seleccionado dos modelos para la realización de este análisis comparativo: Amazon EC2 y Google App Engine, las mismas que gozan de una gran popularidad entre los desarrolladores, ya que son las más comunes en los resultados de las encuestas antes mencionadas, además cuentan con una amplia gama de recursos para su utilización.

⁶ <http://www.networkworld.es/Cloud-Computing:-Las-TI-como-servicio/seccion-recursos/articulo-191003>

⁷ http://www.venamcham.org/index.php?option=com_content&view=article&id=242%3Acomite-de-tecnologia-de-lainformacion-cloud-computing&catid=8%3Acomitesaldia&lang=es

3.3. ANÁLISIS DE LOS MODELOS SELECCIONADOS

A continuación analizaremos más de cerca cada uno de los modelos definidos anteriormente:

3.3.1. Amazon Ec2

Amazon Elastic Compute Cloud (Amazon EC2) es un servicio web que proporciona capacidad de cálculo de tamaño variable en la nube. Está diseñado para que la escalabilidad de la web de Computación sea más fácil para los desarrolladores.

La interfaz web de Amazon permite obtener una capacidad de configuración con una mínima fricción. Le proporciona un control total de los recursos informáticos y le permite ejecutar en un entorno informático seguro y confiable.

Amazon EC2 reduce el tiempo necesario para obtener y dar nuevos servicios de arranque en minutos, lo que le permite conseguir de forma rápida la capacidad de escalabilidad, tanto hacia arriba como hacia abajo. Amazon EC2 ha cambiado la economía de la informática, por lo que le permite pagar sólo por la capacidad que realmente utiliza. Amazon EC2 proporciona a los desarrolladores las herramientas para construir aplicaciones resistentes al fracaso y aislarse de los escenarios de fallo común⁸.

3.3.1.1. Funcionalidad

Amazon EC2 presenta un cierto entorno virtual informático, lo que le permite utilizar el servicio de interfaces web para poner en marcha las instancias con una variedad de sistemas operativos, carga con su entorno de aplicación personalizada, maneja los permisos de acceso a la red y ejecuta la aplicación utilizando pocos o muchos servicios como usted desee.

Para utilizar Amazon EC2, simplemente:

⁸ <http://aws.amazon.com/es/ec2/>

- Seleccione una plantilla de imagen pre-configurada para ejecutarla inmediatamente. O crear una AMI (Amazon Machine Image) la cual contendrá su aplicación, librerías, datos y parámetros de configuración asociados.
- Configurar la seguridad y el acceso a la red en su instancia de Amazon EC2.
- Seleccione el tipo de instancia y sistema operativo desea, arranque, detenga y monitoree cuantas instancias de sus AMI necesite, utilizando las API de servicio web, o la variedad de herramientas de administración que se le provee.
- Determine si desea ejecutar en múltiples ubicaciones, utilizando IP estáticas en los endpoints, o agregar bloques de persistencias para almacenar sus instancias.
- Pague sólo por los recursos que realmente consume, como ejemplo: horas o transferencia de datos.

3.3.1.2. Características

Amazon EC2 proporciona una serie de características de gran alcance para la construcción escalable, aplicaciones de clase empresarial, incluyendo:

- Amazon Elastic Block Store, ofrece almacenamiento persistente, fiabilidad y disponibilidad para las instancias Amazon EC2.
- Cuando se utiliza como partición de arranque, las instancias Amazon EC2 pueden detenerse y subsecuentemente reiniciarse, permitiéndole pagar solo por los recursos de almacenamiento utilizados durante el mantenimiento de las instancias.

- Amazon EBS son automáticamente replicados en el backend.
- Amazon EBS provee la capacidad de crear point-in-time consistiendo en capturas de los volúmenes que son almacenadas en Amazon S3, y automáticamente replicados a través de múltiples zonas de disponibilidad.
- Amazon EC2 provee la capacidad para ubicar instancias en ubicaciones múltiples. Las ubicaciones Amazon EC2 están compuestas de regiones y zonas de disponibilidad. Las zonas de disponibilidad están en ubicaciones distintas, las cuales han sido diseñadas para aislarse de otras zonas de disponibilidad con fallas y zonas de baja latencia con otras zonas en la misma región. Al lanzar las instancias en zonas separadas puede proteger su aplicación de fallas en una ubicación.
- Amazon Virtual Private Cloud, es un puente seguro entre la infraestructura existente de la compañía y la nube AWS. Permite a las empresas conectar y aislar los recursos computacionales AWS a través de una conexión VPN, y extender sus capacidades de administración tales como servicios de seguridad, firewalls y sistemas de detección de intrusión.
- Proporciona visualización de la utilización de recursos, rendimiento operativo y en general la demanda de patrones, incluyendo datos como el uso de CPU, lectura/escritura en disco y el tráfico de red.
- Permite escalar automáticamente la capacidad de Amazon EC2, de acuerdo a las condiciones definidas. Con Auto Scaling se asegura que el número de instancias que se encuentra utilizando durante picos de demanda son óptimos, de esta manera permite minimizar los costos.
- Permite tener el control completo y acceso como root a cada una de las instancias, además, se puede interactuar con ellas como lo haría con cualquier máquina. Las instancias se pueden reiniciar de forma remota,

además, puede detener la instancia, manteniendo los datos en la partición de arranque y posteriormente reiniciar la misma instancia utilizando los servicios web API. También se tiene acceso a la consola de salida de las instancias.

- Amazon proporciona la elección de múltiples tipos de instancias, sistemas operativos y paquetes de software. Amazon EC2 le permite seleccionar la configuración de la memoria, CPU, almacenamiento de instancias, y el tamaño de la partición de arranque que optimiza la elección del sistema operativo y aplicación. Por ejemplo, la elección de sistemas operativos incluye numerosas distribuciones de Linux, Microsoft Windows Server y OpenSolaris.
- Amazon EC2 ofrece un ambiente altamente fiable, en casos de sustitución se lo realiza rápidamente y de manera predecible. Los servicios se ejecutan dentro de la infraestructura de red y centros de datos de Amazon. El nivel de servicio de Amazon EC2 tiene una disponibilidad del 99.95% para cada región de Amazon EC2.
- Amazon EC2 proporciona numerosos mecanismos para asegurar los recursos informáticos, incluye el servicio de interfaces web para configurar el firewall que controla el acceso a la red y entre grupos de instancias.
- Amazon EC2 pasa a ser un beneficio financiero a gran escala, ya que se paga una tasa muy baja por la capacidad de cómputo que realmente consumen.

3.3.1.3. Restricciones

- Para poder escalar una aplicación de forma horizontal, es necesario agregar más procesador, más memoria, más disco, o más instancias, que vendrían a ser como mas servidores en paralelo.

- Para el mantenimiento y la administración de una aplicación en Amazon EC2 se necesita conocimientos medios de administración de servidores ya que Ec2 requiere un mantenimiento especializado casi constante.
- Con respecto a los costos Amazon resulta caro en sitios chicos. A pesar de que se paga lo que se consume, para grandes aplicación implican costos demasiado elevados, resulta complicado conseguir instancias más poderosas dentro de los usd 100 mensuales.

3.3.1.4. Instancias / Cuotas

3.3.1.4.1. Norma de instancias

Las instancias de Amazon EC2 son muy adecuadas para la mayoría de las aplicaciones.

Tabla III - I Norma de Instancias

Instancia	Memoria	Unidad	Almacenamiento	Plataforma
Instancia Pequeña	1.7 GB de memoria	1 EC2 Unidad de Computo(1 Nucleo Virtual con 1 EC2 Unidad de Computo)	160 GB de almacenamiento de instancia local	Plataforma de 32-bit
Instancia Grande	7.5 GB de memoria	4 EC2 Unidad de Computo (2 Nucleos Virtuales con 2 EC2 Unidades de computo cada uno)	850 GB de almacenamiento de instancia local	Plataforma de 64-bit
Instancia Extra Grande	15 GB de memoria	8 EC2 Unidad de Computo (4 Nucleos Virtuales con 2 EC2 Unidades de computo cada uno)	1690 GB de almacenamiento de instancia local	Plataforma de 64-bit

3.3.1.4.2. Instancias con Gran Capacidad de Memoria

Las instancias de este grupo ofrecen una gran cantidad de espacio en memoria, incluyendo mejoras para aplicaciones de caching y base de datos.

Tabla III - II Instancias con Gran Capacidad de Memoria

Instancia	Memoria	Unidad	Almacenamiento	Plataforma
Memoria Alta extra grande	17.1 GB de memoria	6.5 ECU (2 Nucleos Virtuales con 3.25 EC2 Unidades de Computo cada uno)	420 GB de almacenamiento de instancia local	Plataforma de 64-bit
Memoria Alta de doble instancia extra grande	34.2 GB de memoria	13 EC2 Unidades de computo (4 Nucleos Virtuales con 3.25 EC2 Unidades de Computo cada uno)	850 GB de almacenamiento de instancia local	Plataforma de 64-bit
Memoria Alta de cuádruple instancia extra grande	68.4 GB de memoria	26 EC2 Unidades de Computo (8 Nucleos Virtuales con 3.25 EC2 Unidades de computo)	1690 GB de almacenamiento de instancia local	Plataforma de 64-bit

3.3.1.4.3. Instancias con gran capacidad de CPU

Las instancias de este grupo tienen proporcionalmente más recursos de CPU que la memoria (RAM) y son muy adecuadas para aplicaciones de cálculo intensivo.

Tabla III - III Instancias con gran capacidad de CPU

Instancia	Memoria	Unidad	Almacenamiento	Plataforma
CPU Alto de media instancia	1.7 GB de memoria	5 EC2 Unidades de computo (2 Nucleos virtuales con 2.5 EC2 Unidades de computo cada uno)	350 GB de almacenamiento de instancia local	Plataforma de 32-bit
CPU Alto de instancia extra grande	7 GB de memoria	20 EC2 Unidades de computo(8 Nucleos virtuales con 2.5 EC2 Unidades de computo cada uno)	1690 GB de almacenamiento de instancia local	Plataforma de 64-bit

3.3.1.5. Sistema Operativo

Las AMI (Amazon Machine Images) están pre-configuradas con una lista de sistemas operativos. Se trabaja con sociedades y una comunidad de usuarios los cuales proveen una gran cantidad de opciones en cuanto AMI. Además brinda la oportunidad de que el usuario haga uso de sus propias aplicaciones. Los sistemas operativos que se encuentran actualmente disponibles son:

Tabla III - IV Sistema Operativo

Sistema Operativo		
Red Hat Enterprise Linux	Windows Server 2003/2008	Oracle Enterprise Linux
OpenSolaris	openSUSE Linux	Ubuntu Linux
Fedora	Gentoo Linux	Debian

3.3.1.6. Software

Amazon EC2 permite a nuestros socios y clientes construir y personalizar las (AMI) con software basado en sus necesidades. Tenemos cientos de AMI gratuitas y de pago. Una pequeña muestra del software disponible para uso en la actualidad dentro de Amazon EC2 incluye:

Tabla III - V Base de Datos disponibles en Amazon EC2

Bases de datos	Procesamiento por lotes	Alojamiento Web
IBM DB2	Hadoop	Apache HTTP
IBM Informix Dynamic Server	Condor	IIS /Asp.Net
Microsoft SQL Server Standard 2005	Open MPI	IBM Lotus Web Content Management
MySQL Enterprise		IBM WebSphere Portal Server IB
Oracle Database 11g		

Tabla III - VI Entornos de Desarrollo en Amazon EC2

Entornos de desarrollo de aplicaciones	Servidores de aplicaciones	Codificación de Video y Streaming
IBM sMash	IBM WebSphere Application Server	Wowza Media Server Pro
JBoss Enterprise Application Platform	Java Application Server	Windows Media Server
Ruby on Rails	Oracle WebLogic Server	

3.3.1.7. Precios

Se paga solo lo que se utiliza. No existe un monto mínimo. Puede estimar su gasto mínimo utilizando AWS Simple Monthly Calculator. Los precios se basan en la Región en la que la instancia se está ejecutando. Para una comparación detallada entre instancias bajo demanda e instancias Spot.

3.3.1.7.1. Instancias bajo demanda

On-Demand instancias le permiten pagar por la capacidad de cómputo por horas sin compromisos a largo plazo. Esto le libera de los costes y complejidades de la planificación, compra y mantenimiento de hardware y transforma lo que comúnmente son costos fijos y elevados en costos variables y menores.

Los precios a continuación incluye el coste para el funcionamiento de AMIS público y privado en el sistema operativo específico.

- US – N. Virginia
- US – N. California
- EU – Ireland
- APAC – Singapore

Tabla III - VII Precios en Amazon EC2

Estandar de demanda de Instancias	Uso de Linux/UNIX	Uso de Windows
Pequeño	\$0.085 por hora	\$0.12 por hora
Grande	\$0.34 por hora	\$0.48 por hora
Extra Grande	\$0.68 por hora	\$0.96 por hora
Memoria Alta bajo la demanda de instancias		
Extra Grande	\$0.50 por hora	\$0.62 por hora
Doble Extra Grande	\$1.20 por hora	\$1.44 por hora
Cuádruple Extra Grande	\$2.40 por hora	\$2.88 por hora
CPU Alto bajo la demanda de instancias		
Mediano	\$0.17 por hora	\$0.29 por hora
Extra Grande	\$0.68 por hora	\$1.16 por hora

El precio de la hora consumida por cada tipo de instancia es desde el momento en que se puso en marcha la instancia hasta que sea resuelta. Cada instancia parcial de horas consumidas se facturará como una hora completa.

3.3.1.7.2. Instancias Reservadas

Instancias reservadas le dan la opción de hacer un pago único para cada instancia que desea reservar, y a su vez reciben un descuento significativo en la carga horaria de uso para esa instancia. Después del tiempo de pago único de una instancia, esa instancia está reservada para su uso, y no tiene ninguna obligación adicional, cuando no se utiliza la instancia, no se tendrá que pagar cargos por uso de ella.

Tabla III - VIII Precio de Instancias reservadas en Amazon EC2

Estándar de Instancias Reservadas	1 Año Plazo	3 Año Plazo	Uso de Linux/UNIX	Uso de Windows
Pequeño	\$227.50	\$350	\$0.03 por hora	\$0.05 por hora
Grande	\$910	\$1400	\$0.12 por hora	\$0.20 por hora
Extra Grande	\$1820	\$2800	\$0.24 por hora	\$0.40 por hora
Memoria Alta para instancias reservadas				
Extra Grande	\$1325	\$2000	\$0.17 por hora	\$0.24 por hora
Doble Extra Grande	\$3185	\$4900	\$0.42 por hora	\$0.55 por hora
Cuádruple Extra Grande	\$6370	\$9800	\$0.84 por hora	\$1.10 por hora
CPU Alta para instancias reservadas				
Mediano	\$455	\$700	\$0.06 por hora	\$0.125 por hora
Extra Grande	\$1820	\$2800	\$0.24 por hora	\$0.50 por hora

Las Instancias Reservadas se pueden adquirir por 1 o 3 años plazo, y solo se paga una vez por cada instancia, este valor no es reembolsable.

Si decide utilizar Microsoft aumenta el valor que cobra por Windows, que en consecuencia puede aumentar el uso de la tarifa por hora. Las instancias reservadas estarán disponibles para Linux / UNIX y sistemas operativos de Windows. No hay disponible una instancia reservada que se pueden utilizar con Microsoft SQL Server.

3.3.1.7.3. Instancias Spot

Instancias Spot le permiten hacer una oferta para Amazon EC2. Estas instancias pagan un precio de contado que se fija por Amazon EC2 y fluctúa periódicamente en función de la oferta y la demanda de capacidad de Spot. Para utilizar Instancias Spot, se coloca una instancia de solicitud de Spot, especificando el tipo de instancia, la región deseada, el número de Instancias Spot que desea ejecutar, y el precio máximo que está dispuesto a pagar por cada instancia de hora.

Tabla III - IX Precio de Instancias Spot en Amazon EC2

Estandar de Instancia Spot	Uso de Linux/UNIX	Uso de Windows
Pequeño	\$0.03 por hora	\$0.051 por hora
Grande	\$0.114 por hora	\$0.206 por hora
Extra Grande	\$0.361 por hora	\$0.409 por hora
Memoria alta para instancias Spot		
Extra Grande	\$0.169 por hora	\$0.23 por hora
Doble Extra Grande	\$0.41 por hora	\$0.55 por hora
Cuádruple Extra Grande	\$0.799 por hora	\$1.082 por hora
CPU Alto para instancias Spot		
Mediano	\$0.06 por hora	\$0.126 por hora
Extra Grande	\$0.243 por hora	\$0.515 por hora

3.3.1.7.4. Transferencia de datos

La fijación de precios a continuación se basa en los datos transferidos "in" y "out" de Amazon EC2.

Tabla III - X Precio por Transferencia de Datos en Amazon EC2

Transferencia de Datos de Entrada	US & EU Regions	APAC Region
Todas las Transferencia de Datos	Gratis hasta 30 de junio 2010	Gratis hasta 30 de junio 2010
Transferencia de Datos de Salida		
Primer GB por Mes	\$0.00 por GB	\$0.00 por GB
Hasta 10 TB por Mes	\$0.15 por GB	\$0.19 por GB
Próximos 40 TB por mes	\$0.11 por GB	\$0.15 por GB
Próximos 100 TB por mes	\$0.09 por GB	\$0.13 por GB
Más de 150 TB por mes	\$0.08 por GB	\$0.12 por GB

No hay transferencia de carga de datos entre Amazon EC2 y otros servicios web Amazon dentro de la misma región (es decir, entre Amazon EC2 Oeste de los EE.UU. y Amazon S3 en el oeste de los EE.UU.). Los datos transferidos entre instancias Amazon EC2 situada en las zonas de disponibilidad diferentes en la misma región se cargará la transferencia de datos regional.

Para otros Amazon Web Services se factura por separado de Amazon EC2.

3.3.1.7.5. Disponibilidad de transferencia de datos por zona

\$0,00 dólares por GB. Todos los datos transferidos entre las instancias en la misma Zona de disponibilidad utilizando direcciones IP privadas.

3.3.1.7.6. Transferencia de datos Regionales

\$0,01 dólares por GB in / out - todos los datos transferidos entre las instancias en las zonas de disponibilidad diferentes en la misma región.

Si se requiere comunicarse con la dirección pública o la dirección IP elástica el equilibrador de carga en el interior de la red Amazon EC2, tendrá que pagar tasas de transferencia de datos regional, incluso si las instancias se encuentran

en la misma Zona de disponibilidad. Para la transferencia de datos dentro de la misma Zona de disponibilidad, usted puede fácilmente evitar este cargo (y obtener un mejor rendimiento de red) mediante su IP privado siempre que sea posible.

3.3.1.7.7. Amazon Elastic Block Store

Tabla III - XI Precio de Amazon Elastic Block Store

Amazon EBS Volumes
<ul style="list-style-type: none">• \$0.10 por GB-mes de almacenamiento provisional• \$0.10 por 1 million de peticiones de I/O
Amazon EBS Instancias a Amazon S3 (el mismo precio que Amazon S3)
<ul style="list-style-type: none">• \$0.15 por GB-mes de datos almacenados• \$0.01 por 1,000 peticiones para PONER (al guardar una instantánea)• \$0.01 por 10,000 peticiones para OBTENER (cuando se carga una imagen)

3.3.1.7.8. Direcciones IP elásticas

Sin coste para direcciones IP elásticas mientras está en uso

- \$0.01 dólares por no suscriptores por hora o completa
- \$0.00 dólares por reasignar una dirección IP

3.3.1.7.9. Amazon CloudWatch

- 0,015 dólares por cada instancia de horas (o una hora parcial).

3.3.1.7.10. Auto Escalabilidad

Auto Escalabilidad está habilitado por Amazon CloudWatch y no conlleva costos adicionales. Cada instancia y puesta en marcha por Auto Escala se habilita automáticamente para el seguimiento y vigilancia del Amazonas CloudWatch.

3.3.1.7.11. Equilibrio de carga elástica

- 0,025 \$ por hora de equilibrador de carga elástico (u hora parcial)
- 0,008 \$ por GB de datos procesados por un equilibrador de carga elástico

3.3.2. Google App Engine

Google App Engine permite ejecutar aplicaciones web en la infraestructura de Google. Las aplicaciones App Engine son fáciles de crear, mantener y actualizar al ir aumentando el tráfico y las necesidades de almacenamiento de datos. Con App Engine, no necesita utilizar ningún servidor, sólo se tendrá que subir la aplicación para que los usuarios puedan empezar a utilizarla.

Puede proporcionar a la aplicación su propio nombre de dominio (como por ejemplo <http://www.example.com/>) a través de Google Apps. También se puede proporcionar a la aplicación un nombre gratuito del dominio appspot.com. Se podrá compartir la aplicación con todo el mundo o limitar el acceso a los miembros de una organización.

Google App Engine admite aplicaciones escritas en varios lenguajes de programación. Gracias al entorno de tiempo de ejecución Java de App Engine se puede crear una aplicación a través de tecnologías Java estándar, que incluyen JVM, servlets Java y el lenguaje de programación Java o cualquier otro lenguaje que utilice un intérprete o compilador basado en JVM como, por ejemplo, JavaScript o Ruby. App Engine también ofrece un entorno de tiempo de ejecución Python dedicado, que incluye un rápido intérprete Python y la biblioteca estándar Python. Los entornos de tiempo de ejecución Java y Python se generan para garantizar que la aplicación se ejecuta de forma rápida, segura y sin interferencias de otras aplicaciones en el sistema.

3.3.2.1. Funcionalidad

Con App Engine, sólo se paga lo que se utiliza. No existen costes de configuración ni tarifas recurrentes. Los recursos que utiliza la aplicación, como por ejemplo el almacenamiento y el ancho de banda, se miden por gigabytes y se facturan según competitivas tarifas. Se controla la cantidad máxima de recursos que consume la aplicación, de modo que siempre permanezcan dentro del presupuesto.

Se puede empezar a utilizar App Engine de forma totalmente gratuita. Todas las aplicaciones pueden utilizar hasta un GB de almacenamiento y suficiente CPU y ancho de banda como para permitir un servicio eficaz de la aplicación de alrededor de 5 millones de visitas a la página al mes, totalmente gratuitas.

Cuando se habilita la facturación para la aplicación, se incrementan los límites gratuitos y sólo se paga aquellos recursos que se utilice por encima de los niveles gratuitos.

3.3.2.2. Características

- Google App Engine ofrece un entorno completamente integrado. Esto da una facilidad enorme, pero con muchas limitaciones.
- Permite a empresas y otras organizaciones ejecutar sus aplicaciones web en la sólida infraestructura de Google, la misma que le permite responder a 3 mil millones de búsquedas al día y ofrecer Google Apps a un millón de organizaciones en el mundo.
- Implementaciones rápidas y sin problemas. Google App Engine proporciona un entorno de aplicación completamente integrado, por lo que no requiere ensamblajes ni procesos complicados.

- Las aplicaciones pueden crecer fácilmente. Google App Engine facilita la creación de aplicaciones escalables, destinadas a unos pocos usuarios o a millones, sin complejidades técnicas. Entre sus muchas funciones, Google App Engine incluye el soporte para tareas regulares, la importación y exportación de bases de datos, el acceso a datos protegidos con firewall, y soporte a estándares y a lenguajes como Python y Java, entre otras.
- Administración centralizada: Una nueva administración de la sociedad centrada en la consola permite administrar todas las aplicaciones del dominio.
- Fiabilidad y soporte: 99,9% de acuerdo a los niveles de servicio, con el apoyo de desarrolladores prima disponible.
- Seguro de manera predeterminada: sólo los usuarios de su dominio Google Apps puede acceder a las aplicaciones por defecto.
- Auspicio de bases de datos SQL, SSL en el dominio de acuerdo a la empresa para las comunicaciones seguras y el acceso a los servicios avanzados de Google.
- Comienzo fácil App Engine es una completa pila de desarrollo que emplea tecnologías habituales para crear y alojar aplicaciones web. En App Engine se puede crear código de determinada aplicación, probar la aplicación en un equipo local y subir a Google únicamente haciendo clic en un botón o introduciendo una secuencia en la línea de comandos. Una vez que se haya subido la aplicación a Google, el mismo que se encargará de su alojamiento y mantenimiento, y el cliente no tendrá que preocuparse de la administración del sistema, de la activación de instancias nuevas de la aplicación, de la fragmentación de la base de datos ni de la adquisición de equipos.

- Las aplicaciones podrán aprovechar las mismas tecnologías escalables sobre las que están creadas las aplicaciones de Google como, por ejemplo, BigTable y GFS. App Engine dispone de una función de escalabilidad automática. App Engine satisface las necesidades del cliente independientemente del número de usuarios que disponga y de la cantidad de datos que almacene su aplicación.
- La infraestructura de Google es famosa por su gran fiabilidad y por su alto rendimiento. A todas las aplicaciones de App Engine aplica las mismas políticas de seguridad, privacidad y protección de datos empleadas en Google.
- Período de prueba libre de riesgos. Crear una aplicación en App Engine brinda facilidades en el proceso de desarrollo incluyendo su gratuidad.⁹

3.3.2.3. Restricciones

- Los desarrolladores tienen acceso de sólo lectura para el sistema de archivos de App Engine.
- Los usuarios pueden cargar módulos de Python arbitrarias, pero sólo si son Python, C y Pyrexia que los módulos no son compatibles.
- No es compatible con dominios naked (sin www) como `http://example.com` , debido al uso de DNS los registros CNAME para equilibrar la carga. (sugerida solución consiste en utilizar DNS proveedor de redirección HTTP a un subdominio, por ejemplo " `www.example.com`.)
- No permite cambiar algunas configuraciones por defecto del embarcadero como la aceptación de un formulario POST / GET más de 200KB de tamaño.

⁹ <http://es.debugmodeon.com/articulo/introduccion-a-google-app-engine>

3.3.2.4. Instancias / Cuotas

Una aplicación de App Engine consume recursos hasta cierta cantidad máxima también denominados *cuotas*. Gracias a las cuotas, App Engine garantiza que determinadas aplicaciones no superen el presupuesto y que, otras aplicaciones que se ejecuten en App Engine no afecten al rendimiento de dicha aplicación.

- Cuotas facturables y fijas
- Renovación de recursos
- Cuotas por minuto
- Reducción de un recurso
- Recursos
 - Solicitudes
 - Almacén de datos
 - Correo electrónico
 - Extracción de URL
 - Manipulación de imágenes
 - Memcache
 - Implementaciones

3.3.2.4.1. Cuotas Facturables y Fijas

Cada recurso de App Engine se mide con respecto a uno de los dos tipos de cuotas: una “cuota facturable” o una “cuota fija.”

Las cuotas facturables son el límite máximo de recursos que el administrador de la aplicación establece para evitar que el coste de la aplicación supere el presupuesto. Cada aplicación obtiene una cantidad gratuita de cada cuota facturable. Se puede aumentar las cuotas facturables de la aplicación a través de la habilitación de la facturación, la configuración de un presupuesto diario y a continuación, de la asignación del presupuesto a las cuotas. Sólo se cargarán los recursos que la aplicación utilice actualmente, y sólo la cantidad de recursos que utilices por encima del umbral de la cuota gratuita.

Una vez que se habilite la facturación para la aplicación, se puede establecer el presupuesto diario y ajustar las asignaciones de cuota para la aplicación a través de la consola de administración.

Las cuotas fijas son límites máximos de recursos que establece App Engine para garantizar la integridad del sistema. Estos recursos describen los límites de la arquitectura, y se estima que todas las aplicaciones se ejecuten dentro de los mismos límites. Garantizan que otra aplicación que esté consumiendo demasiados recursos no afecte al rendimiento de la aplicación.

3.3.2.4.2. Renovación de recursos

App Engine registra el uso de cada recurso que hace una aplicación durante un día de calendario y considera el recurso agotado cuando la cantidad alcanza la cuota de la aplicación para dicho recurso. Un día de calendario es un período de 24 horas que empieza a media noche, hora del Pacífico. App Engine restablece todas las medidas de recurso al empezar cada día, excepto en el caso de los datos almacenados que siempre representan la cantidad de almacenamiento de la base de datos en uso.

Nota del historial: el ciclo de renovación de 24 horas se introdujo en diciembre de 2008. Sustituyó a un sistema más complicado de renovación "continua", para facilitar el informe y el control del uso de recursos.

3.3.2.4.3. Cuotas por minuto

Además de las cuotas diarias descritas anteriormente, App Engine modera la rapidez con la que una aplicación consume un recurso, a través de las cuotas por minuto. Esto impide que la aplicación consuma todo el contenido de su cuota en un breve período de tiempo y evita que otras aplicaciones afecten a la aplicación monopolizando un determinado recurso.

Si la aplicación consume un recurso demasiado rápido y se reduce a uno de los límites por minuto, la palabra "Limited" aparecerá en la cuota adecuada de la pantalla de detalles de cuota de la consola de administración. Las solicitudes de recursos que ya han alcanzado su máximo por minuto se deniegan.

Al igual que en el caso de las cuotas fijas diarias, existen dos niveles de cuotas por minuto, dependiendo de si se ha habilitado o no la facturación.

3.3.2.4.4. Reducción de un recurso

Cuando una aplicación consume todo el contenido de un recurso asignado, el recurso se vuelve no disponible hasta que se renueva la cuota. Esto puede querer decir que la aplicación no va a funcionar hasta que se renueve la cuota.

En el caso de los recursos requeridos a la hora de iniciar una solicitud, cuando el recurso se ha agotado, App Engine devuelve un código de estado prohibido HTTP 403 para la solicitud en lugar de ejecutar un controlador de solicitud. Los recursos que se muestran a continuación tienen dicho comportamiento:

- Solicitudes
- Tiempo de CPU
- Ancho de banda, de entrada y de salida

Para los demás recursos, si el recurso se ha agotado y la aplicación intenta consumir el recurso, se generará una excepción. La aplicación puede filtrar dicha excepción y controlarla, como si se mostrara un mensaje de error descriptivo para el usuario.

Si se supera inesperadamente la cuota de consumo de recursos del sistema, se puede ajustar el rendimiento de la aplicación.

3.3.2.4.5. Recursos

Una aplicación puede utilizar los siguientes recursos, sujetos a cuotas. Los recursos evaluados en cuanto a cuotas facturables se indican con el término "(billable)". Las cantidades del recurso representan una asignación de un período de 24 horas.

El coste de recursos facturables adicionales aparece en la página de facturación.

- **Solicitudes**

Es el número total de solicitudes de la aplicación. Las cuotas por minuto para las aplicaciones que tienen habilitada la facturación permiten hasta 500 solicitudes por segundo; más de mil millones de solicitudes al mes. Si la aplicación requiere cuotas superiores a las de los valores de "facturación habilitada".

- **Ancho de banda de salida**

Es la cantidad de datos que envía la aplicación en respuesta a las solicitudes. Esto incluye los datos enviados en respuesta a las solicitudes seguras y a las no seguras, a los datos enviados en mensajes de correo electrónico y a los datos de solicitudes HTTP de salida que envía el servicio de extracción de URL.

- **Ancho de banda de entrada**

Es la cantidad de datos que recibe la aplicación de las solicitudes. Esto incluye los datos que recibe la aplicación de las solicitudes seguras y de las no seguras y de los datos recibidos en respuesta a solicitudes HTTP del servicio de extracción de URL.

- **Tiempo de CPU**

Es el tiempo de procesamiento total empleado en la administración de solicitudes, que incluye el tiempo de ejecución de la aplicación y el de la realización de operaciones del almacén de datos. Esto no incluye el tiempo de espera de otros servicios como, por ejemplo, la espera a que se devuelva una extracción de URL o a que el servicio de imágenes transforme una imagen.

El tiempo de CPU se notifica en "segundos" y es equivalente al número de ciclos de CPU que un procesador Intel x86 de 1,2 GHz puede ejecutar en dicha cantidad de tiempo. El número actual de ciclos de CPU empleados varía mucho en función de las condiciones internas de App Engine, de modo que este número se ajusta por motivos de notificación a través de dicho procesador como medida de referencia.

Una herramienta que ayuda a identificar las áreas de la aplicación que emplean grandes cantidades de cuota de tiempo de ejecución de CPU es el módulo "cProfile".

Se puede consultar el tiempo de CPU que se emplea para suministrar cada solicitud en la sección de registros de la consola de administración. También resulta útil comprender qué operaciones del almacén de datos contribuyen en el uso de CPU.

- Escribe el uso del almacén de datos aproximadamente cinco veces, tantas veces como lo lee la CPU.
- Las escrituras que actualizan índices requieren más cantidad de CPU que las escrituras que no lo hacen.
- A medida que aumenta el número de propiedades asociadas a una entidad concreta, también lo hace el tiempo de CPU necesario para leer y escribir la entidad.

- En la mayoría de los casos, todas las consultas son igual de eficaces, ya que todas utilizan índices. No obstante, los resultados extraídos requieren tiempo de CPU adicional.

- **Solicitudes seguras**

El número de solicitudes que emplean una conexión segura (HTTPS). Las solicitudes seguras también se contabilizan en la cuota de solicitudes.

- **Ancho de banda de salida seguro**

La cantidad de datos que envía la aplicación sobre una conexión segura en respuesta a solicitudes. El ancho de banda de salida seguro también se contabiliza en la cuota de ancho de banda de salida.

- **Ancho de banda de entrada seguro**

La cantidad de datos que recibe la aplicación sobre una conexión segura a partir de solicitudes. El ancho de banda de entrada seguro también se contabiliza en la cuota de ancho de banda de entrada.

Tabla III - XII Ancho de banda de entrada seguro Google App Engine

Recurso	Cuota gratuita predeterminada		Cuota de facturación habilitada	
	Límite diario	Tarifa máxima	Límite diario	Tarifa máxima
Solicitudes	1.300.000 solicitudes	7.400 solicitudes/minuto	43.000.000 de solicitudes	30.000 solicitudes/minuto
Ancho de banda de salida (facturable, incluye HTTPS)	10 gigabytes	56 megabytes/minuto	10 gigabytes gratuitos; 1.046 gigabytes máximo	740 megabytes/minuto
Ancho de banda de entrada (facturable, incluye HTTPS)	10 gigabytes	56 megabytes/minuto	10 gigabytes gratuitos; 1.046 gigabytes máximo	740 megabytes/minuto
Tiempo de CPU (facturable)	46 horas de CPU	15 minutos de CPU/minuto	46 horas de CPU gratuitas; 1.729 horas de CPU máximo	72 minutos de CPU/minuto

- **Almacén de Datos**

- **Llamadas al API del almacén de datos**

El número total de veces que la aplicación ha recuperado, creado, actualizado o eliminado una entidad o que ha realizado una consulta.

- **Datos almacenados**

La cantidad de datos almacenados en entidades y los índices correspondientes.

Es importante tener en cuenta que los datos almacenados en el almacén de datos pueden conllevar importantes reducciones de gastos generales. Esta reducción de gastos generales depende del número y del tipo de propiedades asociadas e incluye el espacio que utilizan los índices integrados y personalizados. Cada entidad almacenada en el almacén de datos requiere los siguientes metadatos:

- ✓ La clave de entidad, incluido el tipo, el nombre de ID o de clave y la clave de la entidad principal de la entidad.
- ✓ El nombre y el valor de cada propiedad. Desde que el almacén de datos no dispone de esquema, el nombre de cada propiedad debe almacenarse con el valor de propiedad de cualquier entidad concreta.
- ✓ Cualquier fila de índice integrado y personalizado que haga referencia a esta entidad. Cada fila contiene un tipo de entidad y cualquier número de valores de propiedad dependiendo de la definición de índice y la clave de entidad.

- **Datos enviados al API**

La cantidad de datos que se envían al almacén de datos cuando se crea o se actualiza una entidad o cuando se realiza una consulta.

- **Datos recibidos del API**

La cantidad de datos que devuelve el almacén de datos cuando se recupera una entidad o se realiza una consulta.

- **Tiempo de CPU del almacén de datos**

Es el tiempo empleado en realizar operaciones del almacén de datos. Los ciclos de CPU del almacén de datos también se contabilizan en la cuota de tiempo de CPU.

Tabla III - XIII Tiempo de CPU en el almacenamiento de datos de Google App Engine

Recurso	Cuota gratuita predeterminada		Cuota de facturación habilitada	
	Límite diario	Tarifa máxima	Límite diario	Tarifa máxima
Llamadas al API del almacén de datos	10.000.000 de llamadas	57.000 llamadas/minuto	140.000.000 de llamadas	129.000 llamadas/minuto
Datos almacenados (facturables)	1 gigabyte	<i>Ninguno</i>	1 gigabyte gratuito; no hay máximo	<i>Ninguno</i>
Datos enviados al API	12 gigabytes	68 megabytes/minuto	72 gigabytes	153 megabytes/minuto
Datos recibidos del API	115 gigabytes	659 megabytes/minuto	695 gigabytes	1.484 megabytes/minuto
Tiempo de CPU del almacén de datos	60 horas de CPU	20 minutos de CPU/minuto	1.200 horas de CPU	50 minutos de CPU/minuto

- **Correo**

- **Llamadas al API de correo**

El número total de veces que la aplicación accedió al servicio de correo para enviar un mensaje de correo electrónico.

- **Destinatarios de mensajes de correo electrónico**

El número total de destinatarios a los que la aplicación ha enviado mensajes de correo electrónico.

- **Administradores destinatarios de mensajes de correo electrónico**

El número total de administradores de la aplicación a los que la aplicación ha enviado mensajes de correo electrónico.

- **Datos del cuerpo del mensaje enviados**

La cantidad de datos enviados en el cuerpo de los mensajes de correo electrónico. Esto también se contabiliza en la cuota de ancho de banda de salida.

- **Archivos adjuntos enviados**

El número total de archivos adjuntos enviados con los mensajes de correo electrónico.

- **Datos de archivo adjunto enviados**

La cantidad de datos enviados como archivos adjuntos en los mensajes de correo electrónico. Esto también se contabiliza en la cuota de ancho de banda de salida.

Tabla III - XIV Datos de archivo adjunto enviados en Google App Engine

Recurso	Cuota gratuita predeterminada		Cuota de facturación habilitada	
	Límite diario	Tarifa máxima	Límite diario	Tarifa máxima
Llamadas al API de correo	7.000 llamadas	32 llamadas/minuto	1.700.000 llamadas	4.900 llamadas/minuto
Destinatarios de mensajes de correo electrónico (facturables)	2.000 destinatarios	8 destinatarios/minuto	2.000 destinatarios gratuitos; 7.400.000 destinatarios máximo	5.100 destinatarios/minuto
Administradores destinatarios de mensajes de correo electrónico	5.000 correos	24 correos/minuto	3.000.000 de correos	9.700 correos/minuto
Datos del cuerpo del mensaje enviados	60 megabytes	340 kilobytes/minuto	29 gigabytes	84 megabytes/minuto
Archivos adjuntos enviados	2.000 archivos adjuntos	8 archivos adjuntos/minuto	2.900.000 archivos adjuntos	8.100 archivos adjuntos/minuto
Datos de archivo adjunto enviados	100 megabytes	560 kilobytes/minuto	100 gigabytes	300 megabytes/minuto

- **Extracción de URL**

- **Llamadas al API de extracción de URL**

El número total de veces que la aplicación accedió al servicio de extracción de URL para realizar una solicitud HTTP o HTTPS.

- **Datos de extracción de URL enviados**

La cantidad de datos enviados al servicio de extracción de URL a modo de solicitudes. Esto también se contabiliza en la cuota de ancho de banda de salida.

- **Datos de extracción de URL recibidos**

La cantidad de datos recibidos del servicio de extracción de URL en forma de respuestas. Esto también se contabiliza en la cuota de ancho de banda de entrada.

Tabla III - XV Datos de extracción de URL recibidos en Google App Engine

Recurso	Cuota gratuita predeterminada		Cuota de facturación habilitada	
	Límite diario	Tarifa máxima	Límite diario	Tarifa máxima
Llamadas al API UrlFetch	657.000 llamadas	3.000 llamadas/minuto	46.000.000 de llamadas	32.000 llamadas/minuto
Datos UrlFetch enviados	4 gigabytes	22 megabytes/minuto	1.046 gigabytes	740 megabytes/minuto
Datos UrlFetch recibidos	4 gigabytes	22 megabytes/minuto	1.046 gigabytes	740 megabytes/minuto

- **Manipulación de Imágenes**

- **Llamadas al API de manipulación de imágenes**

El número total de veces que la aplicación accede al servicio de imágenes.

- **Datos enviados al API**

La cantidad de datos enviados al servicio de imágenes.

- **Datos recibidos del API**

La cantidad de datos recibidos del servicio de imágenes.

- **Transformaciones ejecutadas**

El número de veces que el servicio ha ejecutado una transformación de imágenes para la aplicación.

Tabla III - XVI Transformaciones ejecutadas de Imágenes en Google App Engine

Recurso	Cuota gratuita predeterminada		Cuota de facturación habilitada	
	Límite diario	Tarifa máxima	Límite diario	Tarifa máxima
Llamadas al API de manipulación de imágenes	864.000 llamadas	4.800 llamadas/minuto	45.000.000 de llamadas	31.000 llamadas/minuto
Datos enviados al API	1 gigabyte	5 megabytes/minuto	560 gigabytes	400 megabytes/minuto
Datos recibidos del API	5 gigabytes	28 megabytes/minuto	427 gigabytes	300 megabytes/minuto
Transformaciones ejecutadas	2.500.000 transformaciones	14.000 transformaciones/minuto	47.000.000 de transformaciones	32.000 transformaciones/minuto

- **Memcache**

- **Llamadas al API de Memcache**

El número total de veces que la aplicación accedió al servicio Memcache para obtener, establecer o desactivar valores.

- **Datos enviados al API**

La cantidad de datos enviados a Memcache

- **Datos recibidos del API**

La cantidad de datos recibidos de Memcache.

Tabla III - XVII Datos recibidos del API en Google App Engine

Recurso	Cuota gratuita predeterminada		Cuota de facturación habilitada	
	Límite diario	Tarifa máxima	Límite diario	Tarifa máxima
Llamadas al API de Memcache	8.600.000	48.000 llamadas/minuto	96.000.000	108.000 llamadas/minuto
Datos enviados al API	10 gigabytes	56 megabytes/minuto	60 gigabytes	128 megabytes/minuto
Datos recibidos del API	50 gigabytes	284 megabytes/minuto	315 gigabytes	640 megabytes/minuto

- **Implementaciones**

El número de veces que un desarrollador ha subido la aplicación.

3.3.2.5. Sistema Operativo

Los sistemas operativos que se encuentran actualmente disponibles son:

Tabla III - XVIII Sistema Operativo de Google App Engine

Sistema Operativo
Windows
Mac OSX
Linux

3.3.2.6. Software

Entre los principales software que soporta Google App Engine son:

Tabla III - XIX Software Google App Engine

Software	
Java	Eclipse
Python	WebApp
Base de Datos	
Bigtable	

3.3.2.7. Precios

Al habilitar la facturación de la aplicación, las cuotas fijas de la aplicación aumentarán.

Tabla III - XX Precios Google App Engine

Recurso	Unidad	Coste de la unidad
Ancho de banda de salida	gigabytes	0,12 dólares
Ancho de banda de entrada	gigabytes	0,10 dólares
Tiempo de CPU	horas de CPU	0,10 dólares
Datos almacenados	gigabytes al mes	0,15 dólares
Destinatarios de mensajes de correo electrónico	destinatarios	0,0001 dólares

3.4. DETERMINACIÓN DE LOS PARÁMETROS DE COMPARACIÓN

Para la comparación de los modelos se ha limitado a analizar la usabilidad. Los parámetros elegidos para la comparación han sido determinados analizando las características más relevantes de cada modelo, además con esta comparación es posible realizar un análisis cualitativo y cuantitativo lo cual permitirá reflejar un resultado claro y conciso.

Los parámetros analizados establecen aspectos que ayudan al desarrollo de la aplicación, como por ejemplo la determinación del Lenguaje de Programación y la base de datos. Además permiten elegir Ambientes de desarrollo y Sistemas Operativos ya que los dos modelos ofrecen varias alternativas.

Con respecto a los Costos y a la Configuración los modelos de hosting difieren en gran escala, por lo que se han analizado cada uno de estos con el fin de determinar el resultado que más favorezca al desarrollo de la aplicación.

3.4.1. Lenguajes de Programación

Google y Amazon brindan una variedad de lenguajes en las que se puede desarrollar la aplicación entre los principales están:

AMAZON EC2

- Visual Basic
- C#
- Perl
- Php
- Java
- Python

GOOGLE APP ENGINE

- Python
- Java

Tabla III - XXI Comparación entre Python y Java

	PYTHON	JAVA
Definición de tipos de dato	Dinámico: En Python nunca se declara ningún tipo de dato, al momento de la asignación la sentencia incluye el tipo de dato que maneja el objeto. Esto implica que luego el tipo de dato puede ser reasignado por lo tanto se puede decir que Python es un lenguaje de tipos de datos dinámicos. Los objetos contenedores en Python pueden almacenar objetos de cualquier tipo incluyendo números y listas.	Estático: En Java todos los nombres de variables deben estar explícitamente declarados. Intentar asignar a un objeto el tipo equivocado lanza una excepción de tipo de dato de esta manera Java es un lenguaje de tipos de datos estáticos. Los contenedores de objetos en Java pueden almacenar tipos genéricos pero no almacenar tipos de datos primitivos.
Lenguaje	Extenso: abundante en palabras; usa o contiene más palabras de las necesarias.	Conciso: expresa mucho en pocas palabras. Lenguaje breve alcanzando esto a través de la supresión de palabras irrelevantes.
Estructura	Compacto: Describe la propiedad para el diseño de la aplicación indicando la facilidad de manejo de errores lo cual indica que es una herramienta compacta.	No Compacto.
Preciso	A nivel de clases en etapas de desarrollo, no existe una regla explícita en cuanto al manejo de archivos con sus respectivas clases de esta manera se puede almacenar toda una aplicación con una cantidad de clases en un archivo único.	En las etapas de desarrollo Java define como una regla que cada clase debe estar en su propio archivo lo cual mejora el manejo de grandes aplicaciones ya que organiza la información de una manera más precisa.
Técnico	Posee una característica de control en cuanto a su estructura en el lenguaje, manejando reglas y parámetros obligatorios dejando al programador sin la libertad en la codificación.	El lenguaje posee características mínimas para el control en la estructura de la codificación, no existen parámetros definidos que limiten al programador en el desarrollo de la aplicación,
Plataformas	No lo hace	Crea aplicaciones que funcionan a través de distintas plataformas
Multihebra	Programación Multihebra regular	Permite una buena programación Multihebra es decir, Conmuta entre distintas partes del código de un mismo programa durante la ejecución.

Debido a estos parámetros de comparación hemos decidido realizar la programación de nuestra aplicación en Java.

3.4.2. Sistema Operativo

Un Sistema Operativo (SO) provee una interfaz entre el resto de programas del ordenador, los dispositivos hardware y el usuario. Entre los que soportan Amazon EC2 y Google App Engine son los siguientes:

AMAZON EC2

- Red Hat Enterprise Linux
- OpenSolaris
- Fedora
- Windows Server 2003/2008
- openSUSE Linux
- Gentoo Linux
- Oracle Enterprise Linux
- Ubuntu Linux
- Debian

GOOGLE APP ENGINE

- Windows
- Linux
- OS X

3.4.3. Bases de Datos

Las Bases de datos son muy relevantes ya que sirve para almacenar grandes cantidades de información entre las que nos brindan Amazon EC2 y Google App Engine tenemos:

AMAZON EC2

- IBM DB2
- IBM Informix Dynamic Server
- Microsoft SQL Server Standard 2005
- MySQL Enterprise
- Oracle Database 11g

GOOGLE APP ENGINE

- BigTable

3.4.4. Ambientes de Desarrollo

3.4.4.1. Eclipse

Eclipse es una potente y completa plataforma de programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java.

Se trata de un entorno de desarrollo integrado (IDE) en el que se encuentra todas las herramientas y funciones necesarias para nuevas aplicaciones, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar.

3.4.4.1.1. Características

Eclipse dispone de un Editor de texto con resaltado de sintaxis. La compilación es en tiempo real. Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes (*wizards*) para creación de proyectos, clases, tests, etc., y refactorización.

Asimismo, a través de "plugins" libremente disponibles es posible añadir control de versiones con subversión e integración con Hibernate.

3.4.4.1.2. Plugins

Subclipse es un plugin para el entorno de desarrollo Eclipse, nos proporciona una serie de herramientas mediante las cuales podremos interactuar con un sistema de control de versiones Subversión

Algunas de las funcionalidades más destacadas que nos ofrece subclipse son las siguientes:

- Crear un módulo en el repositorio a partir de un proyecto
- Crear un proyecto a partir de un módulo del repositorio
- Mantenimiento de ficheros:
 - Sincronización de ficheros con el repositorio
 - Actualizar ficheros (*update*)
 - Guardar nuevos cambios en un fichero (*commit*)
 - Asignar etiquetas (*tag*)
 - Crear copias (*branch*)

Las responsabilidades de Subclipse se centran en interactuar con sistemas de control de versiones Subversión desde el entorno de desarrollo Eclipse, permitiendo al usuario tener sus proyectos en un repositorio común al equipo de desarrollo.

A partir de ese punto, el equipo de trabajo podrá trabajar con dichos proyectos, usando las funcionalidades que los sistemas de control de versiones nos ofrecen.

CheckStyle contiene una serie de ficheros de propiedades con mensajes de error. Su formato es *messages_xx.properties*. Como se puede ver, vienen preparados para la internacionalización.

PMD es una herramienta de calidad de código, con calidad nos referimos a cumplir con estándares de codificación cuyo objetivo es el de garantizar que las aplicaciones sean desarrolladas bajo el mismo concepto y con ello facilitar el mantenimiento.

El componente PMD nos permite analizar el código fuente de los proyectos. Se pretende, por tanto, identificar posibles problemas potenciales en el código desarrollado. PMD nos permite detectar:

- Posibles *bugs*: bloques *try/catch* o bloques *if/else* que estén vacíos.
- Código *muerto*: variables locales que no se usan, análisis de variables y métodos privados.
- Código no óptimo: malos usos de objetos tipo *String* o *StringBuffer*.
- Complejidad de código: sentencias *if* innecesarias, bucles *for* que podrían transformarse en un bucle *while*.

PMD cuantifica una métrica muy importante en el análisis de código: la complejidad ciclomática, que es la complejidad del código obtenida del número de rutas linealmente independientes; indica el número de casos de prueba unitarios básicos para obtener una cobertura al 100%, es decir, proporciona un límite máximo para el número de pruebas a realizar para asegurar que se ejecuta cada sentencia al menos una vez.

FindBugs es una herramienta Opensource (licencia LGPL) que realiza un análisis de código estático en *bytecode*, es decir, PMD analiza el código fuente y FindBugs los ficheros *.class*

De acuerdo a sus autores, esta herramienta está enfocada a buscar bugs

reales, es decir no da importancia al estilo y al formato sino a errores de programación potenciales.

En combinación con PMD, esta herramienta permitirá mantener una buena calidad en el código de nuestra aplicación, asegurando así una mayor mantenibilidad y disminuir errores inesperados en tiempo de ejecución. Puede utilizarse como un plug-in de Eclipse o con una interfaz gráfica propia.

FindBugs tiene responsabilidades de Análisis de código estático en *bytecode* y Proporciona recomendaciones a errores de programación potenciales.

Spring IDE for Eclipse es un plug-in para Eclipse 3.x que proporciona un interfaz gráfica para el usuario y algunas facilidades para la manipulación de los diferentes ficheros de configuración de Spring. Este plug-in también proporciona soporte para Spring Web Flow.

Spring IDE para Eclipse es un plug-in que proporciona una serie de vistas, editores y validadores para la manipulación de ficheros Spring. Permite:

- Incorporar a un proyecto las funcionalidades de Spring de forma sencilla lo cual permite soportar una lista *de* ficheros de configuración de Spring y conjuntos de ficheros de configuración. Una de las ventajas de esto, es que se puede crear un conjunto de configuración para las pruebas y otro para la ejecución de la aplicación. Estos conjuntos son usados por el *BeansConfigValidator* para resolver las referencias a otros beans. En relación a esto, un conjunto de configuración también puede contener varios ficheros de configuración definidos en otros proyectos Spring.
- Visualizar gráficamente la dependencia entre distintos Beans gestionados por Spring.
- Vista jerárquica de dichas dependencias.

- Vista donde se muestra un árbol con todos los proyectos de Spring y sus ficheros de configuración.
- Validación de la coherencia interna de los ficheros de Spring definidos en el proyecto. Esta validación se hace mediante las llamadas realizadas por el *BeansProjectValidator* por cada modificación en los ficheros de configuración.
- Editor XML para los ficheros de configuración de Spring entre los que ahora se incluye también los elementos de Spring Security 2.
- Extensión de la búsqueda de Eclipse de modo que se puede realizar una búsqueda de beans (ya sea por el nombre, por una cierta propiedad, por una clase...).
- Wizard para crear un nuevo proyecto que tenga las funcionalidades de Spring

Maven ofrece las distintas funcionalidades necesarias para el ciclo de vida de un desarrollo mediante plugins, además de ofrecer la posibilidad de desarrollar plugins propios y entre sus responsabilidades tiene:

- Construir el tipo de artefacto que se necesite: .jar , .war, .ejb, .ear , .sar
- Controlar dependencias de librerías externas como log4j, struts, etc. poniendo especial atención en la versión de cada una.
- Facilitar el control de dependencias internas como un jar que depende de otro jar.

- Utilizar procedimientos por defecto para la realización de las tareas de compilación y empaquetado, evitando así scripts hechos para cada proyecto, complejos y poco mantenibles.
- Disminuir la curva de aprendizaje de los nuevos recursos, en toda la vida del proyecto van integrándose programadores en el tiempo de desarrollo y las tareas ya automatizadas no deberían de significar problema alguno para ellos.

3.4.4.2. Netbeans

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados *módulos*. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos.

3.4.4.2.1. Características

La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas)
- Administración de las configuraciones del usuario
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato)
- Administración de ventanas
- Framework basado en asistentes (diálogos paso a paso)

3.4.4.2.2. Plugins

Special copy paste permite seleccionar código fuente de las clases, y copiarlo con el formato HTML, es decir que podemos copiar fragmentos desde el editor de NetBeans directamente a sitios como foros, blogs, etc, y se reflejará exactamente el formato (tamaño, color, tipo de letra).

PDF Viewer

Utiliza clases propias de Java (Sin utilizar el Acrobat) para poder leer algún pdf.

Copy & Paste History

Retomando aquello de copiar y pegar, este plugin guarda un histórico de varias copias que hayamos realizado con anterioridad, incluso después de cerrar el IDE y apagar el equipo; Luego de instalarlo se puede configurar el número de históricos que guarde.

3.4.4.3. Comparación entre Eclipse y Netbeans

Tabla III - XXII Comparación entre Eclipse y Netbeans.

	ECLIPSE	NETBEANS
Refactorización	Tiene un buen soporte de refactorización	Carece de refactorización
Webapps (.war, jsp y servlets)	Carece de webapps, los plugins que posee no son tan potentes	Tiene un buen soporte de webapps
Plugins	Es necesario instalar y configurar para su utilización	Se encuentran instalados listos para utilizarse.
Recursos	Utiliza menos recursos	Utiliza más recursos
Comprobación de sintaxis, completado de código y pegado de código.	Si soporta	Si soporta
Compilar, ejecutar y depurar el código	Si soporta	Si soporta
Soporte para JSF	Soporta y se puede utilizar a través de una forma gráfica	Soporta pero se debe hacer configuraciones manuales
Diseñador GUI Visual	Soporta para varios tipos de tecnologías	No posee un diseñador grafico
Generación de Métodos set y get	Soporta y proporciona una opción para generar automáticamente	Soporta pero se debe realizar de una forma manual
Formateo de Fechas	Soporta en la misma pagina	Soporta en la misma pagina

3.4.5. Servidor de Aplicaciones

Servidor de aplicaciones es servidor en una red de computadores que ejecuta ciertas aplicaciones, generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación.

Entre los servidores de aplicaciones de AMAZON EC2 tenemos:

- IBM WebSphere Application Server, WAS está construido usando estándares abiertos tales como J2EE, XML, y Servicios Web, sirve para generar, implementar y administrar aplicaciones SOA robusta, ágil y reutilizables de negocios y servicios de todo tipo y para reducir los costes de infraestructura de aplicaciones de IBM WebSphere Application Server, Esto funciona con varios servidores web incluyendo Apache HTTP Server, Netscape Enterprise Server entre otros.
- Java Application Server, El servidor está basado en la plataforma Java EE y es el núcleo del sistema Java Enterprise, a manera de contenedor de EJB o de proveedor de Web Service. Tiene soporte integrado para interfaces de desarrollo tales como Sun Java Studio Enterprise, Sun Java Studio Creator y NetBeans.
- Oracle WebLogic Server, WebLogic puede utilizar Oracle, DB2, Microsoft SQL Server, y otras bases de datos que se ajusten al estándar JDBC. El servidor WebLogic es compatible con WS-Security y cumple con los estándares de J2EE

Entre los servidores de aplicaciones de GOOGLE APP ENGINE tenemos:

- BigTable, Google creó Bigtable para que fuese, sobre todo, una base de datos en la que se almacenaría una cantidad de información enorme, del orden de Petabytes. Para ello, cada tabla está dividida en tablets que

pueden llegar a ocupar 200 Megabytes. Si superasen ese tamaño serían automáticamente divididas y comprimidas para ser enviadas a más máquinas usando un sistema de compresión propietario de Google

- Google File System, es un sistema de almacenamiento basado en las necesidades de Google diseñado por Sanjay Ghemawat, Howard Gobioff y Shun-Tak Leung. Al no ser un sistema de archivos de uso generalista, GFS, ha sido diseñado teniendo en cuenta las siguientes premisas: que un componente falle es la norma no la excepción, los archivos son enormes (archivos de muchos GB son comunes), es muy común que un archivo cambie porque se le añaden datos pero es muy raro que se sobrescriban los datos existentes, el codiseño de las aplicaciones y de la API del sistema de archivos proporciona un beneficio global.

3.4.6. Costo

Entre ambos modelos de hosting existe una gran diferencia en los costos, debido a que Amazon tiene sus costos establecidos por diferentes parámetros mientras que google hasta cierto punto es gratuito.

AMAZON EC2

- Demanda de instancias
- Reserva de instancias
- Spot Instances
- Transferencia de datos
- Block Store
- Direcciones IP
- Amazon CloudWatch
- Equilibrio de carga elástica

GOOGLE APP ENGINE

- Cuotas facturables y fijas
- Renovación de recursos
- Cuotas por minuto
- Reducción de un recurso
- Recursos

3.4.7. Configuración

3.4.7.1. Amazon EC2

Para establecer la configuración en Amazon EC2, se debe seguir los pasos a continuación:

Ingresar a la página oficial de Amazon EC2 <http://aws.amazon.com/>.



Figura III - 1 Página Oficial de Amazon EC2

Se crea una cuenta en Amazon Web Services.

amazon web services™ Create an Amazon Web Services Account

1 SIGN IN 2 ACCOUNT INFO 3 SUCCESS

Amazon Web Services Sign In

You may sign in using your existing Amazon account or you can create a new account by selecting "I am a new user."

My e-mail address is:

I am a new user.

I am a returning user and my password is:

[Sign in using our secure server](#)

[Forgot your password?](#)

[Has your e-mail address changed?](#)

Figura III - 2 Create an Amazon Web Service Account. Amazon Web Service Sing In

amazon web services™ Create an Amazon Web Services Account

1 SIGN IN 2 ACCOUNT INFO 3 SUCCESS

Registration

New to Amazon.com? Register Below.

My name is:

My e-mail address is:

Type it again:

Protect your information with a password
This will be your only Amazon.com password.

Enter a new password:

Type it again:

[Create account](#)

Figura III - 3 Create an Amazon Web Service Account. Registration

Country*: Ecuador

Phone number*: 087730033

Nombre de la empresa o sitio Web: SGP

URL del sitio Web: www.sigpro.com

Contrato de cliente de AWS [Versión para impresora](#)

SERVICE™ ARE NOW LOCATED AT: <http://affiliate-program.amazon.com/gp/advertising/ap/detail/agreement.html>

PLEASE READ CAREFULLY — THIS IS A BINDING CONTRACT

THIS AWS CUSTOMER AGREEMENT ("AGREEMENT" OR "AMAZON WEB SERVICES CUSTOMER AGREEMENT") IS A BINDING AGREEMENT BETWEEN AMAZON WEB SERVICES LLC ("AWS") AND YOU AND, IF APPLICABLE, THE COMPANY OR OTHER LEGAL ENTITY YOU REPRESENT (COLLECTIVELY, "YOU"). THIS AGREEMENT INCORPORATES BY REFERENCE (1) THE PRIVACY NOTICE POSTED ON WWW.AMAZON.COM ("PRIVACY NOTICE"), (2) THE TERMS OF USE POSTED ON AWS.AMAZON.COM ("TERMS OF USE"), (3) THE ACCEPTABLE USE POLICY POSTED ON

Check here to indicate that you have read and agree to the terms of the Amazon Web Services Customer Agreement.

Security Check

Imagen: 

[Try a different image](#) [¿Por qué le pedimos que escriba estos caracteres?](#)

Escriba los caracteres de la imagen anterior*: 7C61JN

[¿Problemas? Hágase en contacto con nosotros.](#)

[Continuar](#)

Figura III - 4 Formulario de datos y Contrato de cliente de AWS

Bienvenido Sistema de Gestion de Proyectos | [¿No es usted?](#)

Thank You
You Have Created an Amazon Web Services Account

We have e-mailed you a confirmation and information about your new account.

Explorar otros servicios de Amazon Web Services

- Amazon Mechanical Turk: Regístrese en el [sitio Web de solicitantes](#) de Amazon Mechanical Turk para utilizar el [servicio Web Amazon Mechanical Turk](#)
- Amazon CloudFront
- Amazon Elastic Compute Cloud
- Amazon Elastic MapReduce
- Amazon DevPay
- Amazon Flexible Payments Service
- Amazon Fulfillment Web Service
- AWS Import/Export
- Amazon Relational Database Service
- Amazon Simple Storage Service
- Amazon SimpleDB
- Amazon Simple Notification Service
- Amazon Simple Queue Service
- Amazon Virtual Private Cloud

Para descubrir todos los recursos y la información que AWS le ofrece, [visite la página de inicio de Amazon Web Services.](#)

Figura III - 5 Envío de Notificación al e-mail

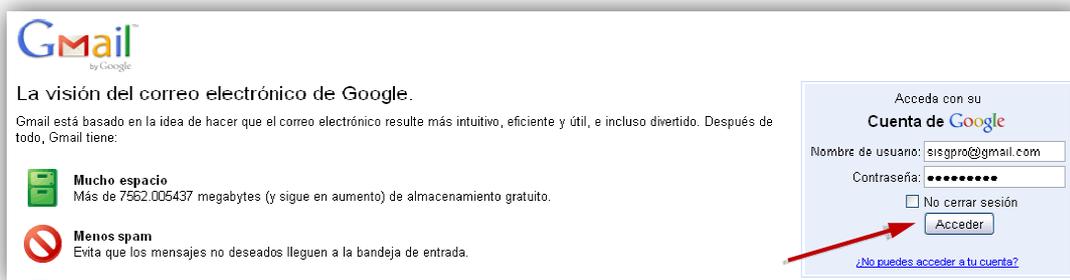


Figura III - 6 Acceso a e-mail

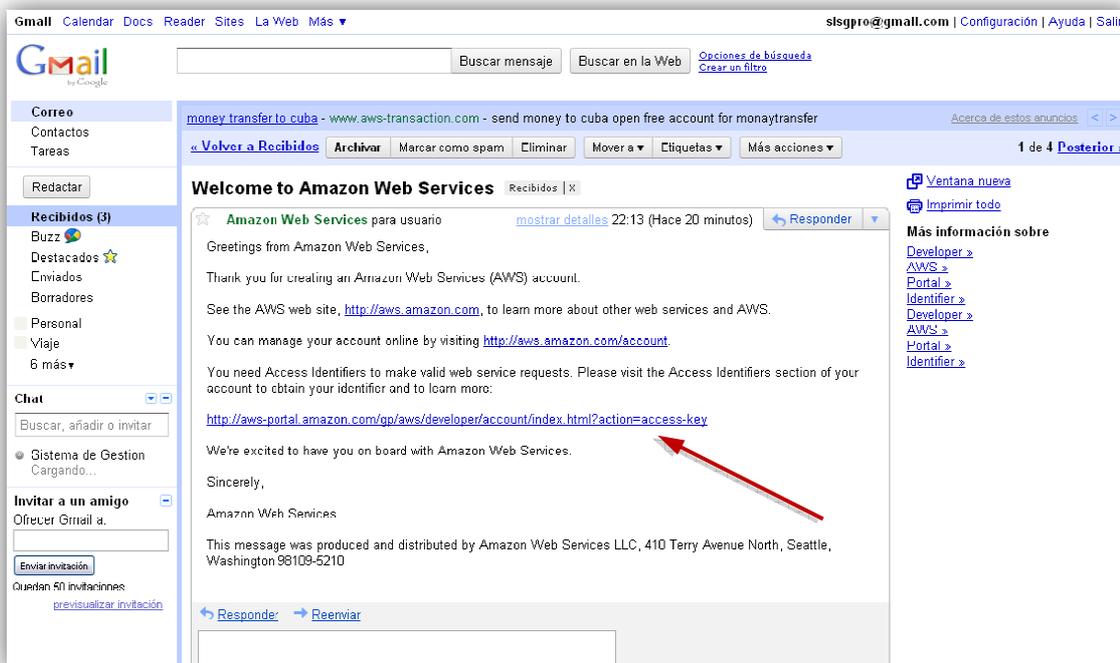


Figura III - 7 Mail de confirmación de la cuenta



Figura III - 8 Amazon Web Service Sing In

En la siguiente pantalla se muestran detalladamente los costos por la creación de instancias o proyectos. En la misma se debe ingresar la forma de pago.

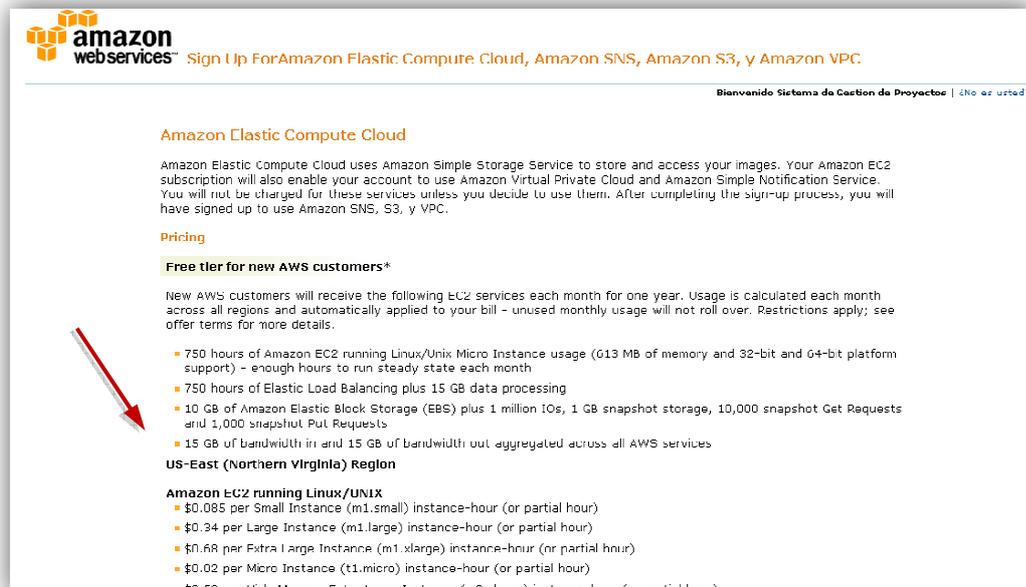


Figura III - 9 Información de Costos en Amazon

Ingresa y una vez dentro, se selecciona la pestaña de Amazon EC2. Esto es como un panel de control para todos los servicios en Amazon Web Services.

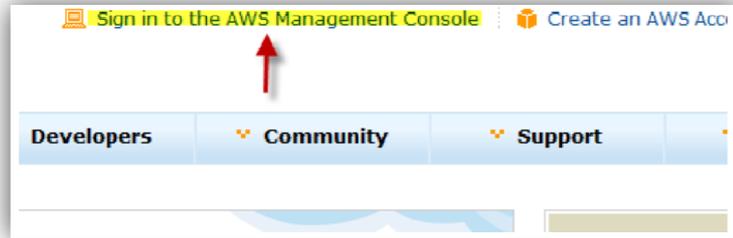


Figura III - 10 Consola de Gestión de AWS

A continuación se debe lanzar una nueva instancia haciendo clic en el botón Launch Instance, esta instancia es básicamente un VPS. Es un sistema virtualizado sobre la nube de Amazon Web Services y se comporta como una computadora cualquiera.

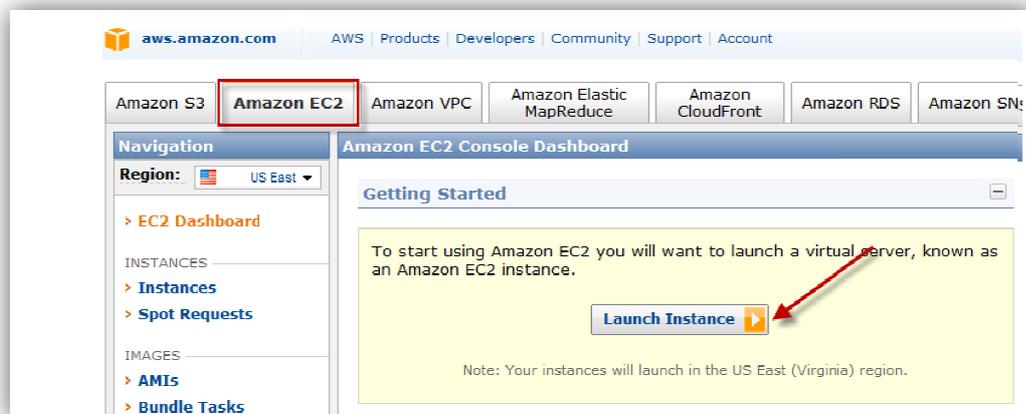


Figura III - 11 Lanzamiento de Nueva Instancia

Se selecciona el sistema operativo que se correrá en la instancia. Amazon se refiere a "discos de instalación" como AMIs. Algunas son mantenidas por Amazon, estas son las Amazon Linux AMI y están muy bien adaptadas para el medioambiente a el que se expondrán en EC2.

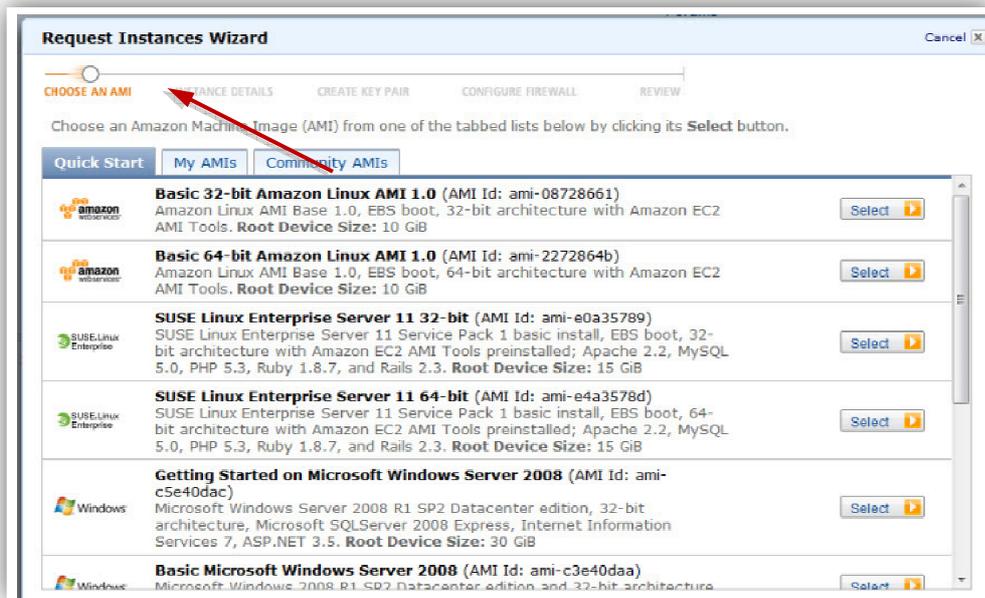


Figura III - 12 Requisitos de la Instancia. Choose an AMI

Ir a alestic.com y copiar el identificador de la AMI. Es recomendable usar: ami-508c7839.



Figura III - 13 Cummunity AMIs

En la pantalla de INSTANCE DETAILS, se selecciona el tipo de instancia que deseamos crear. Hacer clic en Continue hasta llegar a la pantalla llamada CREATE KEY PAIR.

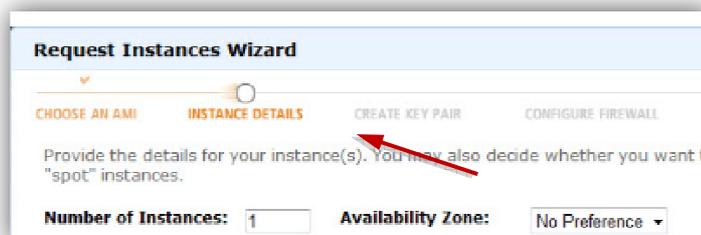


Figura III - 14 Requisitos de la Instancia. Instance Details

Una vez en CREATE KEY PAIR seleccionar la segunda opción llamada Create a new Key Pair para crear un nuevo set de llaves y conectarse al servidor vía SSH. Se asigna un nombre cualquiera y luego se puede descargar haciendo clic en *Create & Download your Key Pair*. Esto va a devolver un archivo .pem. Esta es la única manera de iniciar sesión en el servidor por lo que es importante guardar y no compartir ya que es el único requisito para iniciar sesión debido a que no hay contraseñas requeridas.

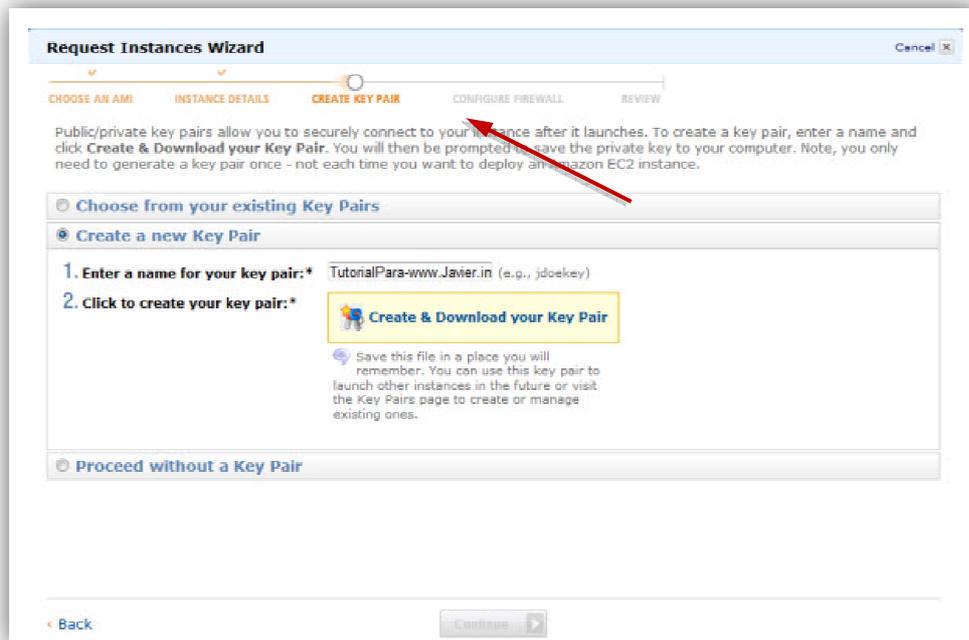


Figura III - 15 Requisitos de la Instancia. Create Key Pair

Amazon Ec2 pone su propio firewall entre el servidor y el Internet. Para poder hacer que el servidor pueda comunicarse con el mundo exterior hay que crear un *Security Group* y abrir en el los puertos necesarios. Durante la instalación seleccionar default luego ir a abrir los puertos necesarios. Clic en Continue.

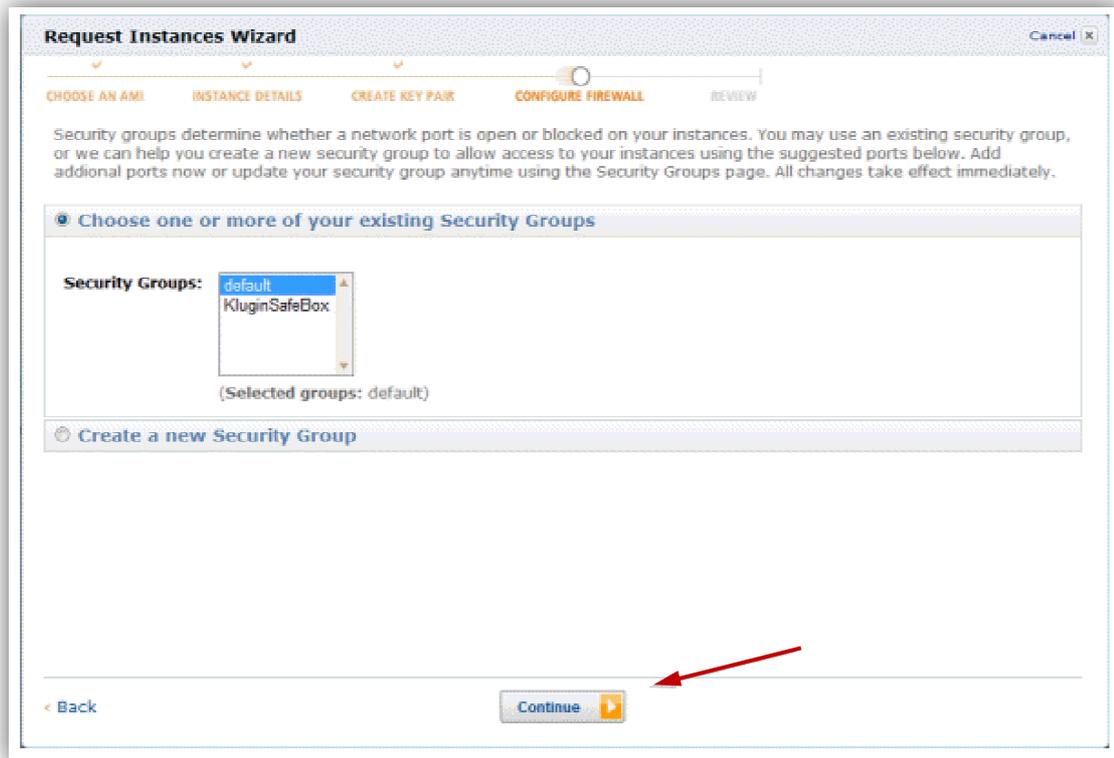


Figura III - 16 Requisitos de la Instancia. Configure Firewall

Finalmente, la instancia está lista para ser creada, hacer clic en Launch para que Amazon comience a crearla.

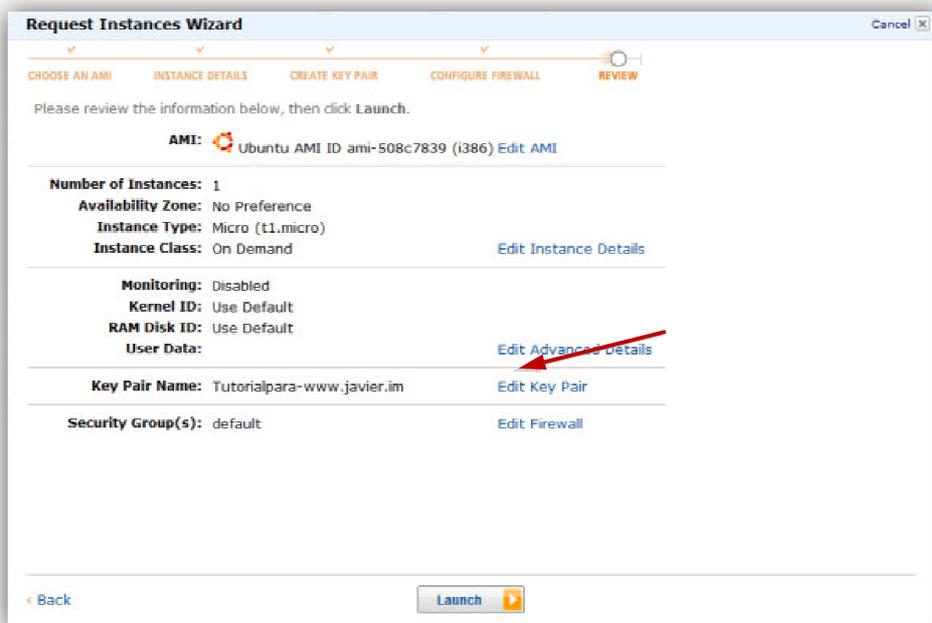


Figura III - 17 Requisitos de la Instancia. Review

Se debe esperar que pase a estar en estado “running” antes de intentar iniciar sesión.

Como la instancia fue configurada para usar el grupo *default* es necesario abrir una serie de puertos. Los *Security Groups* es como Amazon permite que el mundo exterior toque nuestro server.

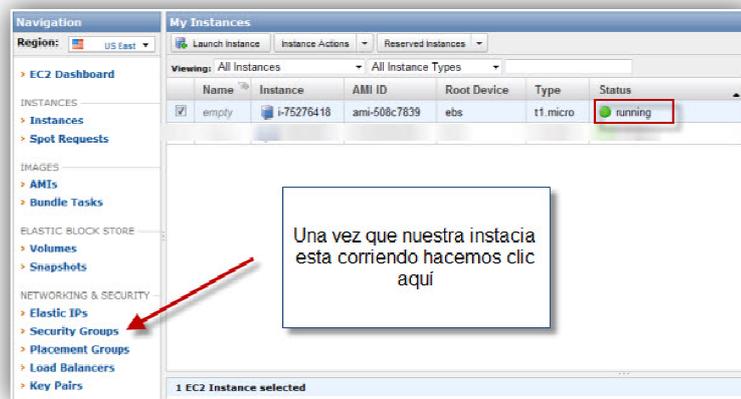


Figura III - 18 Navigation. Security Groups

Establecer las configuraciones necesarias.

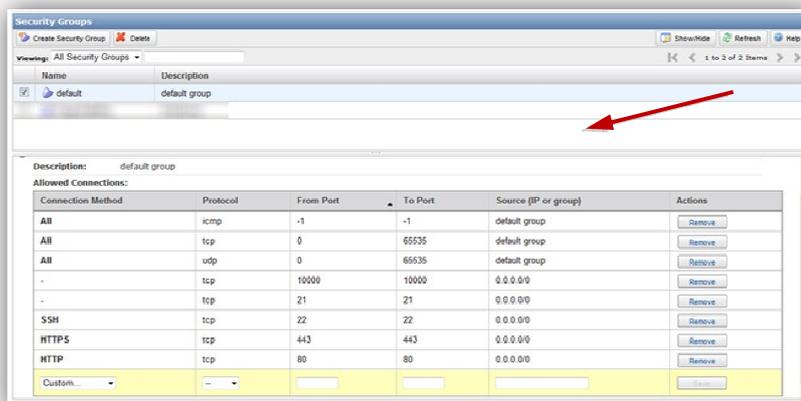


Figura III - 19 Security Groups

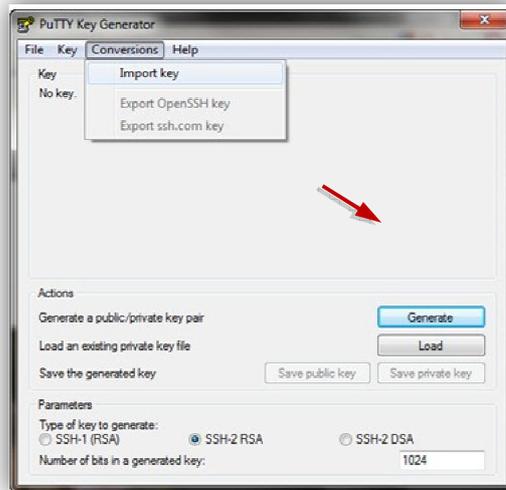


Figura III - 20 PuTTY Key Generator

Si se utiliza Linux o Mac no es necesario hacer esto ya que el archivo .pem es compatible con la terminal. Pero cuando se usa PuTTY en Windows hay que convertir la llave a un formato compatible con PuTTY usando PuTTYgen. Importar la llave .pem y guardar a la llave privada haciendo clic en el botón *Save Private Key*. Esto devuelve un archivo .ppk que se usara para iniciar sesión en el servidor por medio de PuTTY.

Se conecta haciendo clic en Open

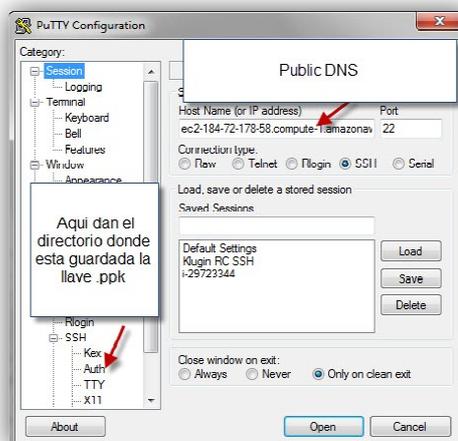
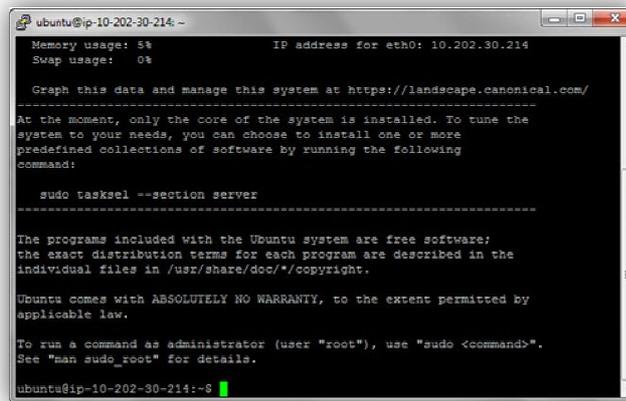


Figura III - 21 PuTTY Configuration



```
ubuntu@ip-10-202-30-214:~$
Memory usage: 5%      IP address for eth0: 10.202.30.214
Swap usage:  0%

-----
Graph this data and manage this system at https://landscape.canonical.com/
-----
At the moment, only the core of the system is installed. To tune the
system to your needs, you can choose to install one or more
predefined collections of software by running the following
command:

  sudo tasksel --section server
-----

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-202-30-214:~$
```

Figura III - 22 Ejecución de Comandos

Ejecutar los comandos dentro de la ventana de PuTTY

Desde PuTTY podemos trabajar como el servidor remoto.

Actualizar Ubuntu y todos sus paquetes. Esto toma un poco de tiempo.

Primero ponemos los repositorios al día. "sudo aptitude update".

Descargar e instalar los paquetes más recientes. "sudo aptitude upgrade".

Luego instalar los componentes del servidor LAMP. "sudo tasksel".

Activar la opción LAMP Server en la pantalla. Para activarla hay que seleccionan y presionar la tecla espacio luego enter.

MySQL solicitará un password para root, ingresar y tomar en cuenta para iniciar sesión en phpmyadmin.

Luego de eso el servidor está listo.

3.4.7.2. Google App Engine

Para establecer la configuración en Google App Engine, se debe seguir los pasos a continuación.

Ingresar a la página oficial de google <http://www.google.com>



Figura III - 23 Página Oficial de Google

Acceder y crear una cuenta en Google App Engine



Figura III - 24 Creación de una nueva cuenta Gmail

Llenar el formulario con los datos del usuario y aceptar las condiciones de uso.

ofrece una experiencia más personalizada de Google, con recomendaciones y resultados de búsqueda más relevantes respecto a sus consultas. [Más información](#)
 Habilitar Historial web

Empezar a utilizar Google App Engine

Ubicación: Ecuador

Cumpleaños: 19/12/83
DD/MM/AAAA (por ejemplo, "16/03/2011")

Verificación de la palabra: Escribe los caracteres que veas en la imagen siguiente.
ticurecci
ticurecci
No se distingue entre mayúsculas y minúsculas.

Condiciones del servicio: Compruebe la información de la cuenta de Google que ha introducido anteriormente (puede cambiar todos los datos que desee) y consulte las Condiciones del servicio que aparecen a continuación. [Versión imprimible](#)

Condiciones de servicio de Google
Le damos la bienvenida a Google
1. Su relación con Google

Al hacer clic en el botón 'Acepto' que aparece a continuación, acepta tanto los [Términos de servicio](#) anteriores como la [Política de privacidad](#).

Figura III - 25 Aceptación de la nueva cuenta

Una vez que iniciada la cuenta en Google App Engine crear una nueva aplicación.

Google app engine sigpro@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

Welcome to Google App Engine

Before getting started, you want to learn more about developing and deploying applications. Learn more about Google App Engine by reading the [Getting Started Guide](#), the [FAQ](#), or the [Developer's Guide](#).

© 2008 Google | [Terms of Service](#) | [Privacy Policy](#) | [Blog](#) | [Discussion Forums](#)

Figura III - 26 Welcome to Google App Engine. Crear Aplicación

Al momento de crear una nueva Aplicación va a solicitar un número móvil al cual llegará un código de confirmación de cuenta.

Google app engine sisgpro@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

Verify Your Account by SMS

To create applications with Google App Engine, you need a verification code. Select the country and carrier for your mobile phone and enter your mobile phone number. The verification code will be sent to it via SMS. Note you will only need to verify your account once.

Country and Carrier:

If your country and carrier are not on the list, select Other (Not Listed). [What carriers are supported?](#)

Mobile Number:

Include your [country code](#) and full phone number. eg. +1 650 555 1212

© 2008 Google | [Terms of Service](#) | [Privacy Policy](#) | [Blog](#) | [Discussion Forums](#)

Figura III - 27 Verify Your Account by SMS

Ingresar el código.

Google app engine sisgpro@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

An Authentication Code Has Been Sent to +593087730033

Within a few minutes, you should receive a text message on your phone that includes a verification code. When you receive it, enter it below. If you don't receive the text message, [try sending it again](#), or see the [App Engine FAQ](#).

Enter Account Code:

© 2008 Google | [Terms of Service](#) | [Privacy Policy](#) | [Blog](#) | [Discussion Forums](#)

Figura III - 28 An Authentication Code Has Been Sent to...

A continuación crear la Aplicación ingresando el nombre de la misma.

Create an Application

You have 10 applications remaining.

Application Identifier:

You can map this application to your own domain later. [Learn more](#)

Application Title:

Displayed when users access your application.

Authentication Options (Advanced): [Learn more](#)
Google App Engine provides an API for authenticating your users, including Google Accounts, Google Apps, and OpenID. If you choose to use this feature for some parts of your site, you'll need to specify now what type of users can sign in to your application.

Open to all Google Accounts users (default)
If your application uses authentication, anyone with a valid Google Account may sign in. (This includes all Gmail Accounts, but does not include accounts on any Google Apps domains.)
[Edit](#)

Terms of Service:

1. Your Agreement with Google

1.1. Your use of the Google App Engine service (the "Service") is governed by this agreement (the "Terms"). "Google" means Google Inc., located at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States, and its subsidiaries or affiliates involved in providing the Service.

1.2. In order to use the Service, you must first agree to the Terms. You can agree to the Terms by actually using the Service. You understand and agree that Google will treat your use of the Service as acceptance of the Terms from that point onwards.

Figura III - 29 Create Application

Una vez ingresado el nombre, muestra una página de confirmación satisfactoria del registro de la Aplicación.

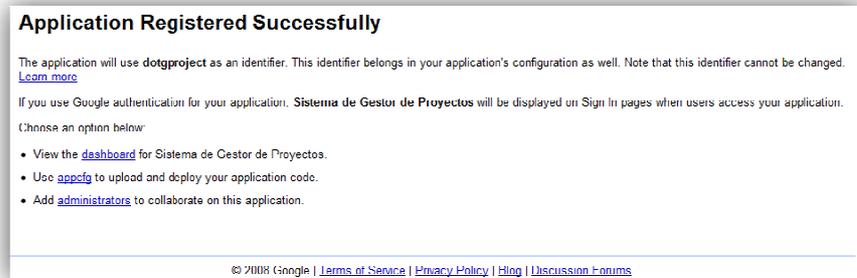


Figura III - 30 Application Registered Successfully

Google App Engine da un listado de las Aplicaciones creadas. Para acceder al Panel de Control de cualquier Aplicación dar clic en el nombre de la misma, esto permitirá tener un acceso directo al Panel de Administración.



Figura III - 31 My Applications

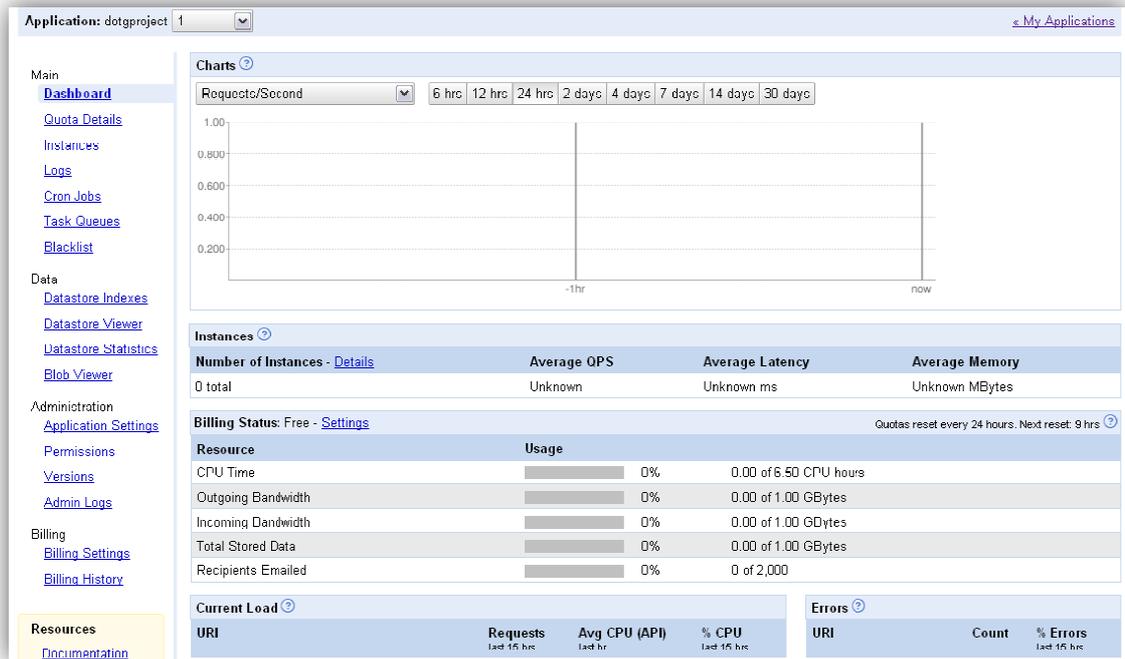


Figura III - 32 Panel de Administración

3.5. Análisis Comparativo

En esta sección se va a mostrar el estudio de los modelos de hosting Amazon EC2 y Google App Engine a manera de cuadros comparativos, seguido de una interpretación y calificación del criterio evaluado por parte del autor, estos cuadros comparativos se encuentran clasificados de acuerdo a los parámetros de comparación definidos anteriormente.

Para obtener los resultados cuantitativos y cualitativos que permitan una selección sustentada de uno de los modelos de hosting analizados, se calificó cada uno de los parámetros de comparación, basados en la siguiente escala:

Tabla III - XXIII Escala Cualitativa

Regular	Bueno	Muy bueno	Excelente
<70%	>=70% y <80%	>=80% y <90%	>=90%

Tabla III - XXIV Escala Cuantitativa

1	2	3
Malo	Regular	Bueno
Ninguno	Pocos	Muchos
No		Si
Relacional		No Relacional
Con Costo		Sin Costo
Complejo	Medianamente Complejo	No Complejo
Alto	Medio	Bajo
Ninguno	Parcialmente	Totalmente

Cada uno de los ítems de la interpretación incluye la siguiente nomenclatura:

(x,y)/z en donde cada letra significa lo siguiente:

X: representa el puntaje que obtiene Amazon EC2

Y: representa el puntaje que obtiene Google App Engine

Z: representa la base del puntaje sobre la cual se está representando el parámetro

La calificación definitiva del modelo de hosting en base a cada parámetro de comparación se obtiene sumando los puntajes obtenidos del análisis, utilizando las siguientes formulas:

$$P_{amz} = \sum(x), P_{gae} = \sum(y), P_c = \sum(z)$$

$$\text{Calificación de Amazon EC2 (Cc - Amz)} = (P_{amz} / P_c) * 100\%$$

$$\text{Calificación de Google App Engine (Cc - Gae)} = (P_{gae} / P_c) * 100\%$$

En donde:

P_{amz}: puntaje acumulado por Amazon EC2 en el parámetro

P_{gae}: puntaje acumulado por Google App Engine en el parámetro

P_c: puntaje sobre el que se califica el parámetro

Cc – Amz: porcentaje de la calificación total que obtuvo Amazon EC2 en el parámetro

Cc – Amz: porcentaje de la calificación total que obtuvo Google App Engine en el parámetro

3.5.1. Lenguajes de Programación

A pesar de la gran variedad de lenguajes de programación que existen en la actualidad, java es uno de los más completos y amigables para los desarrolladores, ya que fue diseñado con lo mejor de los otros lenguajes como C++.

3.5.1.1. Determinación de Variables

- a) Variedad de lenguajes de programación
- b) Dificultad en los lenguajes de programación

3.5.1.2. Valoraciones

a) Variedad de lenguajes de programación

Valoración: 3 puntos.

b) Dificultad en los lenguajes de programación

Valoración: 3 puntos.

Tabla III - XXV Variedad de Lenguajes de Programación.

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Variedad de lenguajes de programación	Muchos	Pocos
Dificultad en los lenguajes de programación	Medio	Medio

3.5.1.3. Interpretación

- En Amazon EC2 ofrece varios lenguajes de programación entre ellos están: Visual Basic, C#, Perl, Php, Java y Python. Y en Google App Engine soporta dos lenguajes de programación como son: Python y Java. (3,2)/3

- Se encontró el mismo nivel de dificultad en los lenguajes de programación de Amazon EC2 y Google App Engine, ya que los dos ofrecen lenguajes reconocidos entre los desarrolladores.(2,2)/3

3.5.1.4. Calificación

$$Pc = \sum(z) = 3+3 = 6$$

$$Pamz = \sum(x) = 3+2=5$$

$$Pgea = \sum(y) = 2+2=4$$

$$Cc - Amz: Pamz / Pc = (5/6)*100\% = 83.33\%$$

$$Cc - Gea: Pgea / Pc = (4/6)*100\% = 66.66\%$$

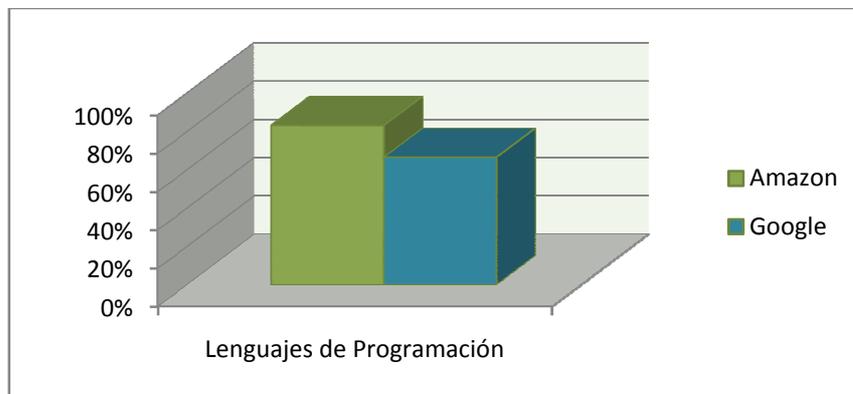


Figura III - 33 Interpretación del parámetro - Lenguaje de Programación

3.5.2. Sistema Operativo

Es responsable de gestionar, coordinar actividades, llevar a cabo el intercambio de recursos y actuar como intermediario para las aplicaciones que se ejecutan.

3.5.2.1. Determinación de Variables

- a) Cantidad de Sistemas Operativos soportados
- b) Sistemas Operativos gratuitos

3.5.2.2. Valoraciones

a) Cantidad de Sistemas Operativos soportados

Valoración: 3 puntos.

b) Sistemas Operativos gratuitos

Valoración: 3 puntos

Tabla III - XXVI Variedad de Sistemas Operativos

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Cantidad de Sistemas Operativos soportados	Muchos	Muchos
Sistemas Operativos gratuitos	Parcialmente	Parcialmente

3.5.2.3. Interpretación

- Los modelos de hosting a comparar permiten utilizar los sistemas operativos más comunes en el mercado como son Windows y Linux. (3,3)/3
- Ambos modelos de hosting soportan Sistemas Operativos gratuitos reconocidos.(2,2)/3

3.5.2.4. Calificación

$P_c = \sum(z) = 3+3=6$

$P_{amz} = \sum(x) = 3+2=5$

$P_{gea} = \sum(y) = 3+2=5$

$C_c - Amz: P_{amz} / P_c = (5/6)*100\% = 83.33\%$

$C_c - Gea: P_{gea} / P_c = (5/6)*100\% = 83.33\%$

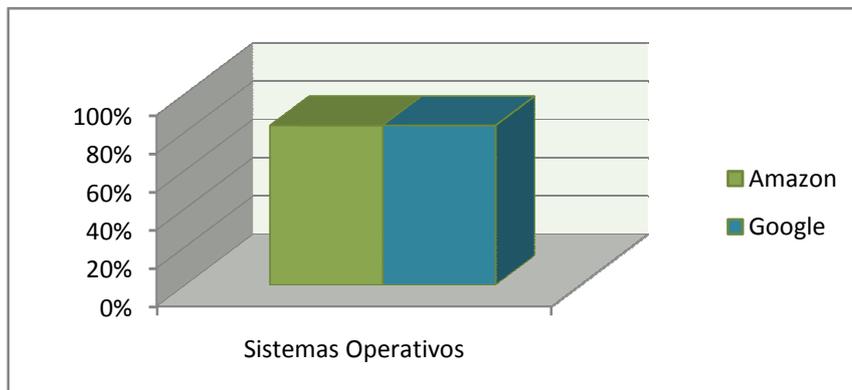


Figura III - 34 Interpretación del parámetro - Sistema Operativo.

3.5.3. Bases de Datos

Las bases de datos permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada.

3.5.3.1. Determinación de Variables

- a) Variedad de Bases de Datos
- b) Relacional

3.5.3.2. Valoraciones

a) Variedad de Bases de Datos

Valoración: 3 puntos.

b) Relacional

Base de Datos Relacional utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente.

Base de Datos no Relacional utilizado en la actualidad para aplicaciones grandes. Valoración: 3 puntos.

Tabla III - XXVII Variedad de Base de Datos

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Variedad de Bases de Datos	Muchos	Pocos
Relacional	Si	No

3.5.3.3. Interpretación

- En Amazon EC2 se encuentra una gran variedad de opciones de bases de datos entre las que se puede mencionar: Oracle, Mysql, IBM; Mientras que en Google App Engine únicamente ofrece BigTable. (3,2)/3
- Debido a la magnitud a la que puede alcanzar esta aplicación lo más recomendable es utilizar una base de datos no relacional. (1,3)/3

3.5.3.4. Calificación

$$P_c = \sum(z) = 3+3=6$$

$$P_{amz} = \sum(x) = 3+1 = 4$$

$$P_{gea} = \sum(y) = 2+3 = 5$$

$$C_c - Amz: P_{amz} / P_c = (4/6)*100\% = 66.66\%$$

$$C_c - Gea: P_{gea} / P_c = (5/6)*100\% = 83.33\%$$

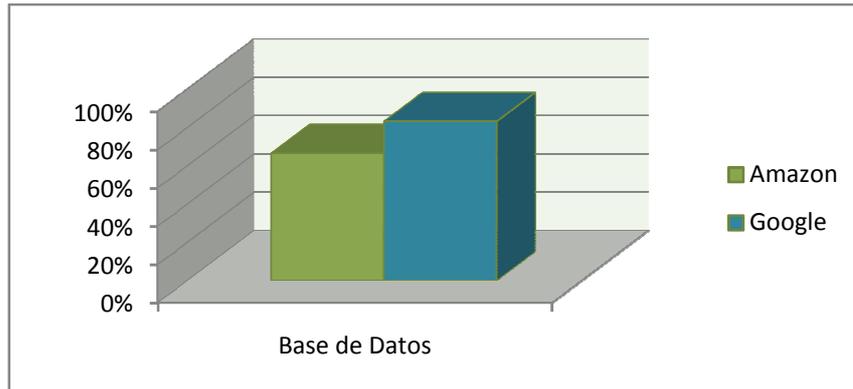


Figura III - 35 Interpretación del parámetro - Base de Datos.

3.5.4. Ambientes de Desarrollo

Son programas informáticos compuestos por un conjunto de herramientas de programación que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica

3.5.4.1. Determinación de Variables

- Cantidad de Ambientes de Desarrollo
- Dificultad en los Ambientes de Desarrollo

3.5.4.2. Valoraciones

a) Cantidad de Ambientes de Desarrollo

Valoración: 3 puntos.

b) Dificultad en los Ambientes de Desarrollo

Valoración: 3 puntos

Tabla III - XXVIII Variedad de Ambientes de Desarrollo

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Cantidad de Ambientes de Desarrollo	Muchos	Pocos
Dificultad en los Ambientes de Desarrollo	Medio	Medio

3.5.4.3. Interpretación

- En Amazon EC2 ofrece una gran variedad de ambientes de desarrollo al igual que Google App Engine que ofrece dos ambientes de desarrollo excelentes como son Netbeans y Eclipse.(3,2)/3
- Debido que los dos hosting soportan los mismos lenguajes la dificultad en los ambientes de desarrollo es la misma.(2,2)/3

3.5.4.4. Calificación

$$P_c = \sum(z) = 3+3=6$$

$$P_{amz} = \sum(x) = 3+2=5$$

$$P_{gea} = \sum(y) = 2+2=4$$

$$C_c - Amz: P_{amz} / P_c = (5/6)*100\% = 83.33\%$$

$$C_c - Gea: P_{gea} / P_c = (4/6)*100\% = 66.66\%$$

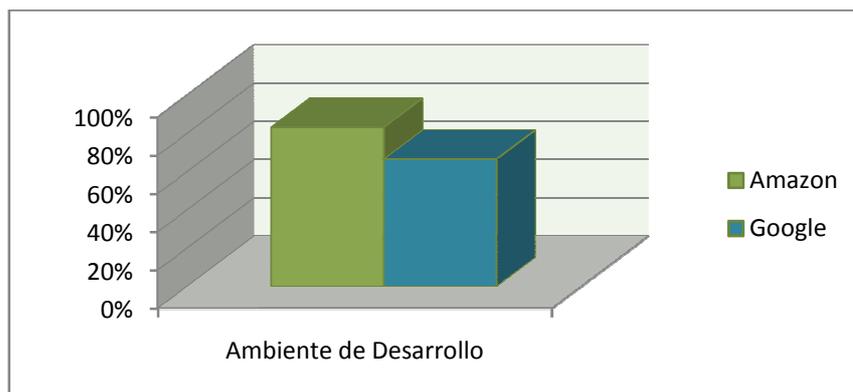


Figura III - 36 Interpretación del parámetro - Ambientes de Desarrollo.

3.5.5. Servidor de Aplicaciones

Usualmente se trata de un dispositivo de software que proporciona servicios de aplicación a las computadoras cliente. Un servidor de aplicaciones

generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación. Los principales beneficios de la aplicación de la tecnología de servidores de aplicación son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones.

3.5.5.1. Determinación de Variables

- a) Variedad de Servidor de Aplicaciones
- b) Servidor de Aplicaciones OpenSource

3.5.5.2. Valoraciones

a) Variedad de Servidor de Aplicaciones

Valoración: 3 puntos.

b) Servidor de Aplicaciones OpenSource

Valoración: 3 puntos.

Tabla III - XXIX Variedad de Servidor de Aplicaciones

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Variedad de Servidor de Aplicaciones	Muchos	Muchos
Servidor de Aplicaciones OpenSource	Muchos	Muchos

3.5.5.3. Interpretación

- Amazon EC2 y Google App Engine ofrecen una gran diversidad de servidor de aplicaciones.(3,3)/3
- Todos los servidores de aplicaciones de Google App Engine son OpenSource y Amazon EC2 en su mayoría brinda la misma ventaja y por otra parte cuenta con servidores pagados.(3,3)/3

3.5.5.4. Calificación

$$P_c = \sum(z) = 3+3=6$$

$$P_{amz} = \sum(x) = 3+3=6$$

$$P_{gea} = \sum(y) = 3+3=6$$

$$C_c - Amz: P_{amz} / P_c = (6/6)*100\% = 100\%$$

$$C_c - Gea: P_{gea} / P_c = (6/6)*100\% = 100\%$$

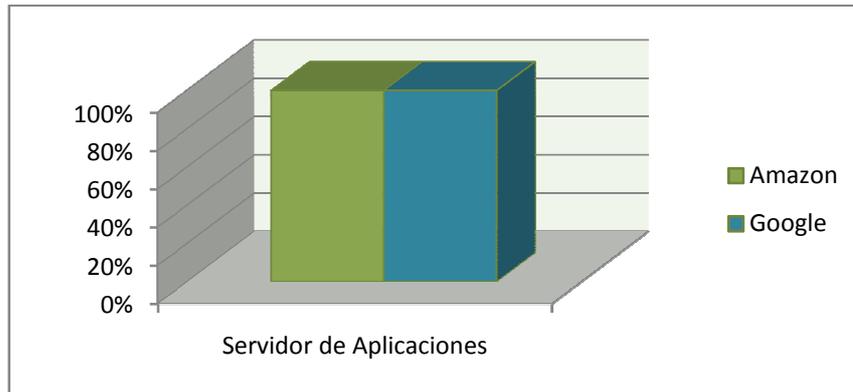


Figura III - 37 Interpretación del parámetro - Servidor de Aplicaciones.

3.5.6. Costos

3.5.6.1. Determinación de Variables

- a) Costo de Tiempo de CPU
- b) Costo de Tráfico de Salida
- c) Costo de Tráfico de Entrada
- d) Costo de envío de Emails
- e) Costo de Almacenamiento

3.5.6.2. Valoraciones

a) Costo de Tiempo de CPU

Es el costo que representa la utilización del CPU durante un periodo de tiempo de 1 mes. Valoración: 3 puntos.

b) Costo de Tráfico de Salida

Es el costo que representa la extracción de datos o muestra de resultados con capacidad de 1 Gb durante un periodo de tiempo de 1 mes. Valoración 3 puntos.

c) Costo de Tráfico de Entrada

Es el costo que representa el envío de datos con capacidad de 1 Gb durante un periodo de tiempo de 1 mes. Valoración 3 puntos.

d) Costo de envío Emails

Es el costo que representa el envío de 2000 emails durante un periodo de tiempo de 1 mes. Valoración 3 puntos.

e) Costo de Almacenamiento

Es el costo que representa el alojamiento de información en memoria con capacidad de 1 Gb durante un periodo de tiempo de 1 mes. Valoración 3 puntos.

Tabla III - XXX Valoración de Costos

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Costo de Tiempo de CPU	Si	No
Costo de Tráfico de Salida	Si	No
Costo de Tráfico de Entrada	No	No
Costo de envío de Emails	Si	No
Costo de Almacenamiento	Si	No

3.5.6.3. Interpretación

- En Amazon EC2 tiene un costo de USD 4.42 en el tiempo de utilización del CPU en un periodo de 1 mes mientras que en Google App Engine tiene un costo de cero. (1,3)/3.^{10 11}
- En Amazon EC2 tiene un costo de USD 0.10 en la extracción de datos con capacidad de 1 Gb en un periodo de 1 mes mientras que en Google App Engine tiene un costo de cero. (1,3)/3.
- En Amazon EC2 tiene un costo de cero el envío de datos con una capacidad de 1 Gb en un periodo de 1 mes al igual que en Google App Engine tiene un costo de cero. (3,3)/3.
- En Amazon EC2 el costo de USD 0.10 en el envío de emails en un periodo de 1 mes mientras que en Google App Engine tiene un costo de cero. (1,3)/3.
- En Amazon EC2 el costo de alojamiento de información en memoria con capacidad de 1 Gb durante un periodo de 1 mes es de USD 0.14 mientras que en Google App Engien tiene un costo de cero. (1,3)/3.

3.5.6.4. Calificación

$$Pc = \sum(z) = 3+3+3+3+3 = 15$$

$$Pamz = \sum(x) = 1+1+3+1+1 = 7$$

$$Pgea = \sum(y) = 3+3+3+3+3 = 15$$

$$Cc - Amz: Pamz / Pc = (7/15)*100\% = 46.66\%$$

$$Cc - Gea: Pgea / Pc = (15/15)*100\% = 100\%$$

¹⁰ <http://aws.amazon.com/es/s3/#pricing>

¹¹ <http://code.google.com/intl/es/appengine/docs/quotas.html>

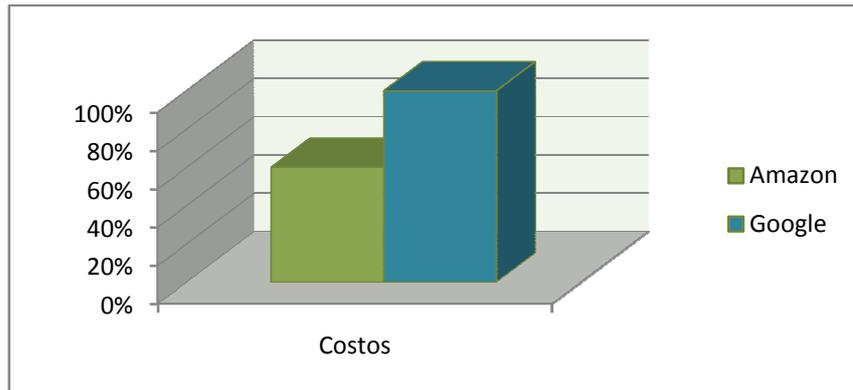


Figura III - 38 Interpretación del parámetro - Costos.

3.5.7. Configuración

3.5.7.1. Determinación de Variables

- a) Complejidad
- b) Nivel de Conocimiento
- c) Tiempo
- d) Recurso Humano

3.5.7.2. Valoraciones

a) Complejidad

Es el número de pasos requeridos para completar la configuración de un proyecto o instancia. Valoración 3 puntos.

b) Nivel de Conocimiento

Es el nivel de conocimiento que se necesita para realizar la configuración. Valoración 3 puntos.

c) Tiempo

Es el tiempo invertido que demora en realizar la configuración inicial. Valoración 3 puntos.

d) Recurso Humano

Es la cantidad de personal requerido para realizar la configuración.
Valoración 3 puntos.

Tabla III - XXXI Valoración de Configuración

VARIABLE	AMAZON EC2	GOOGLE APP ENGINE
Complejidad	Alto	Bajo
Nivel de Conocimiento	Alto	Medio
Tiempo	Alto	Bajo
Recurso Humanos	Alto	Bajo

3.5.7.3. Interpretación

- En Amazon EC2 se requiere un mayor número de pasos para la configuración por lo que se lo califica como más complejo a diferencia de Google App Engine ya que es más simple en su configuración. (1,3)/3.
- En Amazon EC2 se necesita un mayor nivel de conocimiento para su configuración a diferencia de Google App Engine que solo requiere de conocimientos básicos. (1,2)/3.
- En Amazon EC2 se utiliza mayor tiempo de configuración a diferencia de Google App Engine utiliza un mínimo de tiempo. (1,3)/3.
- En Amazon EC2 necesita un mayor número de personal para diferentes áreas de la configuración a diferencia de Google App Engine que fácilmente con una sola persona se puede realizar la configuración. (1,3)/3.

3.5.7.4. Calificación

$$P_c = \sum(z) = 3+3+3+3 = 12$$

$$P_{amz} = \sum(x) = 1+1+1+1 = 4$$

$$P_{gea} = \sum(y) = 3+2+3+3 = 11$$

$$C_c - Amz: P_{amz} / P_c = (4/12)*100\% = 33.33\%$$

$$C_c - Gea: P_{gea} / P_c = (11/12)*100\% = 91.67\%$$

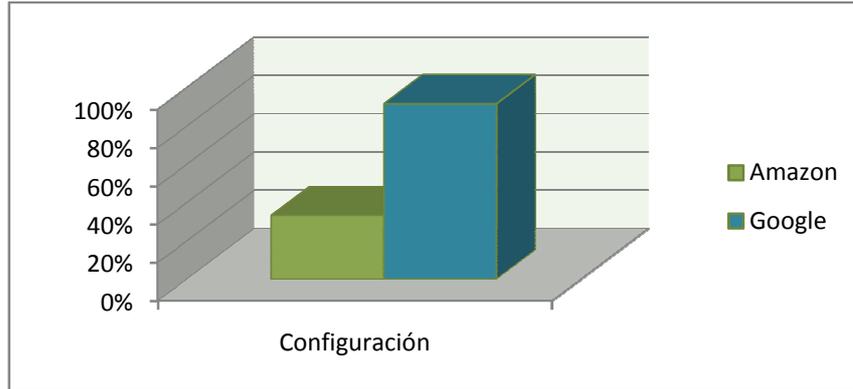


Figura III - 39 Interpretación de Parámetros - Configuración

3.6. RESUMEN COMPARATIVO

Con la finalidad de presentar los resultados del estudio comparativo realizado, se pone a consideración la siguiente tabla:

Tabla III - XXXII Resumen Comparativo

	Lenguaje de Programación (6/6)	Sistema Operativo (6/6)	Bases de Datos (6/6)	Ambientes de Desarrollo (6/6)	Servidor de Aplicaciones (6/6)	Costos (15/15)	Configuración (12/12)	TOTAL (57)
AMAZON EC2	5	5	4	5	6	7	4	31
GOOGLE APP ENGINE	4	5	5	4	6	15	11	50

3.7. COMPROBACIÓN DE LA HIPÓTESIS

El “grado de usabilidad” de un sistema es, por su parte, una medida empírica y relativa de la usabilidad del mismo, la cual se evalúa mediante opiniones relacionadas con la satisfacción de los usuarios frente al uso de elementos que forman parte de los modelos de hosting para Cloud Computing.

La usabilidad para la implementación de un proyecto sobre Google App Engine frente a la implementación en Amazon EC2 ha sido medida en base a opiniones recolectadas en encuestas publicadas en internet en las siguientes URL:

- Google App Engine¹²
- Amazon EC2¹³

Ver ANEXO 3., dirigidas a una muestra tomada de los usuarios que comúnmente utilizan estos modelos.

Representación de los resultados obtenidos.

- Google App Engine:

Tabla III - XXXIII Presentación de resultados de la Encuesta de Google App Engine

Pregunta	0	1	2	3	4	Total	Promedio
¿Qué nivel de dificultad considera usted que tuvo al abrir una cuenta?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	0	0	0	10	263	1082	4
¿Qué nivel de dificultad considera usted que tuvo al configurar un nuevo proyecto/Instancia?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	0	0	0	249	24	843	3
¿Qué nivel de dificultad considera usted que tuvo al publicar un proyecto/Instancia?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	0	0	35	180	58	824	3
¿Cómo considera usted la integración con servicios externos (autenticación, documentos, email)?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	0	0	2	90	181	998	4
¿Qué nivel de dificultad considera usted que tuvo al momento de escalar un	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	0	2	74	175	22	763	3

¹² <https://spreadsheets.google.com/viewform?formkey=dHpnNmpPbkRPcDVycWZEU3cxN0dHbVE6MQ&output=html>

¹³ <https://spreadsheets.google.com/viewform?formkey=dEFVN2tod1JvQ2RGT3RNck1JdjQxaHc6MQ>

proyecto/Instancia?							
¿Qué nivel de dificultad considera usted que tuvo al momento de utilizar al panel de administración?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	0	0	2	120	151	968	4
¿Cómo considera usted los costos para un proyecto/instancia?	Muy Costoso	Costoso	Medianamente económico	Económico	Sin Costo		
	0	0	0	28	245	1064	4
¿Cuál es la cantidad de personal adecuada para administrar un proyecto/instancia?	10 a más P.		5 a 10 P.		1 a 5 P.		
	0		0		273	1092	4
¿Cuál es la cantidad de tiempo promedio para publicar un proyecto/instancia?	10 a más H.		5 a 10 H.		1 a 5 H.		
	0		0		273	1092	4
¿Cómo considera usted el rendimiento en CDN?	Malo	Regular	Bueno	Medianamente bueno	Muy Bueno		
	0	182	69	22	0	386	1
TOTAL						9112	34



Figura III - 40 Encuesta Google App Engine - Primera Pregunta

Qué nivel de dificultad considera usted que tuvo al configurar un nuevo proyecto/Instancia en Google App Engine?

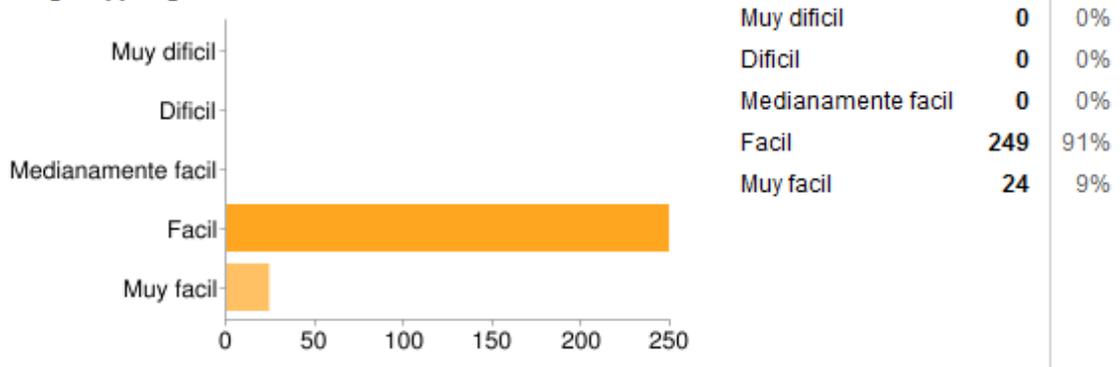


Figura III - 41 Encuesta Google App Engine - Segunda Pregunta

Qué nivel de dificultad considera usted que tuvo al publicar un proyecto/Instancia en Google App Engine?

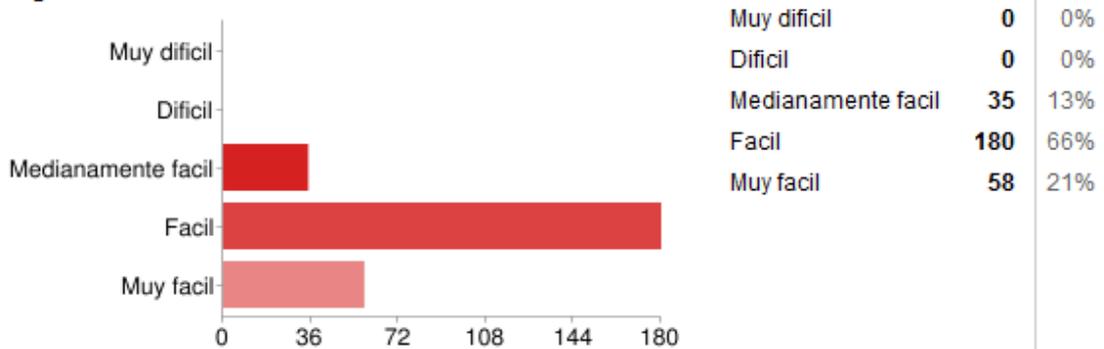


Figura III - 42 Encuesta Google App Engine - Tercera Pregunta

Cómo considera usted la integración con servicios externos (autenticación, documentos, email) con Google App Engine?

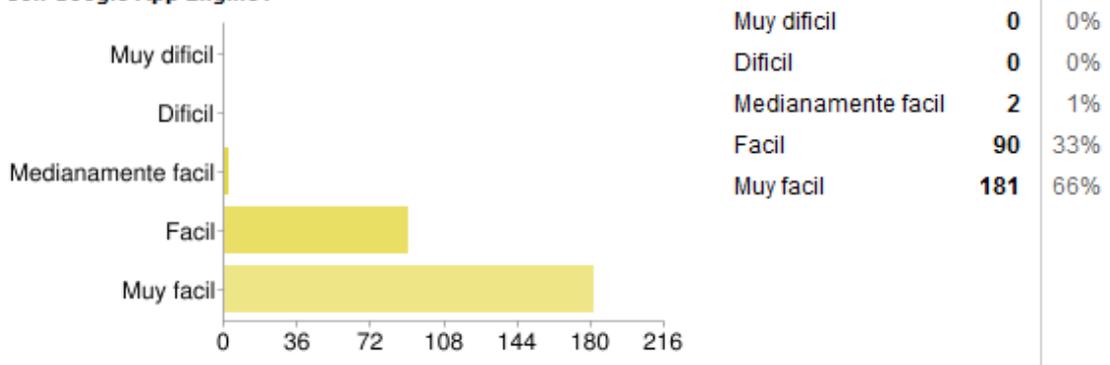


Figura III - 43 Encuesta Google App Engine - Cuarta Pregunta

Qué nivel de dificultad considera usted que tuvo al momento de escalar un proyecto/Instancia en Google App Engine?

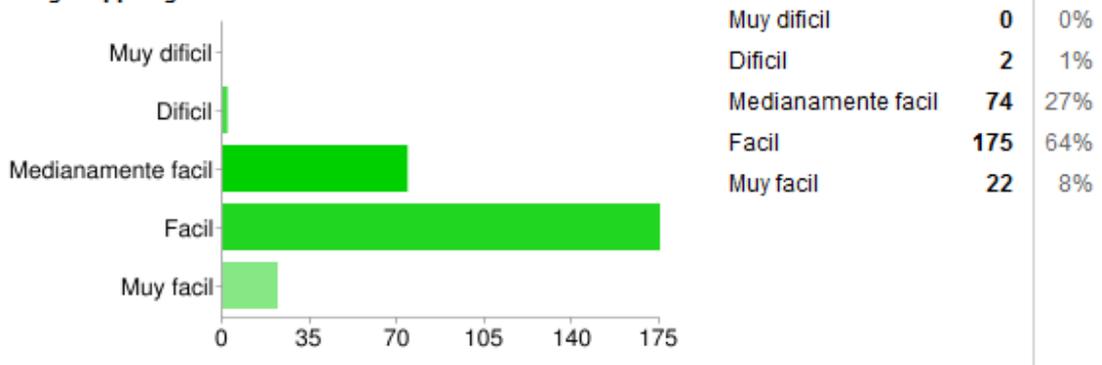


Figura III - 44 Encuesta Google App Engine - Quinta Pregunta

Qué nivel de dificultad considera usted que tuvo al momento de utilizar al panel de administración en Google App Engine?

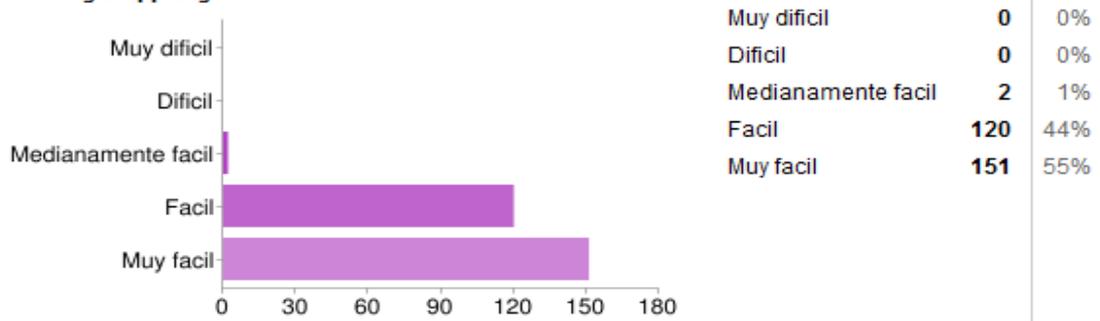


Figura III - 45 Encuesta Google App Engine - Sexta Pregunta

Cómo considera usted los costos para un proyecto/instancia en Google App Engine?

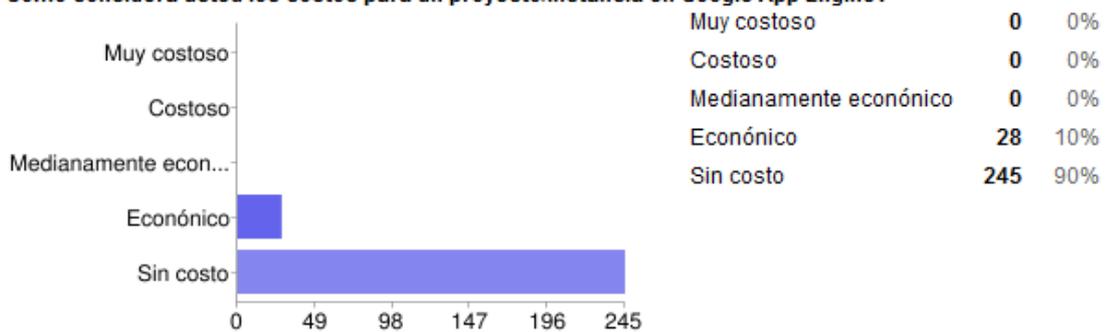


Figura III - 46 Encuesta Google App Engine - Septima Pregunta

Cual es la cantidad de personal adecuada para administrar un proyecto/instacia en Google App Engine?



Figura III - 47 Encuesta Google App Engine - Octava Pregunta

Cual es la cantidad de tiempo promedio para publicar un proyecto/instancia en Google App Engine?



Figura III - 48 Encuesta Google App Engine - Novena Pregunta

Cómo considera usted el rendimiento de Google App Engine con CDN?

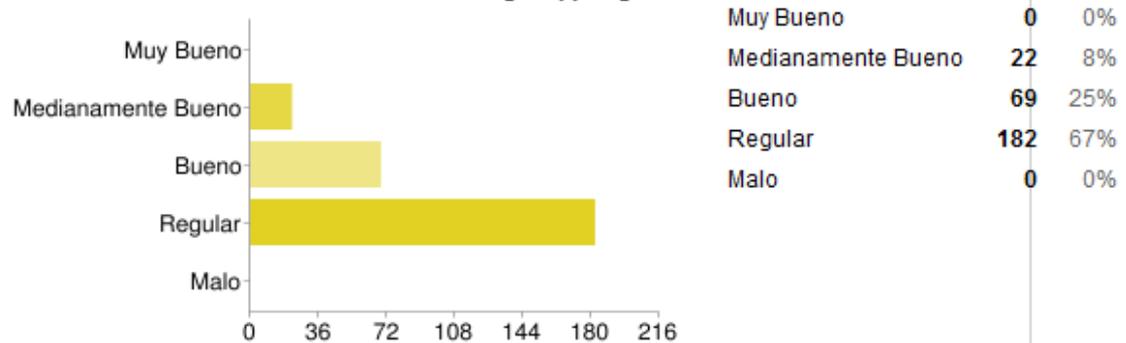


Figura III - 49 Encuesta Google App Engine - Décima Pregunta

- Amazon EC2

Tabla III - XXXIV Presentación de resultados de la Encuesta de Amazon EC2

Pregunta	0	1	2	3	4	Total	Promedi
¿Qué nivel de dificultad considera usted que tuvo al abrir una cuenta?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	69	146	37	4	17	300	1
¿Qué nivel de dificultad considera usted que tuvo al configurar un nuevo proyecto/Instancia?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	197	76	0	0	0	76	0
¿Qué nivel de dificultad considera usted que tuvo al publicar un proyecto/Instancia?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	82	123	68	0	0	259	1
¿Cómo considera usted la integración con servicios externos (autenticación, documentos, email)?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	83	188	2	0	0	192	1
¿Qué nivel de dificultad considera usted que tuvo al momento de escalar un proyecto/Instancia?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	42	172	59	0	0	290	1
¿Qué nivel de dificultad considera usted que tuvo al momento de utilizar al panel de administración?	Muy difícil	Difícil	Medianamente fácil	Fácil	Muy fácil		
	27	51	194	1	0	442	2
¿Cómo considera usted los costos para un proyecto/instancia?	Muy Costoso	Costoso	Medianamente económico	Económico	Sin Costo		
	179	92	2	0	0	96	0
¿Cuál es la cantidad de personal adecuada para administrar un proyecto/instancia?	10 a más P.		5 a 10 P.		1 a 5 P.		
	142		124		7	276	1
¿Cuál es la cantidad de tiempo promedio para publicar un proyecto/instancia?	10 a más H.		5 a 10 H.		1 a 5 H.		
	220		46		7	120	0
¿Cómo considera usted el rendimiento en CDN?	Malo	Regular	Bueno	Medianamente bueno	Muy Bueno		
	0	56	48	113	56	715	3
TOTAL						2766	10

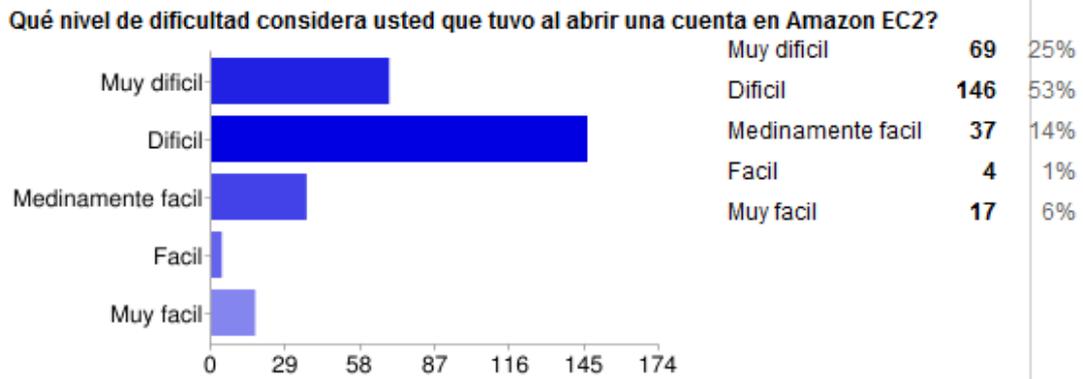


Figura III - 50 Encuesta Amazon EC2 - Primera Pregunta

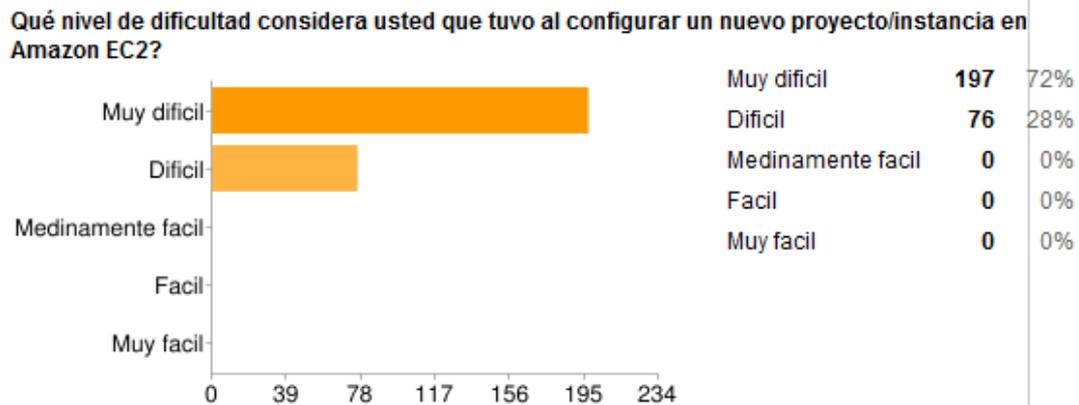


Figura III - 51 Encuesta Amazon EC2 - Segunda Pregunta

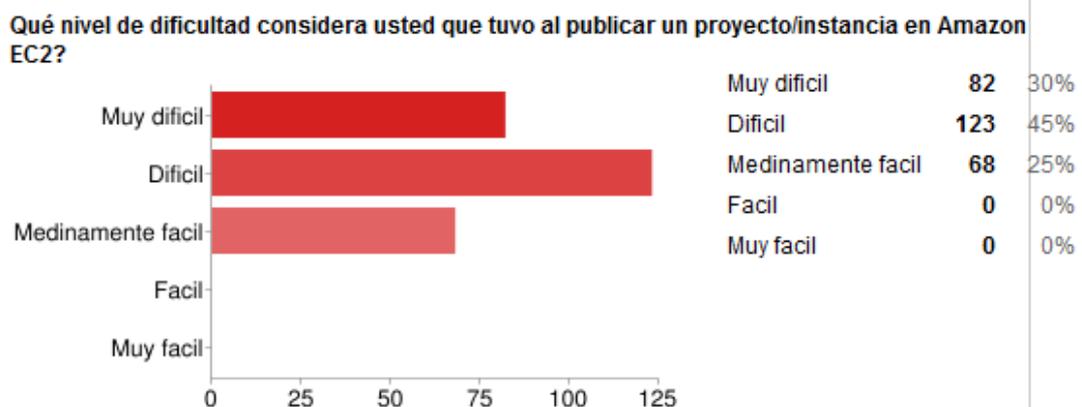


Figura III - 52 Encuesta Amazon EC2 - Tercera Pregunta

Qué nivel de dificultad considera usted que tuvo al momento de escalar un proyecto/instancia en Amazon EC2?

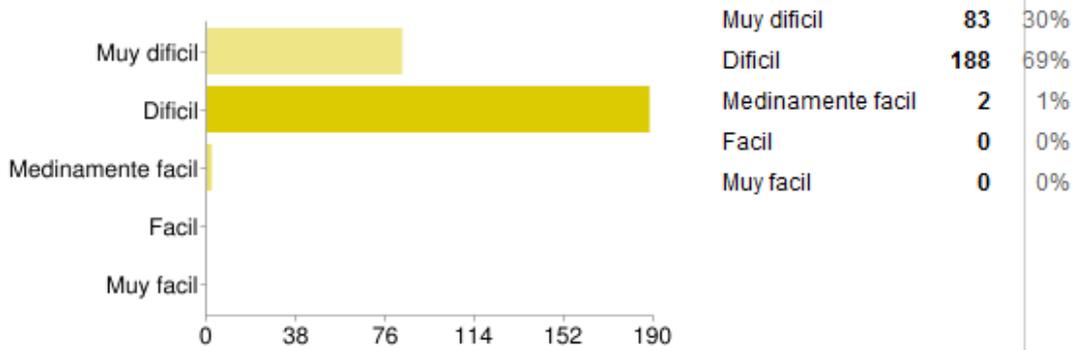


Figura III - 53 Encuesta Amazon EC2 - Cuarta Pregunta

Cómo considera usted la integración con servicios externos (autenticación, documentos, email) con Amazon EC2?

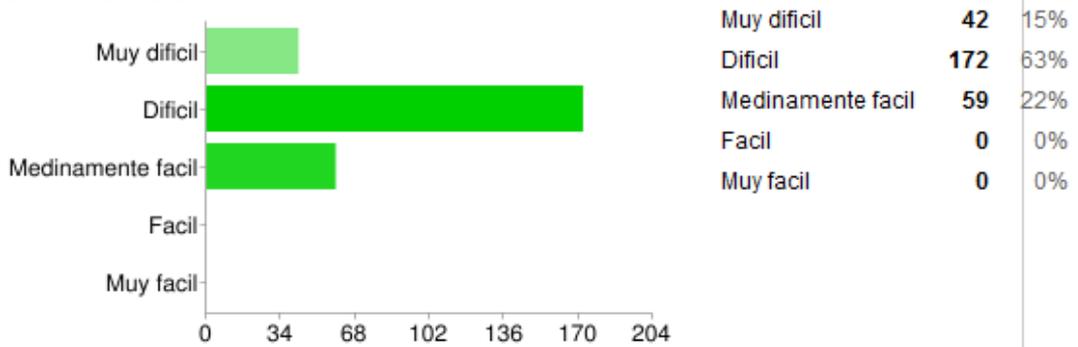


Figura III - 54 Encuesta Amazon EC2 - Quinta Pregunta

Qué nivel de dificultad considera usted que tuvo al momento de utilizar el panel de administración en Amazon EC2?

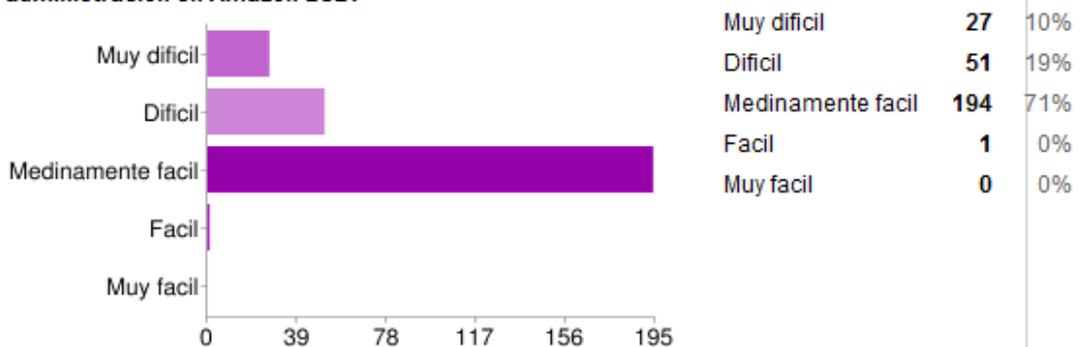


Figura III - 55 Encuesta Amazon EC2 - Sexta Pregunta

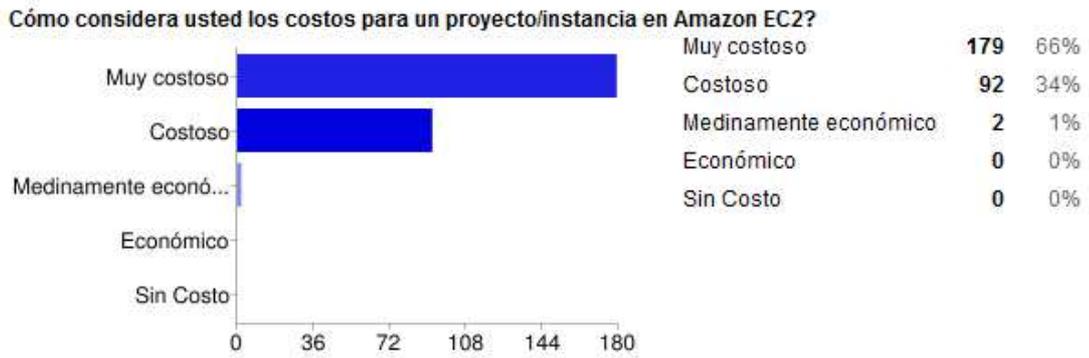


Figura III - 56 Encuesta Amazon EC2 - Séptima Pregunta

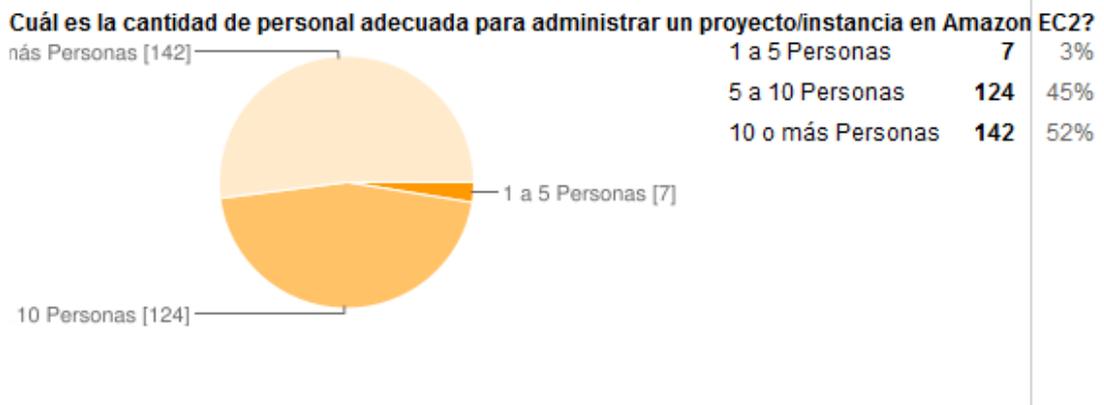


Figura III - 57 Encuesta Amazon EC2 - Octava Pregunta



Figura III - 58 Encuesta Amazon EC2 - Novena Pregunta

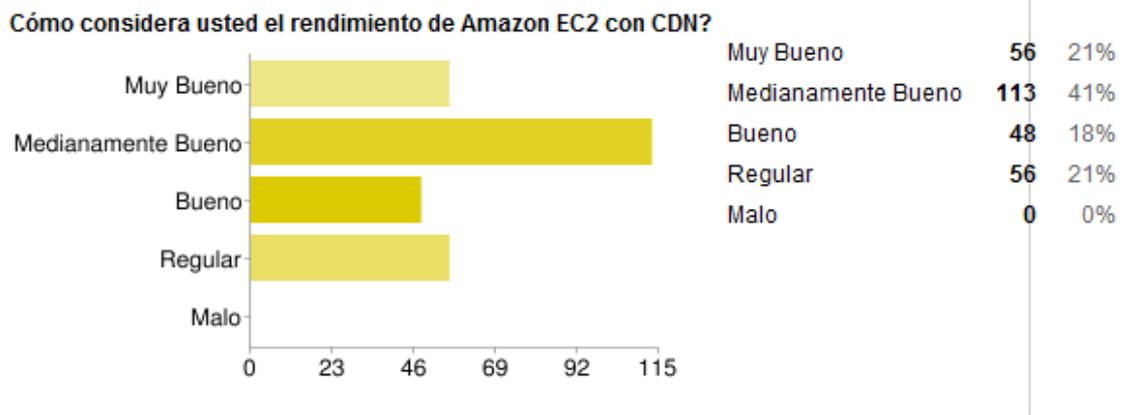


Figura III - 59 Encuesta Amazon EC2 - Décima Pregunta

Hipótesis para chi- cuadrado:

Hi: hipótesis de la investigación

Ho: hipótesis nula

Hi: El modelo Amazon EC2 es la mejor opción con respecto a la usabilidad de Cloud Computing para implementar un Gestor de Proyectos en Línea.

Ho: El modelo Amazon EC2 no es la mejor opción con respecto a la usabilidad de Cloud Computing para implementar un Gestor de Proyectos en Línea.

Variable independiente: Amazon EC2.

Variables dependientes: Usabilidad.

Se presenta los Indicadores y sus respectivos valores de acuerdo a si mejora o no con respecto a la usabilidad

Tabla III - XXXV Indicadores

Indicadores	Amazon EC2		Google App Engine	
	Mejora	No mejora	Mejora	No mejora
Lenguaje de Programación	1			
Sistemas Operativos	1		1	
Base de Datos	1	1	1	
Ambientes de Desarrollo	1			
Servidor de Aplicación	2		2	
Costos	1	4	5	
Configuración		4	3	
Pregunta 1		1	1	
Pregunta 2		1	1	
Pregunta 3		1	1	
Pregunta 4		1	1	
Pregunta 5		1	1	
Pregunta 6		1	1	
Pregunta 7		1	1	
Pregunta 8		1	1	
Pregunta 9		1	1	
Pregunta 10	1			1
TOTAL:	8	18	21	1

Datos Observados

Tabla III - XXXVI Datos Observados

Usabilidad	Mejora	No mejora	Total
Hipótesis			
Amazon EC2 es la mejor opción	8	18	26
Amazon EC2 no es la mejor opción	21	1	22
Total	29	19	48

Datos Esperados

VE: Valores Esperados

Tf: Total Fila

Tc: Total Columna

T: Total

$$VE = \frac{Tf * Tc}{T}$$

Tabla III - XXXVII Datos Esperados

Usabilidad \ Hipótesis	Mejora	No mejora	Total
Amazon EC2 es la mejor opción	14	14	28
Amazon EC2 no es la mejor opción	16	7	24
Total	30	22	52

Chi Cuadrado

S²: Chi cuadrado

O: Valor observado

E: Valor Esperado

$$S^2 = \frac{(O - E)^2}{E}$$

Tabla III - XXXVIII Cálculo de Chi Cuadrado

Usabilidad \ Hipótesis	O	E	(O-E) ²	(O-E) ² /E
Amazon EC2 es la mejor opción	8	14	36	2.5
	18	14	16	1.1
Amazon EC2 no es la mejor opción	21	16	25	1.5
	1	7	36	5.1
Total Calculado:				10.2

Grado de Libertad

Para poder realizar la comparación del resultado de chi cuadrado se debe encontrar los grados de libertad de nuestra tabla, para ello tenemos la siguiente:

r: es el número de filas de la tabla de contingencia

c: es el número de columnas de la tabla de contingencia

$$gl = (f - 1)(c - 1)$$

$$gl = (2-1) \cdot (2-1) = 1$$

$$\alpha = 1\% (0,001)$$

De la tabla de distribución de X^2 , podemos decir que con un 1% de seguridad y con 1 grado de libertad el valor de crítico de chi- cuadrado es 10,82.

Criterio de decisión

- Si X^2 calculado es mayor a X^2_{α} (Valor crítico) de la tabla de distribución se rechaza la hipótesis nula H_0 y por lo tanto se acepta la hipótesis de Investigación.
- Si X^2 calculado es menor a X^2_{α} (Valor crítico) de la tabla de distribución se acepta la hipótesis nula H_0 y por lo tanto se rechaza la hipótesis de Investigación.

Por lo tanto tenemos el siguiente resultado:

$$X^2 \text{ Calculado} = 10,2$$

$$X^2_{\alpha} \text{ (Valor crítico)} = 10,82$$

$$10,2 < 10,82$$

$$\text{Por lo tanto: } X^2 < X^2_{\alpha}$$

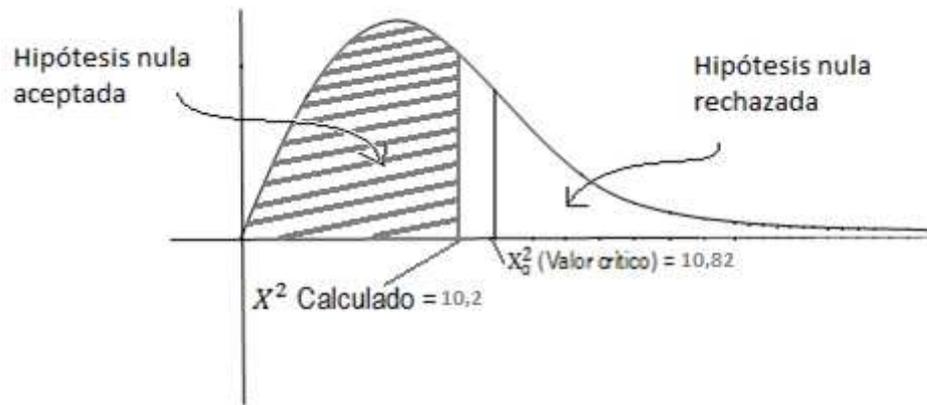


Figura III - 60 Diagrama de Chi cuadrado

Resultados Finales:

Google App Engine	50 / 57 puntos = 87,72%
Amazon EC2	31 / 57 puntos = 54,38%

Google App Engine y Amazon EC2 son modelos de hosting para Cloud Computing, los cuales ofrecen las mejores características para el desarrollo y la implementación de un proyecto o instancia. Luego del estudio comparativo y con los resultados obtenidos de la encuesta se deduce que Google App Engine posee el mayor nivel de usabilidad.

De esta manera negamos que el modelo Amazon EC2 es la mejor opción con respecto a la usabilidad de Cloud Computing para implementar un Gestor de Proyectos en Línea.

CAPITULO IV

DESARROLLO DE LA APLICACIÓN WEB, GESTOR DE PROYECTOS EN LÍNEA

4.1. PARTE TEÓRICA

4.1.1. Introducción

En la actualidad aun existen empresas que trabajan con proyectos en equipo, los mismos que están destinados a trabajar bajo limitantes de tiempo, con riesgos de fallos en conexión y restringidos a una red específica, retrasando de esta manera el proceso de desarrollo y quitando valor a la empresa.

SGP es un sistema web que brinda las herramientas necesarias para su administración, el mismo que posee una interfaz web intuitiva, grafica y eficiente.

4.1.2. Desarrollo

El sistema está basado en una metodología ágil: Programación Extrema XP, que se basa en una serie de reglas y principios de la Ingeniería de Software tradicional, pero con la excepción de que XP da prioridad a las tareas que es lo que precisamente da resultados directos y reduce los procesos largos de recopilación de gran información para generar resultados esperados.

La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck, autor del primer libro sobre la materia, *Extreme Programming Explained: Embrace Change* (1999). Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Es capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto.

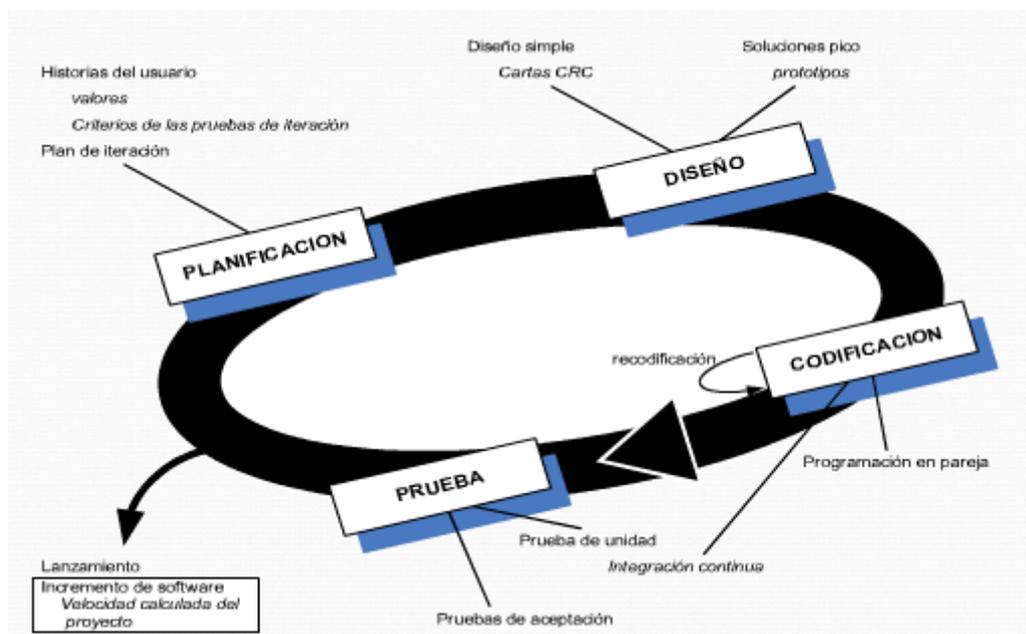


Figura IV - I Metodología XP

Se va a abarcar todo el ciclo de vida, empezando por los requisitos y acabando en el sistema funcionando, proporcionando así una visión completa y coherente de la producción de sistemas software. El enfoque que toma es el de un ciclo de vida iterativo incremental, el cual permite una gran flexibilidad a la hora de adaptarlo a un proyecto y a un equipo de desarrollo específicos. El ciclo de vida está dirigido por casos de uso, es decir, por la funcionalidad que ofrece el sistema a los futuros usuarios del mismo. Así no se pierde de vista la

motivación principal que debería estar en cualquier proceso de construcción de software: el resolver una necesidad del usuario/cliente.

I. Planificación

- Historias de usuario.
- Plan de Publicaciones.
- Iteraciones:
- Velocidad del proyecto.
- Programación en pareja.
- Reuniones diarias.

II. Diseño

- Diseños simples.
- Glosarios de términos.
- Riesgos.
- Funcionalidad extra.
- Tarjetas C.R.C.

III. Codificación

IV. Pruebas

- El uso de los test en X.P es el siguiente.
- Test de aceptación.

I. Planificación

- **Historias de usuario.**

El primer paso de cualquier proyecto que siga la metodología X.P es definir las historias de usuario con el cliente. Las historias de usuario tienen la misma finalidad que los casos de uso pero con algunas diferencias: Constan de 3 ó 4 líneas escritas por el cliente en un lenguaje no técnico sin hacer mucho hincapié en los detalle

- **Plan de Publicaciones.**

Es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa.

- **Iteraciones**

Se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el Plan de Publicaciones que serán implementadas.

- **Velocidad del proyecto.**

La velocidad del proyecto es una medida que representa la rapidez con la que se desarrolla el proyecto; estimarla es muy sencillo, basta con contar el número de historias de usuario que se pueden implementar en una iteración.

- **Programación en pareja.**

El trabajo en pareja involucra a dos programadores trabajando en el mismo equipo; mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. De esta forma se consigue un código y diseño con gran calidad.

- **Reuniones diarias.**

Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta.

II. Diseño

- **Diseños simples.**

Se debe procurar hacer todo lo menos complicado posible para conseguir un diseño fácilmente entendible e implementable que a la larga costará menos tiempo y esfuerzo desarrollar.

- **Glosarios de términos.**

Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

- **Riesgos.**

Si surgen problemas potenciales durante el diseño, X.P sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema.

- **Funcionalidad extra.**

No se debería añadir funcionalidad extra al programar, aunque se piense que en un futuro será utilizada. Sólo el 10% de la misma es utilizada, lo que implica que el desarrollo de funcionalidad extra es un desperdicio de tiempo y recursos.

- **Tarjetas C.R.C.**

El uso de las tarjetas C.R.C (Class, Responsibilities and Collaboration) permiten al programador centrarse y apreciar el desarrollo orientado a

objetos olvidándose de los malos hábitos de la programación procedural clásica.

III. Codificación

La codificación debe hacerse ateniendo a estándares de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad.

X.P también propone un modelo de desarrollo colectivo en el que todos los programadores están implicados en todas las tareas; cualquiera puede modificar o ampliar una clase o método de otro programador si es necesario y subirla al repositorio de código. El permitir al resto de los programadores modificar códigos que no son suyos no supone ningún riesgo ya que para que un código pueda ser publicado en el repositorio tiene que pasar los test de funcionamiento definidos para el mismo.

IV. Pruebas

Hay que someter a tests las distintas clases del sistema omitiendo los métodos más triviales.

Test de aceptación. Los test mencionados anteriormente sirven para evaluar las distintas tareas en las que ha sido dividida una historia de usuario. Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear "Test de aceptación"; estos test son creados y usados por los clientes para comprobar que las distintas historias de usuario cumplen su cometido.

4.1.2.1. Herramientas utilizadas

- **Eclipse**

Es una herramienta para el programador diseñada principalmente para el desarrollo de aplicaciones Java, facilitando al máximo la gestión de proyectos colaborativos mediante el control de versiones 'cvs', es posible también con subversión, exportar e importar proyectos.

Esta plataforma, típicamente ha sido usada para implementar entornos de desarrollo integrados, como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse).

- **BigTable**

Herramienta para el programador desarrollada principalmente para el desarrollo de aplicaciones Java, facilitando al máximo la gestión de proyectos colaborativos mediante el control de versiones 'cvs', es posible también con subversión, exportar e importar proyectos.

Google creó Bigtable para que fuese una base de datos en la que se almacenará una cantidad de información enorme, del orden de Petabytes. Para ello, cada tabla está dividida en “tablets” que pueden llegar a ocupar 200 Megabytes. Si superasen ese tamaño serían automáticamente divididas y comprimidas para ser enviadas a más máquinas usando un sistema de compresión propietario de Google.

4.2. PARTE PRÁCTICA

4.2.1. Introducción

La implementación del gestor de proyecto, independiente de la infraestructura de la ESPOCH permitirá un acceso sin restricciones de ubicación debido a que en otros tipos de plataformas implementadas en la institución su limitante principal fue que los miembros del equipo de desarrollo obligadamente debían

estar bajo la red de la institución.

Gracias al control de versiones del Almacén de información, ninguna información se pierde y puede ser accedida en cualquier momento. Todas las modificaciones de esos datos estarán inmediatamente disponibles para todo el mundo. Esto reduce la duplicación de datos y su mantenimiento.

4.2.2. Desarrollo

4.2.2.1. Fase 1: Definición del Ámbito

La Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo considerando que la investigación científico tecnológico es factor básico del desarrollo económico, social y cultural de un país, creó el Centro de Investigaciones y Producción (CIPFIE) como unidad académica administrativa, que se encargará de la planificación, ejecución, asesoramiento, coordinación y control de los proyectos de Investigación, Tesis, Memorias y se rige por el presente reglamento.

4.2.2.1.1 Equipo de Trabajo

Director:

Ing. Wladimir Castro.

Miembros:

Ing. Danilo Pastor

Docente Delegado

Srta. Ana Guaño

Sra. Gabriela Altamirano.

4.2.2.1.2 Servicios

- Desarrollar infraestructura investigativa, realizar cursos de especialización o actualización, seminarios, mesas redondas, respecto al desarrollo de investigación.
- Difundir periódicamente los resultados de las investigaciones a través de revistas y otros medios tecnológicos de comunicación.
- Buscar y tramitar el auspicio a través de organismos nacionales e internacionales para el desarrollo de los proyectos de investigación que este centro considere de importancia para su ejecución.

4.2.2.1.3 Funciones

- Seccionar ordinariamente una vez por mes y extraordinariamente cuando el caso lo requiera.
- Proponer al H. Consejo Directivo las reformas al Reglamento Interno y normativos de este Centro.
- Planificar, coordinar y evaluar los proyectos de investigación, tesis, memorias, y prácticas de pre-grado en la Facultad, a través de comisiones de ser necesario.
- Elaborar mensualmente su plan de actividades.
- Desarrollar infraestructura investigativa.
- Realizar cursos de especialización o actualización, seminarios, mesas redondas, respecto al desarrollo de investigación y capacitación.

4.2.2.1.4 Descripción del Sistema Actual

En nuestra actualidad la ESPOCH cuenta con cátedras donde se desarrollan proyectos en equipo, los mismos que están destinados a trabajar bajo limitantes de tiempo, con riesgos de fallos en conexión y restringidos a una red específica, retrasando de esta manera el proceso de desarrollo y quitando valor al proyecto.

4.2.2.1.5 Identificación del Problema

El problema de forma general se presenta como la falta de interacción entre el estudiante y el gestor de proyectos ocasionando problemas en el seguimiento del proyecto.

4.2.2.2. Fase 2: Modelamiento del negocio

4.2.2.2.1 Definir los límites del proceso

Descripción de los Procesos

Tabla IV - I Caso de Uso Autenticar

Identificador Caso de Uso	Caso de Uso Autenticar
Nombre del Caso de Uso	CU_Autenticar
Actores	Administrador, Jefe de Equipo Usuarios(miembros de equipo)
Propósito	Verificar si un usuario esta registrado en el sistema.
Visión General	El sistema verifica si un usuario esta registrado dentro del sistema y le permite o niega el acceso de la aplicación
Tipo	Esencial y expandida.
Referencias	
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. Iniciar cuando el usuario requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su nombre de usuario y password
3. El usuario ingresa su nombre de usuario y password	4. Muestra una ventana de la aplicación
5. El usuario utiliza la aplicación	
Cursos Alternos	
Línea 4: Nombre de usuario y/o password son incorrectas, se indica el error	

Tabla IV - II Caso de Uso Cerrar Sesión

Identificador Caso de Uso	Caso de Uso Cerrar Sesión
Nombre del Caso de Uso	CU_Cerrar_Sesión
Actores	Administrador, Jefe de Equipo Usuarios(miembros de equipo)
Propósito	Permitir que un usuario que está utilizando la aplicación pueda cerrar su sesión
Visión General	El usuario al momento que desee finalizar su sesión correspondiente, después de haber realizado todas sus actividades a las que tiene acceso dentro del sistema, procederá a cerrar la sesión.
Tipo	Esencial y expandido.
Referencias	CU_Autenticar
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El usuario cierra su sesión	2. El sistema cierra la sesión activa y presenta una página para un nuevo inicio de sesión.
Cursos Alternos	
Línea 1: Si el usuario no cierra la sesión correctamente esto puede ser una falta grave de seguridad.	

Tabla IV - III Caso de Uso Agregar Proyecto

Identificador Caso de Uso	Caso de Uso Agregar Proyecto
Nombre del Caso de Uso	CU_Agregar_Proyectos
Actores	Administrador
Propósito	Agregar proyectos a la aplicación
Visión General	El Administrador es el encargado de agregar nuevos proyectos los mismos que tendrán un jefe de proyecto.
Tipo	Esencial, expandido
Referencias	CU_Autenticar
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El Administrador debe autenticarse con su Nombre de usuario y password	2. Visualiza pantalla de actividades del administrador
3. Ingresa al modulo Agregar Proyecto.	4. Ingresa el nombre del proyecto y su descripción
5. Establecer el tiempo de inicio y fin del proyecto.	6. Asigna un Jefe de Proyecto
Cursos Alternos	
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error	
Línea 4,5 y 6: Si los requerimientos no están completos muestra un mensaje de error	

Tabla IV - IV Caso de Uso Agregar Usuarios

Identificador Caso de Uso	Caso de Uso Agregar_Usuario
Nombre del Caso de Uso	CU_Agregar_Usuario
Actores	Administrador
Propósito	Agregar usuarios al Proyecto
Visión General	El Administrador es el encargado de agregar nuevos Usuarios a los proyectos ya existentes.
Tipo	Esencial, expandido
Referencias	CU_Autenticar
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El Administrador debe autenticarse con su Nombre de usuario y password	2. Visualiza pantalla de actividades del administrador
3. Ingresa al modulo Agregar usuarios.	4. Ingresa el nombre del usuario y su descripción
5. Asignar el proyecto al que pertenece el usuario, en el caso de requerirlo.	
Cursos Alternos	
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error	
Línea 4 y5: Si los requerimientos presentados no están completos se visualiza un mensaje de error	

Tabla IV - V Caso de Uso Agregar Requerimiento

Identificador Caso de Uso	Caso de Uso Agregar_Requerimiento
Nombre del Caso de Uso	CU_Agregar_Requerimientos
Actores	Jefe de Proyecto
Propósito	Agregar requerimientos a un proyecto
Visión General	El Jefe de proyecto es el encargado de agregar nuevos requerimientos los mismos pertenecen a un solo proyecto.
Tipo	Esencial, expandido
Referencias	CU_Autenticar
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El Jefe de Proyecto debe autenticarse con su Nombre de usuario y password	2. Visualiza pantalla de actividades del Jefe de Proyecto
3. Ingresa al modulo Agregar Requerimientos.	4. Ingresa el nombre del requerimiento.
5. Establecer el proyecto al que pertenece el requerimiento.	
Cursos Alternos	
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error	
Línea 4 y 5: Si los requerimientos presentados no están completos se visualiza un mensaje de error	

Tabla IV - VI Caso de Uso Agregar Tareas

Identificador Caso de Uso	Caso de Uso Agregar Tareas
Nombre del Caso de Uso	CU_Agregar_Tareas
Actores	Jefe de Proyecto
Propósito	Agregar tareas a los requerimientos del proyecto
Visión General	El Jefe de Proyecto es el encargado de agregar nuevas tareas las cuales serán asignadas a un usuario.
Tipo	Esencial, expandido
Referencias	CU_Autenticar
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El Jefe de Proyecto debe autenticarse con su Nombre de usuario y password	2. Visualiza pantalla de actividades del Jefe de Proyecto.
3. Ingresa al modulo Agregar Tareas.	4. Ingresa el nombre de la tarea.
5. Establecer el requerimiento al que pertenece la tarea.	
Cursos Alternos	
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error	
Línea 4 y 5: Si los requerimientos presentados no están completos se visualiza un mensaje de error	

Tabla IV - VII Caso de Uso Asignar Tareas

Identificador Caso de Uso	Caso de Uso Asignar Tarea
Nombre del Caso de Uso	CU_Asignar_Tarea
Actores	Usuario (Jefe de Equipo)
Propósito	Asignar las tareas a cumplir por cada miembro del equipo.
Visión General	El Jefe de equipo determina la actividad y el tiempo plazo que tiene cada miembro del equipo para desarrollar las tareas.
Tipo	Esencial, expandido
Referencias	CU_Autenticar
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El jefe de equipo debe autenticarse con su Nombre de usuario y password	2. Visualiza pantalla de actividades del jefe de equipo
3. Ingresa al modulo asignar tareas	4. Selecciona la tarea y el usuario al que desea asignar.
5. Establecer el tiempo límite de entrega.	6. Marcar la tarea como Pendiente de entrega.
Cursos Alternos	
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error	
Línea 4,5 y 6: Si los requerimientos presentados no están completos se visualiza un mensaje de error	

Tabla IV - VIII Caso de Uso Subir Tarea

Identificador Caso de Uso		Proceso Subir_Tarea	
Nombre del Caso de Uso		CU_Subir_Tarea	
Actores		Usuario (Miembros de Equipo)	
Propósito		Subir al sistema la tarea que ha sido asignada	
Visión General		Una vez concluida la tarea el miembro del equipo se encargará de subir la tarea que le fue asignada a la aplicación para su revisión y aprobación.	
Tipo		Esencial, expandido	
Referencias		CU_Autenticar. CU_Asignar_Tarea	
Curso Típico de Eventos			
Acción de los Actores		Respuesta del Sistema	
1. El miembro del equipo debe autenticarse con su Nombre de usuario y password		2. Visualiza pantalla de actividades del miembro del equipo.	
3. Ingresa al modulo visualizar tareas pendientes		4. Selecciona la tarea.	
5. Subir la tarea a la aplicación		6. Cambiar el estado de la tarea a Pendiente de revisión.	
7. Enviar notificación al miembro del equipo encargado de realizar las pruebas.			
Cursos Alternos			
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error			
Línea 4,5 y 6: Si los requerimientos presentados no están completos se visualiza un mensaje de error			

Tabla IV - IX Caso de Uso Aprobar Tarea

Identificador Caso de Uso		Caso de Uso Aprobar_Tarea	
Nombre del Caso de Uso		CU_Aprobar_Tarea	
Actores		Usuario (Jefe de equipo)	
Propósito		Aprobar la tarea verificada por el probador	
Visión General		Una vez que la tarea haya sido Verificada y sometida a las pruebas pertinentes por parte del probador, el jefe de equipo cambiara el estado de la tarea según el reporte enviado.	
Tipo		Esencial, expandido	
Referencias		CU_Autenticar. CU_Asignar_Tarea, CU_Subir_Tarea. CU_Probar_Tarea	
Curso Típico de Eventos			
Acción de los Actores		Respuesta del Sistema	
1. El miembro del equipo debe autenticarse con su Nombre de usuario y password		2. Visualiza pantalla de actividades del jefe de equipo.	
3. Ingresa al modulo visualizar tareas probadas.		4. Selecciona la tarea.	
5. Analiza las notificaciones enviadas por el probador del equipo.		6. Cambiar el estado de la tarea a Aprobada o Pendiente de corrección.	
7. Enviar notificación al miembro del equipo responsable de la corrección.			
Cursos Alternos			
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error			
Línea 4,5 y 6: Si los requerimientos presentados no están completos se visualiza un mensaje de error			
Línea 7: en el caso de ser Pendiente de Corrección, cambiar el estado de la tarea a Pendiente de entrega.			

Tabla IV - X Caso de Uso Aprobar Tarea

Identificador Caso de Uso		Caso de Uso Aprobar_Tarea	
Nombre del Caso de Uso		CU_Aprobar_Tarea	
Actores		Usuario (Jefe de equipo)	
Propósito		Aprobar la tarea verificada por el probador	
Visión General		Una vez que la tarea haya sido Verificada y sometida a las pruebas pertinentes por parte del probador, el jefe de equipo cambiara el estado de la tarea según el reporte enviado.	
Tipo		Esencial, expandido	
Referencias		CU_Autenticar. CU_Asignar_Tarea, CU_Subir_Tarea. CU_Probar_Tarea	
Curso Típico de Eventos			
Acción de los Actores		Respuesta del Sistema	
1. El miembro del equipo debe autenticarse con su Nombre de usuario y password		2. Visualiza pantalla de actividades del jefe de equipo.	
3. Ingresa al modulo visualizar tareas probadas.		4. Selecciona la tarea.	
5. Analiza las notificaciones enviadas por el probador del equipo.		6. Cambiar el estado de la tarea a Aprobada o Pendiente de corrección.	
7. Enviar notificación al miembro del equipo responsable de la corrección.			
Cursos Alternos			
Línea 1: Si el usuario o password no son correctos se visualiza mensaje de error			
Línea 4,5 y 6: Si los requerimientos presentados no están completos se visualiza un mensaje de error			
Línea 7: en el caso de ser Pendiente de Corrección, cambiar el estado de la tarea a Pendiente de entrega.			

Tabla IV - XI Proceso Generar Reportes

Identificador Proceso	Proceso Generar_Reportes
Nombre del Proceso	Generar Reportes
Actores	Usuario (Jefe de equipo, miembros del equipo)
Propósito	Permitir generar reportes
Visión General	Los miembros del equipo solicitan generar el reporte.
Tipo	Esencial, expandido
Referencias	CU_Autenticar. CU_Aprobar_Tarea
Curso Típico de Eventos	
Acción de los Actores	Respuesta del Sistema
1. El miembro del equipo debe autenticarse con su Nombre de usuario y password	2. Visualiza pantalla de actividades del miembro de equipo.
3. Solicitar la generación del reporte	4. Presenta un reporte con la información solicitada
5. Permitir la impresión del reporte	6. Imprimir el reporte
Cursos Alternos	
Línea 1: SI el usuario o password no son correctos se visualiza mensaje de error	

4.2.2.3. Fase 3: Estudio de Factibilidad

4.2.2.3.1 Factibilidad Operativa

Para determinar la factibilidad operativa del sistema a desarrollar se basará en el comportamiento de sistema anteriores, permitiendo determinar el comportamiento que tendrá el sistema a desarrollar.

Formulación del Problema

El problema radica principalmente en la gestión de archivos que intervienen en la creación de un proyecto de software, administración de usuarios, creación de requerimientos y tareas designadas a miembros de un equipo y que pueda ser accesible a todos los usuarios a través de cualquier parte del mundo por medio del Internet.

4.2.2.3.2 Factibilidad Técnica

A continuación se presenta la caracterización con respecto al hardware, software y recursos humanos y adicionales que demanda la nueva aplicación con la finalidad de convertir al proyecto en algo técnicamente factible.

- **Identificación del Sistema**

Titulo: Sistema Gestor de Proyectos en Línea.

Nombre Característico: SGP.

- **Recursos humanos involucrados**

Número de Desarrolladores: Dos

Número de asesores: Dos

Número de Usuarios: Varios

- **Recurso Hardware**

Número de Computadoras de desarrollo: Dos

Número de Estaciones de trabajo: Dos

Numero de Dispositivos de impresión: Dos

- **Recurso Software**

Sistema Operativo: Windows 7.

Plataforma: J2EE

ID: Eclipse

- **Grupo de Trabajo**

Las personas que intervienen en el desarrollo del sistema de automatización son:

Director o Coordinador: Ing. Wladimir Castro.

Informáticos: Srta. Ana Guaño

Sra. Gabriela Altamirano

4.2.2.3.3 Factibilidad Económica

Costos Tangibles del Proyecto

Tabla IV - XII Costos Tangibles del proyecto

Cantidad	Descripción	Valor Unitario	Valor Total	Fuente de Financiamiento
Licenciamiento				
1	Windows 7	--	--	Licencia Educativa
1	JDK 6	--	--	Libre
1	Eclipse Helios	--	--	Libre
1	Microsoft Office	\$120	\$120	Propio
Total			\$120	

Beneficios Intangibles del Proyecto

- Mejor calidad del diseño mediante una optimización iterativa.
- Mayor satisfacción de los usuarios debido a un control permanente de las actividades de los proyectos.
- Satisfacción laboral
- Competitividad tecnológica.
- Mayor disponibilidad de Información

4.2.2.4. Fase 4: Exploración

4.2.2.4.1 Historia de Usuario

Se han definido las siguientes historias de usuario:

Tabla IV - XIII Historia de Usuario 1

Descripción del proceso de negocio	
Número: 1	Usuario: Administrador
Nombre proceso: Autenticación de los usuarios	
Prioridad en negocio: Alta	Riesgo de desarrollo: Medio
Puntos Estimados: 3	Interacción asignada: 2
Programador Responsable: Gabriela Altamirano y Ana Guaño	
Todos los usuarios deben autenticarse antes de usar el sistema, este proceso permite ingresar a las opciones propias de cada usuario. Los usuarios existentes en el sistema son: <ul style="list-style-type: none">- Administrador- Jefe de Proyecto- Desarrolladores- Tester	

Tabla IV - XIV Historia de Usuario 2

Descripción del proceso de negocio	
Número: 2	Usuario: Administrador
Nombre proceso: Crear Usuarios	
Prioridad en negocio: Alta	Riesgo de desarrollo: Medio
Puntos Estimados: 3	Interacción asignada: 2
Programador Responsable: Gabriela Altamirano y Ana Guaño	
En primera instancia el Administrador debe iniciar sesión con su nombre de usuario y contraseña, a continuación le permitirá visualizar todos los procesos disponibles para este usuario, siendo el Administrador el único responsable de la creación de usuarios. Una vez ingresado al proceso de creación el administrador deberá llenar todos los campos con información del usuario nuevo y confirmar la creación.	

Tabla IV - XV Historia de Usuario 3

Descripción del proceso de negocio	
Número: 3	Usuario: Administrador
Nombre proceso: Crear Proyecto	
Prioridad en negocio: Alta	Riesgo de desarrollo: Medio
Puntos Estimados: 3	Interacción asignada: 2
Programador Responsable: Gabriela Altamirano y Ana Guaño	
<p>Una vez que existan usuarios disponibles se procederá a crear los proyectos, el responsable de este proceso es el Administrador, el mismo que debe estar previamente autenticado. Una vez creado el proyecto se deberá asignar a un Jefe de Proyecto el cual será el responsable de llevar el control y administración del mismo y de los usuarios que intervengan en el desarrollo.</p>	

Tabla IV - XVI Historia de Usuario 4

Descripción del proceso de negocio	
Número: 4	Usuario: Jefe de Proyecto
Nombre proceso: Crear Requerimientos	
Prioridad en negocio: Alta	Riesgo de desarrollo: Medio
Puntos Estimados: 3	Interacción asignada: 2
Programador Responsable: Gabriela Altamirano y Ana Guaño	
<p>Una vez creados los proyectos y su respectivo responsable. El jefe de Proyecto podrá acceder al sistema con el fin de crear los requerimientos de todos los proyectos disponibles para él. Este proceso es único para este usuario.</p>	

Tabla IV - XVII Historia de Usuario 5

Descripción del proceso de negocio	
Número: 5	Usuario: Jefe de Proyecto
Nombre proceso: Crear Tarea	
Prioridad en negocio: Alta	Riesgo de desarrollo: Medio
Puntos Estimados: 3	Interacción asignada: 2
Programador Responsable: Gabriela Altamirano y Ana Guaño	
<p>El Jefe de Proyecto deberá ingresar al sistema con su nombre de usuario y su contraseña. Una vez en el menú propio del usuario, acceder a la opción Agregar Tarea, este proceso crea las tareas de cada requerimiento disponible en ese proyecto, además en este proceso se debe asignar el usuario responsable de la misma, el cual deberá cumplirla y modificar el estado para verificar su cumplimiento.</p>	

Riesgos.

Identificación de los riesgos

Tabla IV - XVIII Identificación de Riesgos

ID	DESCRIPCIÓN DEL RIESGO
R1	El tiempo disponible para la elaboración del proyecto de tesis no es suficiente para concluir satisfactoriamente con el desarrollo de la solución.
R2	Definición ambigua de requerimientos.
R3	Aparición de nuevos requerimientos una vez que el proyecto está en las fases finales.
R4	Reajuste de cronograma de actividades.
R5	Falta de conocimiento de manejo de la herramienta de desarrollo.
R6	El software no satisface con todas las necesidades previstas por el usuario.
R7	Daños en el hardware determinado para el desarrollo del proyecto.

Análisis de los Riesgos

- Criterios de valoración de la probabilidad

Tabla IV - XIX Criterios de valoración de la probabilidad de riesgos

Rango Probabilidad	Descripción	Valor
1% - 33%	Baja	1
34% - 67%	Media	2
68% - 99%	Alta	3

- Criterios de valoración del impacto

Tabla IV - XX Criterio de valoración del impacto de riesgos

Impacto	Costo	Retraso	Impacto Técnico	Valor
Bajo	<1%	1 semana	Ligero efecto en el desarrollo del proyecto	1
Moderado	<5%	2 semanas	Moderado efecto en el desarrollo del proyecto	2
Alto	<10%	1 mes	Severo efecto en el desarrollo del proyecto	3
Crítico	>10%	Más de 1 mes	Proyecto no puede ser culminado	4

- Determinación de exposición al riesgo

Tabla IV - XXI Determinación de exposición al riesgo

Exposición al riesgo	Valor	Color
Baja	1 o 2	Verde
Media	3 o 4	Amarillo
Alta	>= 6	Rojo

Tabla IV - XXII Valoración de determinación de exposición al riesgo

Impacto Probabilidad	Bajo = 1	Moderada = 2	Alta = 3	Crítico = 4
Alta = 3	3	6	9	12
Media = 2	2	4	6	8
Baja = 1	1	2	3	4

- Tabla de impactos esperados

Tabla IV - XXIII Impacto Esperados

Identificación	Probabilidad			Impacto		Exposición al riesgo	
	%	Valor	Probabilidad	Valor	Impacto	Valor	Exposición
R1	50	2	Media	3	Alto	9	Alto
R2	40	2	Media	2	Medio	6	Alto
R3	10	1	Bajo	3	Alto	3	Medio
R4	10	1	Bajo	4	Crítico	4	Medio
R5	15	2	Medio	3	Alto	3	Medio
R6	5	1	Baja	4	Medio	2	Bajo
R7	5	1	Baja	4	Medio	2	Bajo

- Priorización de los Riesgos

Tabla IV - XXIV Priorización de los Riesgos

ID	EXPOSICIÓN	
	Valor	Exposición
R1	9	Alto
R2	6	Alto
R3	3	Medio
R4	4	Medio
R5	3	Medio
R6	2	Bajo
R7	2	Bajo

- Plan de Reducción, Supervisión y Gestión del Riesgo

Tabla IV - XXV Hoja de Gestión de Riesgo 1

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R1			
Probabilidad: 50 Valor: 2	Impacto: Alto Valor: 3	Exposición: Alta Valor: 9	Prioridad: Alta Valor: 9
Descripción: El tiempo disponible para la elaboración del proyecto de tesis, no es suficiente para concluir satisfactoriamente con el desarrollo de la solución.			
REFINAMIENTO: Causas: <ul style="list-style-type: none"> • Retraso en la ejecución de las actividades planificadas • Falta de cumplimiento del cronograma • Reducción del tiempo Consecuencias: <ul style="list-style-type: none"> • Afecta la calidad del Software • El proyecto no podrá ser entregado en la fecha planificada 			
REDUCCIÓN Y SUPERVISIÓN: <ul style="list-style-type: none"> • Elaborar planes para cumplir las fechas planificadas • Adquirir nuevos horarios que permitan lograr el cumplimiento de las actividades 			
SUPERVISIÓN: <ul style="list-style-type: none"> • Seguimiento continuo en el desarrollo del proyecto • Incrementar procesos de seguridad en cuanto a la verificación del cronograma 			
GESTIÓN: Plan de Contingencia: <ul style="list-style-type: none"> • Realizar la factibilidad técnica antes del desarrollo del proyecto • Incrementar procesos de seguridad en cuanto a la verificación del cronograma 			
ESTADO ACTUAL <ul style="list-style-type: none"> Fase de reducción iniciada <input checked="" type="checkbox"/> Fase de Supervisión iniciada <input type="checkbox"/> Gestionando el riesgo <input type="checkbox"/> 			
RESPONSABLES: Gabriela Altamirano Ana Guaño			

Tabla IV - XXVI Hoja de Gestión de Riesgo 2

HOJA DE GESTIÓN DEL RIESGO			
ID DEL RIESGO: R2			
Probabilidad: 40 Valor: 2	Impacto: Medio Valor: 2	Exposición: Medio Valor: 2	Prioridad: Alta Valor: 6
Descripción: Definición ambigua de requerimientos.			
REFINAMIENTO: Causas: <ul style="list-style-type: none">• Mala comunicación con el cliente• Falta de coordinación entre los miembros del equipo• Acumulación de Información Consecuencias: <ul style="list-style-type: none">• Afecta el desarrollo del sistema• El proyecto no podrá ser entregado en la fecha planificada			
REDUCCIÓN Y SUPERVISIÓN: <ul style="list-style-type: none">• Establecer acuerdos entre los miembros del equipo que lleven al mejoramiento del desarrollo del sistema.• Reorganización de información			
SUPERVISIÓN: <ul style="list-style-type: none">• Seguimiento continuo en el cumplimiento de requerimientos• Evitar modificaciones frecuentes de los requerimientos establecidos.			
GESTIÓN: Plan de Contingencia: <ul style="list-style-type: none">• Establecer claramente los requerimientos con el usuario• Establecer las limitaciones del proyecto de tesis.			
ESTADO ACTUAL Fase de reducción iniciada <input checked="" type="checkbox"/> Fase de Supervisión iniciada <input type="checkbox"/> Gestionando el riesgo <input type="checkbox"/>			
RESPONSABLES: Gabriela Altamirano Ana Guaño			

Prototipo de interfaz

Se va a tomar el siguiente prototipo de interfaz como referencia:

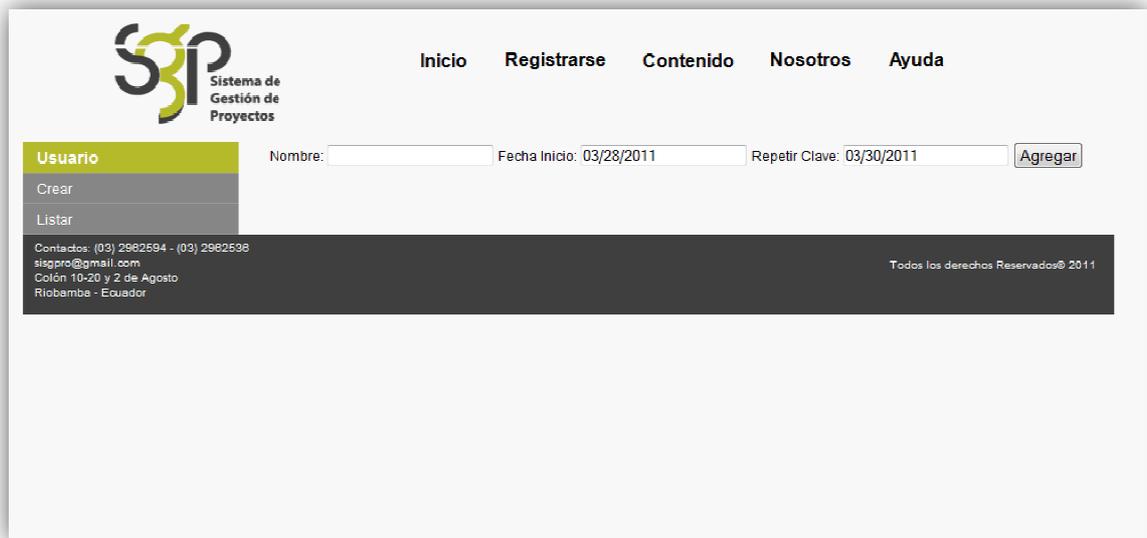


Figura IV - 2 Prototipo

4.2.2.5. Fase 5: Iteración

El desarrollo de la aplicación se detalla mediante iteraciones y los datos obtenidos en cada una de ellas, mediante la cual nos permite la depuración de los requerimientos y las dificultades que se presentan en el desarrollo.

4.2.2.5.1 Iteración 1

Recopilación de Información necesaria para desarrollar el proyecto de tesis.

4.2.2.5.2 Iteración 2

Análisis de la base de datos Bigtable que utiliza Google App Engine para el desarrollo del proyecto de tesis.

4.2.2.5.3 Iteración 3

Generación del primer prototipo funcional, análisis de los problemas presentados y documentación del proceso.

4.2.2.5.4 Análisis de resultados

Iteración 1

Como resultado de la primera iteración se obtuvo una propuesta para el desarrollo del proyecto de tesis.

Iteración 2

Como resultado de esta iteración se obtuvo una conexión exitosa con la base de datos Bigtable.

Iteración 3

Como resultado se obtiene el prototipo funcional del sistema gestión de proyectos versión 1.0.

4.2.2.6. Fase 6: Diseño

- Definición de los requerimientos reales. Ver Anexo 1.
- Refinamiento de los casos de uso. Ver Anexo 2.
- Definición e interacción de formularios para la interfaz de usuario

Página Principal



Figura IV - 3 Página Principal

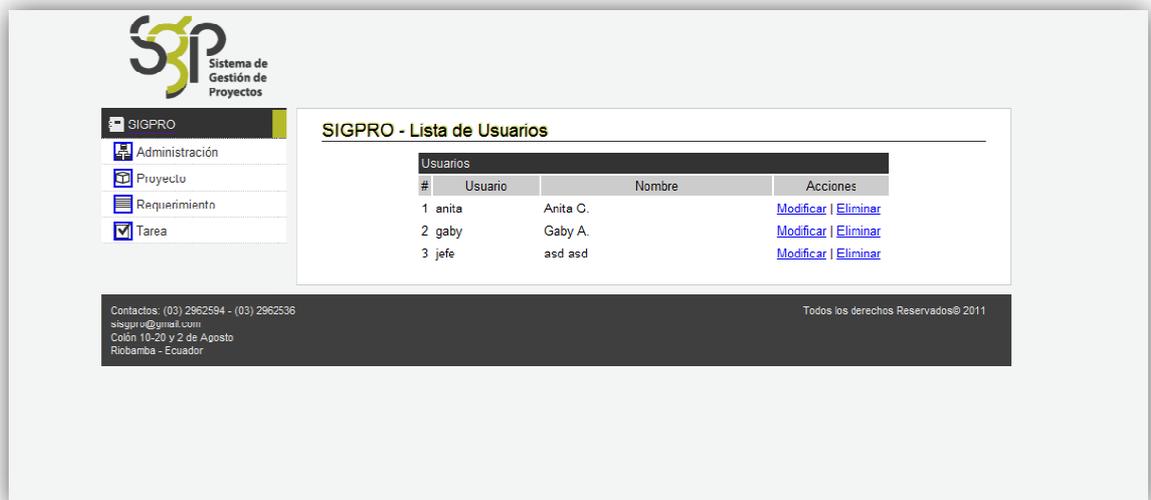


Figura IV - 4 Lista Usuario

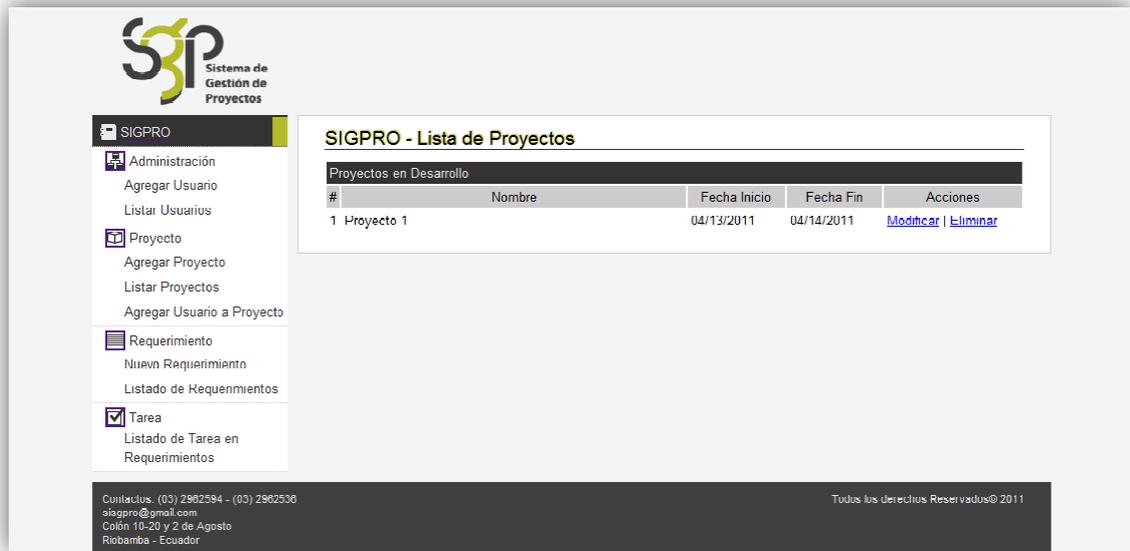


Figura IV - 5 Lista proyectos

Arquitectura lógica general de la aplicación

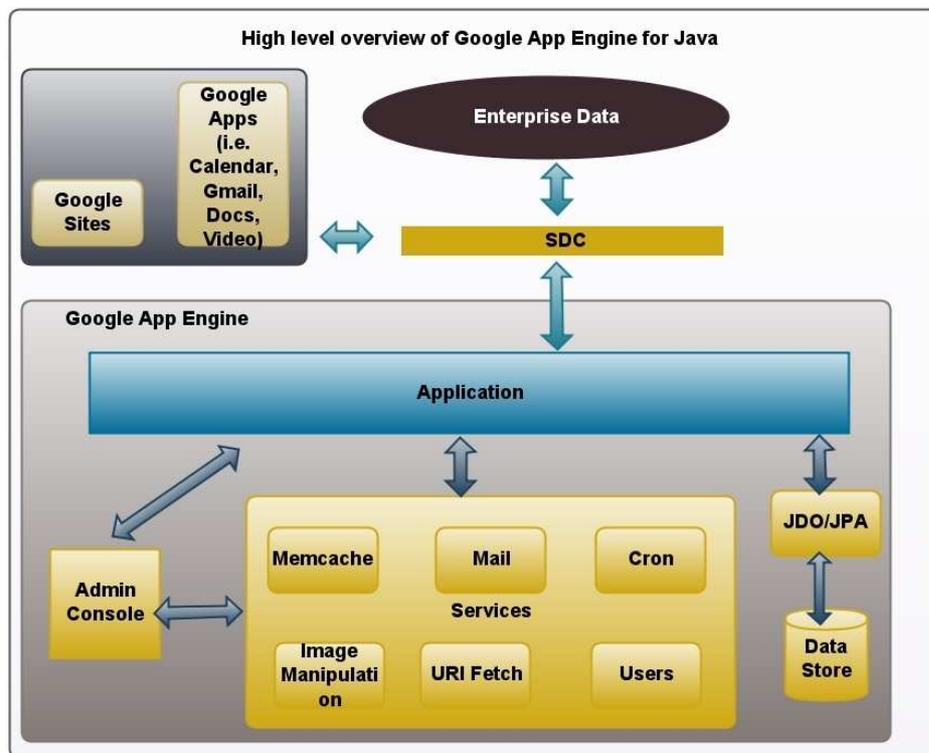


Figura IV - 6 Arquitectura Lógica

Diagrama de Componentes

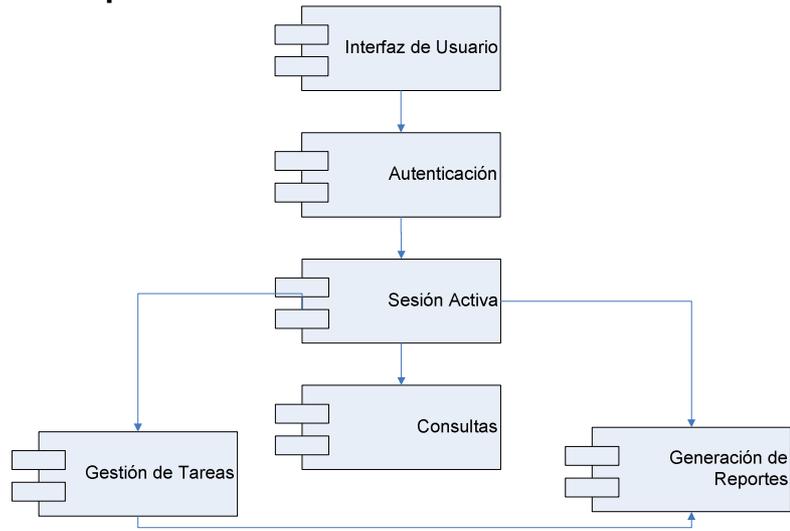


Figura IV - 7. Diagrama de componentes

Arquitectura Física de la Aplicación

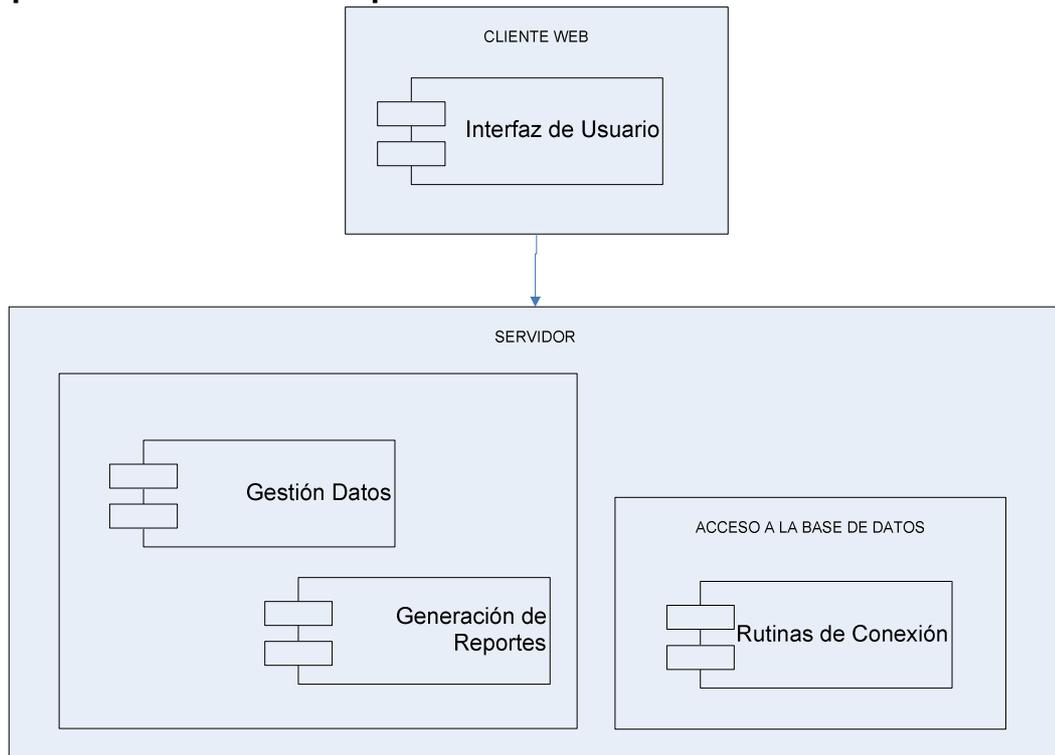


Figura IV - 8 Arquitectura física de la aplicación

Diseño de la Base de Datos

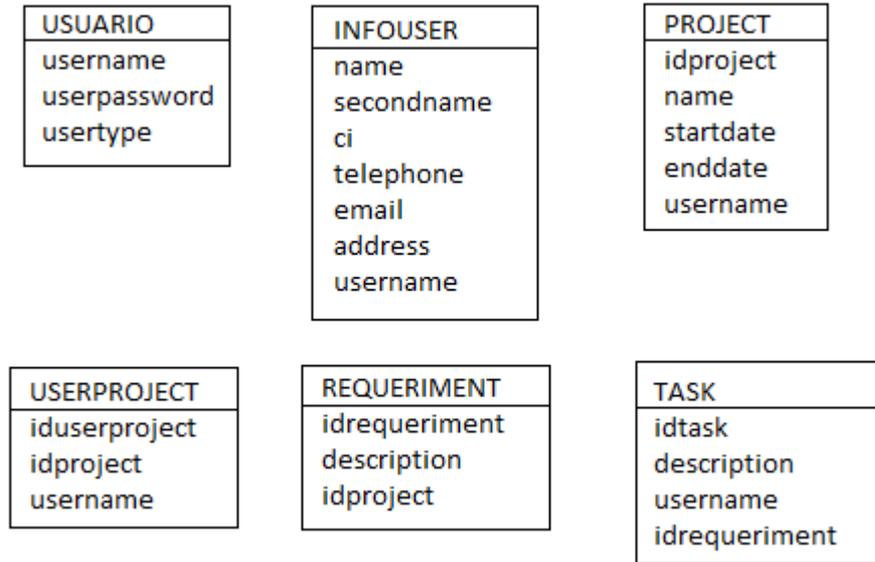


Figura IV - 9. Diagrama de Base de Datos

CONCLUSIONES

Al finalizar el estudio comparativo de los modelos de hosting para Cloud Computing se han obtenido las siguientes conclusiones:

- Se determinó que el modelo Amazon EC2 no es la mejor opción con respecto a la usabilidad de Cloud Computing para implementar un Gestor de Proyectos en Línea.
- Cloud Computing es el futuro de la computación que representa una manera en la cual se puede aprovechar la tecnología de Internet para manipular, conservar y crear información, por esta razón se realizó un análisis comparativo de la usabilidad de modelos hosting para Cloud Computing. Este tipo de aplicación invita a ver el software como un servicio externo. Aunque el Cloud Computing aún está en una fase temprana de desarrollo, estamos seguros que muy pronto alcanzará la edad adulta, ya que cuenta con el soporte de empresas como Google y Amazon.
- Se fijó un conjunto de parámetros en común y de mayor relevancia entre ambos modelos de hosting tomando en cuenta la usabilidad de dichos modelos, los mismos que ayudaron a elegir y realizar la aplicación aprovechando todas las características que nos ofrecen.
- Se aplicó una serie de pruebas a cada uno de los modelos de hosting según los parámetros establecidos para demostrar la complejidad y eficacia en la configuración de cada uno de ellos.
- Se implementó un sistema gestor de proyecto que permite llevar un registro de uno o varios proyectos, requerimientos y tareas los cuales son asignados a usuarios registrados de acuerdo a un proyecto.
- Google App Engine nos da la facilidad de utilizar sus servicios gratuitamente hasta un límite de un Gigabyte de almacenamiento, CPU y ancho de banda

suficiente para servir aproximadamente 5 millones de páginas vistas al mes. Si se requiere más cuota ofrece unas tarifas muy competitivas.

- Google App Engine se encarga de la asignación de servidores y de la configuración importante mientras el usuario no tiene que preocuparse por estos aspectos.
- Amazon ofrece mayor libertad a cambio de una mayor complejidad.
- En el aspecto económico EC2 es un servicio cuya viabilidad hay que estudiar antes de adquirir sus servicios, ya que hay que tener en cuenta el diferente modelo de facturación, hacer planes fiables sobre la cantidad de recursos a necesitar, adaptar la aplicación para sacar provecho a EC2 y finalmente dimensionar el uso a lo que se necesite.

RECOMENDACIONES

Al finalizar el proyecto de tesis, se han establecido las siguientes recomendaciones:

- Los parámetros a evaluar en un estudio comparativo deben ser seleccionados cuidadosamente, y estos deben estar orientados a características que permitan la comprobación de la hipótesis.
- Analizar cuidadosamente los beneficios y características de los modelos de hosting, de manera que su selección permita al máximo sus prestaciones, y por ende obtener el resultado deseado en el trabajo realizado.
- Para un sitio pequeño o una aplicación que no genera un gran tráfico o en la cual no importa que durante los picos de actividad algunos usuarios queden sin atender satisfactoriamente, Amazon EC2 puede no ser una opción viable, ya que un hosting tradicional económico podría servir perfectamente a los propósitos y a un precio muy inferior.
- Google App Engine es una de las mejores opciones al momento de implementar un proyecto o instancia, ya que a pesar de ser gratuito hasta cierto punto brinda las mismas o mejores prestaciones de otros modelos con costos elevados.
- Se debe prestar especial atención a los riesgos relacionados con la seguridad de la información por lo que es necesario realizar una labor de gestión de riesgos que identifique, en los componentes técnicos y regulatorios, las condiciones para el aseguramiento en el buen uso y manejo de la información en un proyecto o instancia.

RESUMEN

Se implementó un sistema gestor de proyectos en base al análisis comparativo de los modelos de hosting para Cloud Computing entre Amazon EC2 y Google App Engine.

El sistema está orientado a usuarios que requieren llevar control de tareas de cada proyecto, con el fin de facilitar la administración de los mismos.

Se utilizaron los métodos científico y sintético con técnicas de comparación, revisión de información y TFA'S (Técnicas de especificación de aplicaciones informáticas).

Se estableció siete parámetros de medición: lenguajes de programación, sistema operativo, base de datos, ambientes de desarrollo, servidor de aplicaciones, costos y configuración, obteniendo los siguientes puntajes: Amazon EC2 con 54,38% y Google App Engine con 87,72%, siendo el de mayor usabilidad el modelo Google App Engine con una equivalencia de Muy bueno. Implementando el sistema gestor de proyectos con este último en la aplicación Java Eclipse y Bigtable para el almacenamiento de datos, el mismo que cuenta con un diseño de ambientes amigables y es de fácil uso para el usuario.

Se determinó que la herramienta desarrollada obtuvo excelentes resultados en cuanto a usabilidad y control de tareas de proyectos prueba, lográndose alcanzar el objetivo.

Se recomienda para la realización de proyectos en línea usar el hosting de Google App Engine ya que ofrece mejores beneficios.

SUMMARY

A management system of projects is implemented based on a comparative analysis of hosting models for Cloud Computing between Amazon EC2 and Google App Engine.

The system is oriented to users who requires to get task control of each Project in order to make their management easier.

Scientific and synthetic methods were used with comparison techniques, information revision and TFA's (Techniques of informatics applied specifications).

Seven measure parameters were established: program languages, operative system, data base, development environments, application server, costs and configuration, getting the following marks: Amazon EC2 54,38% and with the most usability Google App Engine 87,72% with an equivalent of Very Good. Implementing the management system of projects with the last one on Java Eclipse and Bigtable application to store data, it counts with a design and friendly environment and it is very easy to the user.

It was determined that the developed tool got excellent results according to usability and control of test project tasks for getting the proposal.

It is recommended for line projects to use hosting of Google App Engine which offers better benefits.



ANEXO

Anexo 1. Requerimientos

Requerimientos de Seguridad

- Req(1): Privacidad en los usuarios como: Administrador, Jefe de Proyecto, Desarrollador y Diseñador.

Requerimientos de Funcionalidad

- Req(2): Interfaz 100% intuitiva y permitir una fácil navegación
- Req(3): Reportes estadísticos.

Requerimientos de Interfaz de Desarrollo

- Req(4): Uso de Tecnologías actuales como Eclipse que nos permite realizar pruebas en un servidor virtual para comprobar los errores y luego efectuar la publicación del sistema.

Requerimientos de Documentación

- Req(5): Características de Ayuda, como los respectivos manuales.
- Req(6): Informe de gestión del proyecto
- Req(7): Documentación técnica (Proceso de desarrollo con la metodología ágil xp).

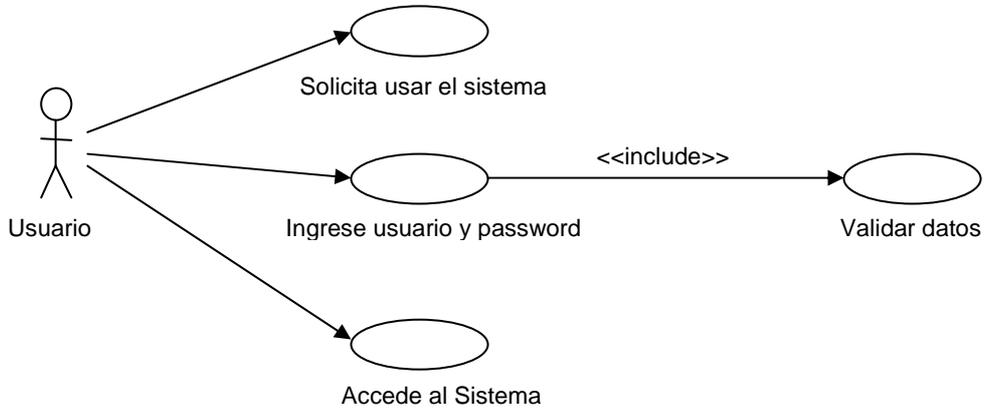
Requerimientos Funcionales

- Req(8): Autenticación de los usuarios (LOGIN)
- Req(9): Menú de opciones por cargo
- Req(10): Agregar usuario
- Req(11): Listado de usuarios con cargo en determinado proyecto
- Req(12): Modificar usuario
- Req(13): Eliminar usuario

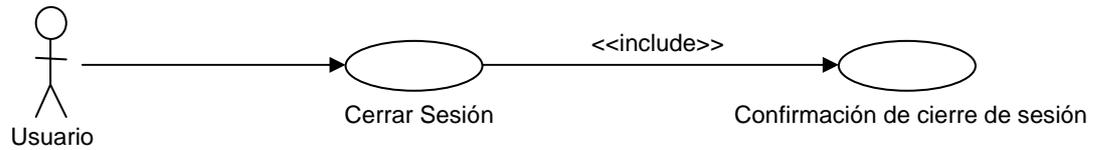
- Req(14): Agregar proyecto con usuario (jefe de proyecto)
- Req(15): Listado de proyectos
- Req(16): Modificar proyecto
- Req(17): Eliminar proyecto
- Req(18): Listado de tareas con estado y prioridad mas porcentaje completado
- Req(19): Asignar usuario a proyecto
- Req(20): Listado de usuario en proyecto mas cargo
- Req(21): Modificar usuario en proyecto
- Req(22): Eliminar usuario en proyecto
- Req(23): Agregar requerimiento
- Req(24): Listado de requerimientos
- Req(25): Modificar requerimiento
- Req(26): Eliminar requerimiento
- Req(27): Asignar tarea a usuario
- Req(28): Listado de tareas
- Req(29): Modificar tarea
- Req(30): Eliminar tarea
- Req(31): Listado de tareas con estado y prioridad
- Req(32): Detalle de tarea con opción cambio de estado (Desarrolladores/Diseñadores)
- Req(33): Detalle de tarea con opción cambio de estado (Tester)
- Req(34): Detalle de requerimiento opción de visualización de archivos
- Req(35): Cerrar sesión

Anexo 2. Diagramas de Caso de Uso

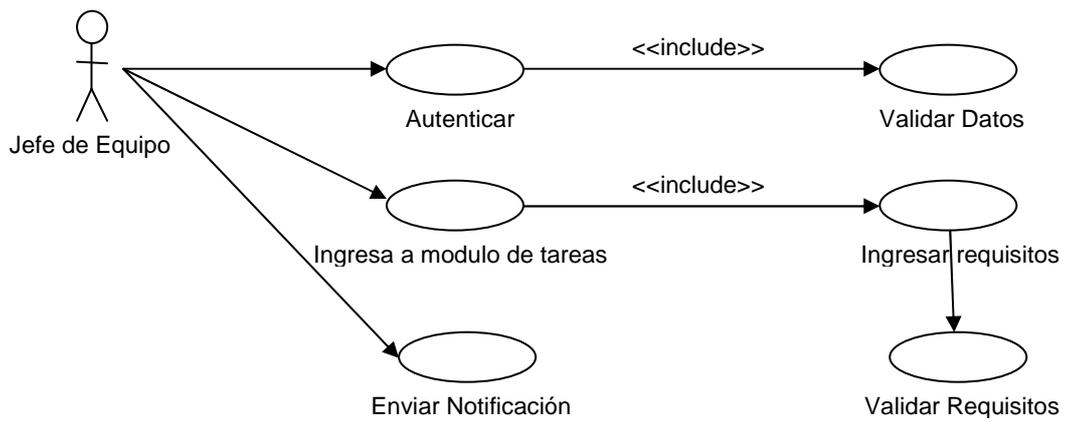
Caso de uso Autenticar



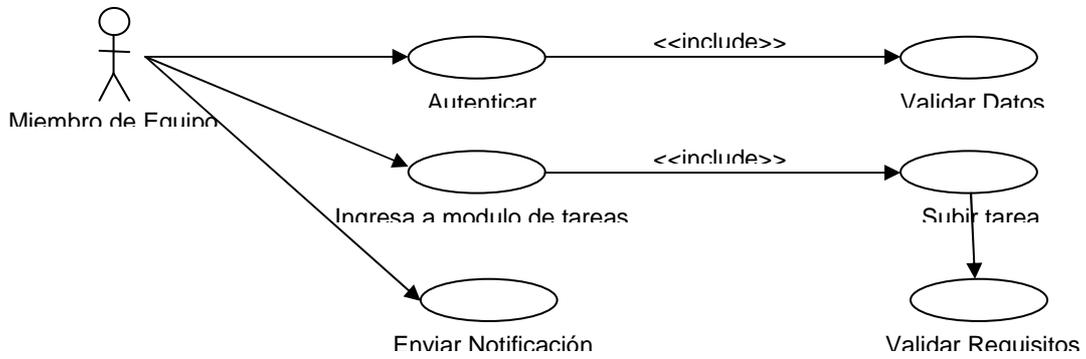
Caso de Uso Cerrar Sesión



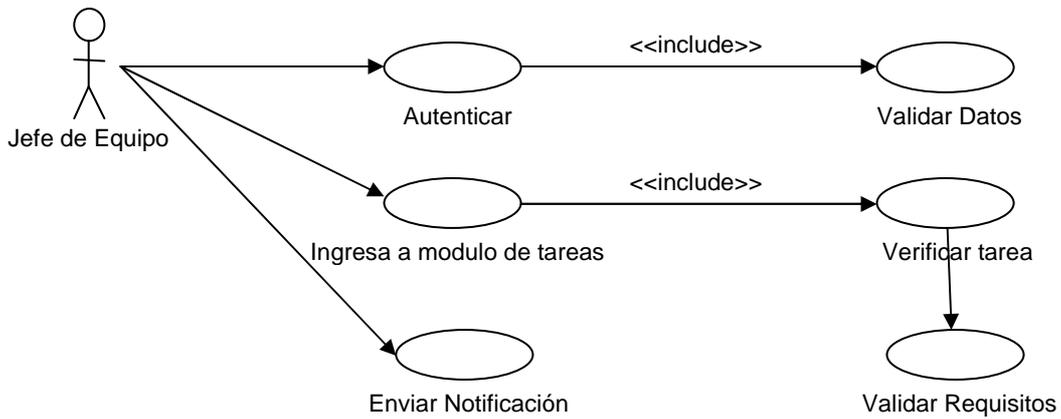
Caso de Uso Agregar Tarea



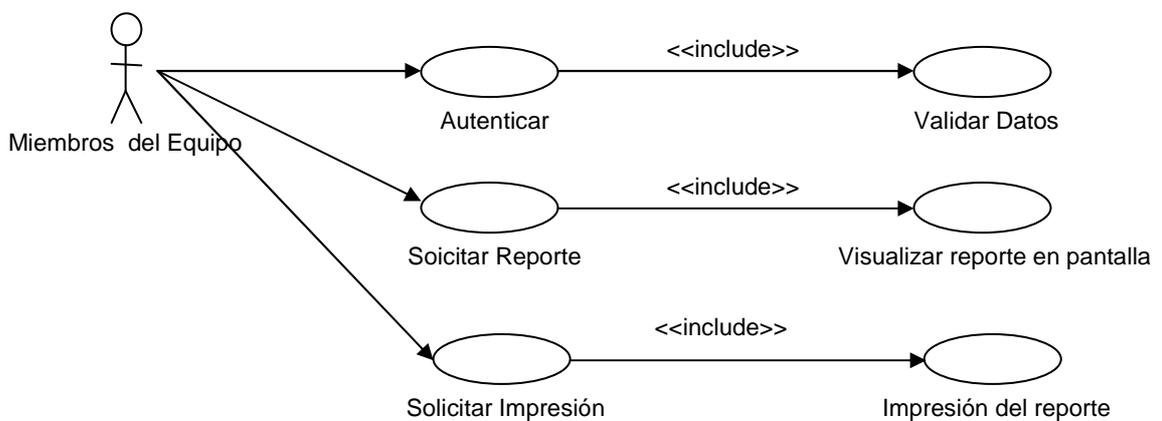
Caso de Uso Subir tarea



Caso de Uso Aprobar Tarea



Caso de Uso Generar Reportes



Anexo 3. Encuestas Google App Engine y Amazon EC2



Encuesta para determinar la usabilidad del hosting para Cloud Computing¹⁴

*Obligatorio

Qué nivel de dificultad considera usted que tuvo al abrir una cuenta en Google App Engine? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al configurar un nuevo proyecto/Instancia en Google App Engine? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al publicar un proyecto/Instancia en Google App Engine? *

- Muy difícil
- Difícil

¹⁴ <https://spreadsheets.google.com/viewform?formkey=dHpnNmpPbkRPcDVycWZEU3cxN0dHbVE6MQ&output=html>

- Medianamente fácil
- Fácil
- Muy fácil

Cómo considera usted la integración con servicios externos (autenticación, documentos, email) con Google App Engine? *

- Muy Difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al momento de escalar un proyecto/Instancia en Google App Engine? *

- Muy Difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al momento de utilizar al panel de administración en Google App Engine? *

- Muy Difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Cómo considera usted los costos para un proyecto/instancia en Google App Engine? *

- Muy costoso

- Costoso
- Medianamente económico
- Económico
- Sin costo

Cuál es la cantidad de personal adecuada para administrar un proyecto/instancia en Google App Engine? *

- 1 a 5 Personas
- 5 a 10 Personas
- 10 o más Personas

Cuál es la cantidad de tiempo promedio para publicar un proyecto/instancia en Google App Engine? *

- 1 a 5 Horas
- 5 a 10 Horas
- 10 o más Horas

Cómo considera usted el rendimiento de Google App Engine con CDN? *

- Muy Bueno
- Medianamente Bueno
- Bueno
- Regular
- Malo



Encuesta para determinar la usabilidad del hosting para Cloud Computing

*Obligatorio

Qué nivel de dificultad considera usted que tuvo al abrir una cuenta en Amazon EC2? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al configurar un nuevo proyecto/instancia en Amazon EC2? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al publicar un proyecto/instancia en Amazon EC2? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al momento de escalar un proyecto/instancia en Amazon EC2? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Cómo considera usted la integración con servicios externos (autenticación, documentos, email) con Amazon EC2? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Qué nivel de dificultad considera usted que tuvo al momento de utilizar el panel de administración en Amazon EC2? *

- Muy difícil
- Difícil
- Medianamente fácil
- Fácil
- Muy fácil

Cómo considera usted los costos para un proyecto/instancia en Amazon EC2? *

- Muy costoso
- Costoso
- Medianamente económico
- Económico
- Sin Costo

Cuál es la cantidad de personal adecuada para administrar un proyecto/instancia en Amazon EC2? *

- 1 a 5 Personas
- 5 a 10 Personas
- 10 o más Personas

Cuál es la cantidad de tiempo promedio para publicar un proyecto/instancia en Amazon EC2? *

- 1 a 5 Horas
- 5 a 10 Horas
- 10 o más Horas

Cómo considera usted el rendimiento de Amazon EC2 con CDN? *

- Muy Bueno
- Medianamente Bueno
- Bueno
- Regular
- Malo

GLOSARIO

TI (Tecnologías de la Información): es el estudio, diseño, desarrollo, implementación, soporte o dirección de los sistemas de información computarizados, en particular de software de aplicación y hardware de computadoras. Se ocupa del uso de las computadoras y software para convertir, almacenar, proteger, procesar, transmitir y recuperar la información.

Grids: es una tecnología innovadora que permite utilizar de forma coordinada todo tipo de recursos (entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado. En este sentido es una nueva forma de computación distribuida, en la cual los recursos pueden ser heterogéneos

Batches: es un archivo de procesamiento por lotes. Se trata de archivos de texto sin formato, guardados con la extensión BAT que contienen un conjunto de comandos MS-DOS.

XAAS: El término se refiere a un número cada vez mayor de servicios que se entregan sobre la red en vez de ser provistos localmente o en sitio. XaaS es la esencia del cómputo en la nube.

SAAS: Software as a Service (Software como servicio) es un término utilizado para describir el software desplegado en Internet y se caracteriza por que el proveedor licencia la aplicación al suscriptor en un modelo de "servicio por demanda".

IAAS: Infrastructure as a Service (Infraestructura como servicio) es el tercer modelo de implementación de Cloud Computing y hace referencia a plataformas que ofrecen infraestructura de cómputo y usualmente se encuentran desplegadas sobre un entorno de virtualización.

PAAS: Platform as a Service (Plataforma como Servicio) se refiere a un modelo que no sólo ofrece la plataforma de despliegue y adicionalmente una plataforma de desarrollo de aplicaciones completa.

SimpleDB: es un motor de bases de datos, similar a MySQL, Amazon SimpleDB es capaz de brindar la solución de almacenamiento para aplicaciones, CMS y casi cualquier script que requiera guardar información.

SQS (Software Quality System): es la empresa líder en servicios de Testing y Calidad de Software.

QoS (Calidad de Servicio): son las tecnologías que garantizan la transmisión de cierta cantidad de información en un tiempo dado.

SCSI (Sistema de Interfaz para Pequeñas Computadoras): es una interfaz estándar para la transferencia de datos entre distintos dispositivos del bus de la computadora.

Endpoints: Un extremo de la red es el punto final lógico de tráfico de protocolo por separado de una capa de protocolo específico.

Amazon EBS: Amazon Elastic Block Store ofrece volúmenes de almacenamiento altamente disponibles y fiables que se pueden adjuntar a una instancia de Amazon EC2 en ejecución y se pueden exponer como un dispositivo dentro de la instancia.

CloudWatch: es un servicio web que proporciona supervisión para los recursos en nube de AWS.

Amazon Simple Storage Service: Amazon S3 es almacenamiento para Internet. Está diseñado para facilitar a los desarrolladores la informática a escala web.

Amazon VPC: Virtual Private Cloud: es un puente seguro y sin fisuras entre la infraestructura de IT existente de una empresa y la nube de AWS.:

AMIS: Amazon Machine Images. Son las imágenes que se utilizan en Amazon Para arrancar instancias. El concepto de las amis es similar las máquinas virtuales de otros sistemas.

Instancias Spot: es una nueva forma de comprar y consumir instancias de Amazon EC2

BIBLIOGRAFÍA

ROCHE, Kyle and DOUGLAS, Jeff. Beginning Java Google App Engine. New York, Paul Manning, 2009. 265 p.

SANDERSON, Dan. Programming Google App Engine. Sebastopol CA – USA, Sumita Mukherji, 2010. 392 p.

CLOUD COMPUTING

- **Aspectos Técnicos**

<http://translate.google.com.ec/translate?hl=es&langpair=en|es&u=http://www.microsoft.com/windowsazure/windowsazure/>

(20100622)

<http://translate.google.com.ec/translate?hl=es&langpair=en|es&u=http://www.salesforce.com/force/>

(20100622)

http://www.idg.es/pcworldtech/Cloud-computing:-_que-hay-en-la-nube_-_/art194631-comunicaciones.htm

(20100624)

- **Características**

<http://www.mkm-pi.com/mkmpi.php?article3828>

(20100624)

<http://www.infoworld.com/d/cloud-computing/what-cloud-computing-really-means-031?page=0,0>

(20100624)

<http://es.calameo.com/read/0001476363464a9395672>

(20100624)

- **Clases**

<http://www.nubeblog.com/2008/10/15/saas-iaas-y-paas-las-tres-clases-de-cloud-computing/>

(20100622)

- **Definición**

<http://redindustria.blogspot.com/2009/10/introduccion-cloud-computing.html>

(20100618)

<http://www.sensiblecloud.com/what-is-cloud-computing/>

(20100622)

<http://www.scribd.com/doc/25483344/Cloud-Computing>

(20100621)

<http://www.scribd.com/doc/25323224/Computacion-en-Nubes>

(20100621)

<http://www.dreig.eu/caparazon/2008/10/30/%C2%BFque-es-el-cloud-computing-definicion-tendencias-y-precauciones/>

(20100622)

Evolución

<http://www.maestrosdelweb.com/editorial/cloud-computing-nueva-era-de-desarrollo/>

(20100622)

<http://translate.google.com.ec/translate?hl=es&langpair=en|es&u=http://rationalsecurity.typepad.com/blog/2008/11/james-urquhart-posted-a-summary-a-week-or-so-ago-of-what-he-called-described-as-the-big-4-players-in-cloud-computing-it-was.html>

(20100622)

<http://lanoticiatecnologicadelasemana.wordpress.com/2008/10/26/cloud-computing-todo-a-la-nube/>

(20100624)

- **Servicios**

<http://120linux.com/instalando-servicios-ii-introduccion-a-cloud-computing/>

(20100618)

<http://blog.espol.edu.ec/taws/category/web-services/>

(20100620)

ECLIPSE

http://es.wikipedia.org/wiki/Eclipse_%28software%29

(20100628)

<http://amap.cantabria.es/confluence/display/DEV/Eclipse>

(20100628)

NETBEANS

<http://vramirez.wordpress.com/2009/02/01/5-plugins-sencillos-pero-muy-utiles-para-netbeans-65/>

(20100628)

<http://foros.3dgames.com.ar/programacion.97/353631.eclipse-vs-netbeans.html>

(20100628)

AMAZON EC2

- **Características**

http://es.wikipedia.org/wiki/WebSphere_Application_Server

(20100705)

http://es.wikipedia.org/wiki/Sun_Java_System_Application_Server

(20100705)

http://es.wikipedia.org/wiki/Amazon_EC2

(20100707)

- **Configuración**

http://es.wikipedia.org/wiki/Fedora_%28distribuci%C3%B3n_Linux%29

(20100710)

<http://es.wikipedia.org/wiki/CentOS>

(20100707)

- **Definición**

<http://aws.amazon.com/ec2/>

(20100705)

<http://translate.google.com.ec/translate?hl=es&sl=en&u=http://aws.amazon.com/ec2/&ei=Y64OTJ-UO4GC8galtuzzCA&sa=X&oi=translate&ct=result&resnum=1&ved=0CCoQ7gEwAA&prev=/search%3Fq%3DAmazon%2BEC2%26hl%3Des%26newwindow%3D1%26prmd%3DI>

[UO4GC8galtuzzCA&sa=X&oi=translate&ct=result&resnum=1&ved=0CCoQ7gEwAA&prev=/search%3Fq%3DAmazon%2BEC2%26hl%3Des%26newwindow%3D1%26prmd%3DI](http://translate.google.com.ec/translate?hl=es&sl=en&u=http://aws.amazon.com/ec2/&ei=Y64OTJ-UO4GC8galtuzzCA&sa=X&oi=translate&ct=result&resnum=1&ved=0CCoQ7gEwAA&prev=/search%3Fq%3DAmazon%2BEC2%26hl%3Des%26newwindow%3D1%26prmd%3DI)

(20100705)

- **Servicios**

<http://es.wikipedia.org/wiki/GNU/Linux>

(20100707)

http://es.wikipedia.org/wiki/Windows_Server

(20100707)

<http://es.wikipedia.org/wiki/Ubuntu>

(20100707)

GOOGLE APP ENGINE

- **Características**

<http://code.google.com/intl/es-ES/appengine/docs/billing.html>

(20100802)

<http://code.google.com/intl/es-ES/appengine/docs/whatisgoogleappengine.html>

(20100802)

<http://osum.sun.com/group/usolsv/forum/topics/trabajando-con-google-web>

(20100802)

http://eliax.com/?post_id=4671

(20100804)

- **Configuración**

http://translate.google.com.ec/translate?hl=es&sl=en&u=http://code.google.com/appengine/&ei=6KAWTM2PG8P_lgeHrO3xDA&sa=X&oi=translate&ct=result&resnum=2&ved=0CCkQ7gEwAQ&prev=/search%3Fq%3Dgoogle%2Bapps%2Bapp%2Bengine%26hl%3Des%26sa%3DX%26prmd%3Dv

(20100806)

<http://code.google.com/intl/es/appengine/docs/quotas.html>

(20100806)

<http://code.google.com/intl/es-ES/appengine/whyappengine.html>

(20100806)

<http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=735>

(20100806)

<http://blog.pablo-morales.com/2010/06/resumen-de-lo-aprendido-con-amazon-ec2/>

(20100806)

http://translate.google.com.ec/translate?hl=es&langpair=en|es&u=http://www.ferg.org/projects/python_java_side-by-side.html

(20100806)

<http://code.google.com/intl/es-ES/appengine/docs/java/gettingstarted/>

(20100806)

http://code.google.com/intl/es/appengine/docs/java/tools/eclipse.html#Installing_the_Google_Plugin_for_Eclipse

(20100810)

<http://code.google.com/intl/es/appengine/docs/java/gettingstarted/installing.html>

(20100810)

- **Definición**

<http://www.error500.net/google-app-engine>

(20100802)

http://www.error500.net/web_20?page=11

(20100802)

<http://www.coplec.org/?q=2009/06/20/google-app-engine>

(20100802)

<http://www.skripsi->

[thesis.com/site/http://digg.com/programming/Bootstrap_your_Google_App_Engine_application](http://www.skripsi-thesis.com/site/http://digg.com/programming/Bootstrap_your_Google_App_Engine_application)

(20100802)

- **Servicios**

http://translate.google.com.ec/translate?hl=es&sl=en&u=http://en.wikipedia.org/wiki/Google_App_Engine&ei=-dQOTNzCOYT68Abfy8SBCg&sa=X&oi=translate&ct=result&resnum=9&ved=0CFsQ7gEwCA&prev=/search%3Fq%3Dgoogle%2Bapp%2Bengine%26hl%3Des%26newwindow%3D1%26prmd%3Dnl

(20100804)

<http://code.google.com/intl/es-ES/appengine/docs/whatisgoogleappengine.html>

(20100805)

<http://www.eforcers.com/portafolio/servicios/plataforma-google/desarrollo-de-aplicaciones-con-google-app-engine/caractersticas>

(20100805)

<http://es.debugmodeon.com/articulo/introduccion-a-google-app-engine>

(20100805)

GESTOR DE PROYECTO

<http://sourceforge.net/projects/collabtive/>

(20101008)

<http://www.cmstecno.com/collabtive-para-trabajar-en-equipo>

(20101008)

<https://www.wrike.es/workspace.htm>

(20101009)