

8^{VO} CONGRESO NACIONAL INGENIERÍA INFORMÁTICA / SISTEMAS DE INFORMACIÓN

VIRTUAL 2020 CONAISI



www.sanfrancisco.utn.edu.ar

05 | NOV.
06



150
ING

la Argentina celebra
su ingeniería
1870-2020



**Octavo Congreso Nacional de Ingeniería
Informática/Sistemas de información
CONAIISI**

5 y 6 de noviembre de 2020

Universidad Tecnológica Nacional, Facultad
Regional San Francisco

Memoria de Trabajos

Claudia Verino, Juan Carlos Calloni, Gabriel Cerutti, Alfonsina E. Andreatta
(Compiladores)

San Francisco, Córdoba - Argentina, Marzo de 2021

Octavo Congreso Nacional de Ingeniería Informática/Sistemas de Información: CONAIISI.
5 y 6 de noviembre de 2020 / Claudia Verino, Juan Carlos Calloni, Gabriel Cerutti,
Alfonsina E. Andreatta; compilado por Claudia Verino , Juan Carlos Calloni, Gabriel Cerutti,
Alfonsina E. Andreatta. - 1a ed. -
Ciudad Autónoma de Buenos Aires: Universidad Tecnológica Nacional.
Facultad Regional San Francisco, 2021.
Libro digital, PDF

Archivo Digital: descarga
ISBN 978-950-42-0202-8

1. Sistemas de Información. 2. Ingeniería Informática. I. Verino, Caludia, Calloni, Juan
Carlos, Cerutti, Gabriel, Andreatta, Alfonsina E. comp.
CDD 004.07

Octavo Congreso Nacional de Ingeniería Informática/Sistemas de información
CONAIISI
5 y 6 de noviembre de 2020
Universidad Tecnológica Nacional, Facultad Regional San Francisco

Memorias de trabajo

Diseño de Tapa: María Laura Vaudagna

ISBN 978-950-42-0202-8



Auspiciantes:



metodo_SCGE validando las tres primeras etapas en un organismo gubernamental

Marcelo Castro, Claudia Fabiana Pons, Rocío Andrea Rodríguez

Facultad de Ingeniería, Universidad Nacional de Jujuy

mcastro.fi.unju@gmail.com

Laboratorio de Investigación y formación en Informática Avanzada

Universidad Nacional de La Plata

cpons@lifa.info.unlp.edu.ar

Departamento de Ingeniería e Investigaciones Tecnológicas

Universidad Nacional de la Matanza

rrodriguez@ing.unlam.edu.ar

Resumen

En el presente trabajo se realiza la validación de las tres primeras etapas correspondientes a la metodología para sistematizar y estandarizar los procesos de Gobierno Electrónico en la gestión pública, a través de servicios computacionales (metodo_SCGE). Esta metodología considera software, hardware y comunicaciones, el análisis de estos componentes permite generar un modelo integral basado en Servicios Computacionales para Gobierno Electrónico (SCGE). El artículo contiene una presentación de algunas metodologías existentes que se pueden aplicar a servicios computacionales, una breve descripción de metodo_SCGE, incluyendo componentes, características, etapas y actividades. Al final se presenta la validación de las tres primeras etapas, sobre un servicio específico perteneciente a un organismo gubernamental.

1. Introducción

En general la tarea que implica formalizar un proceso de gestión de las TIC, en forma global, si bien es ambicioso; resulta muy interesante intentar controlar y estandarizar algunos procesos, como por ejemplo la sistematización del proceso de Gobierno Electrónico (GE). Una gran cantidad de bibliografía relacionada a la gestión de GE, como las guías de los gobierno de Ecuador [1] intenta proponer alguna secuencia de pasos a seguir para la implementación del mismo.

El objetivo del trabajo es validar las tres primeras etapas de metodo_SCGE, que integra las herramientas de gestión y los servicios al ciudadano, con conceptos de teoría de las organizaciones, como así también la utilización de los aspectos relacionados a la Calidad del

Software, agregando una estimación de los recursos computacionales; como procesamiento, almacenamiento y conectividad, a través de los componentes de los Servicios Computacionales de Gobierno Electrónico (SCGE).

Con el desarrollo de metodo_SCGE, se espera obtener una metodología basada en SCGE que permita analizar, diseñar e implementar soluciones de GE. Para el desarrollo de metodo_SCGE se tuvieron en cuenta otras metodologías de análisis y diseño de software existentes, que van desde modelos de desarrollo estructurado con metodologías orientadas al flujo de datos, hasta modelos de desarrollo orientados a objetos como el Lenguaje Unificado de Modelado (UML)[2] o más recientemente las Arquitecturas orientadas a Servicios (SOA)[3], la metodología desarrollada propone integrar en un SCGE, tanto los aspectos relacionados al software, como a la estimación de los recursos computacionales.

2. Algunas metodologías que resultan aplicables a servicios computacionales

En este apartado se detallan una serie de metodologías que se tuvieron en cuenta al momento de diseñar metodo_SCGE:

- a. El “Mapa para el desarrollo del GE en los países en desarrollo” [4], realizado por el Pacific Council of International Policy, establece una serie de preguntas fundamentales que los gobiernos deben hacerse antes de encarar algún proyecto de GE.
- d. En relación a la estrategia de un proyecto de gobierno electrónico basado en el servicio, los autores Affisco y Soliman [5] proponen en su trabajo las siguientes etapas: segmentación del mercado, diferenciación de un producto o de un servicio, modelo de servicio para el proyecto gobierno electrónico, generación de valor, rediseño de operaciones,

alineación del sistema y promoción de buenas prácticas, entrega de servicios y productos.

f. En la Universidad de Ciencia y Tecnología de Kaohsiung en Taiwan [6], los investigadores del área desarrollaron un trabajo en el cual los autores invitaron a todas las agencias del sector público involucradas en el estudio, a integrar sus procesos de servicio en un modelo de ventanilla única y establecer un mecanismo colaborativo para proporcionar un servicio personalizado a los ciudadanos.

g. El proyecto Access-eGov ha desarrollado herramientas de software que permiten la integración de servicios utilizando tecnologías semánticas [7].

h. En el trabajo “Integración de Modelos de Negocios Electrónicos para Soluciones Gubernamentales: Metodologías y Procesos Orientados a Servicios centrados en el Ciudadano”, los autores Chhabra y Kumar [8] proporcionan una colección única de herramientas esenciales y estrategias alternativas de comercio electrónico que se pueden incorporar a las soluciones de GE.

k. Dzhusupova, Shareef y Janowski [9] proponen una metodología para la Evaluación de la Preparación para GE que tiene como objetivo abordar los requisitos de información para la planificación estratégica mediante la evaluación de la demanda y las capacidades del gobierno electrónico en contextos nacionales e internacionales, así como el entorno propicio y la infraestructura de las TIC.

Además existen tecnologías como la Arquitectura Orientada a Servicios (SOA) [3] y la Arquitectura Dirigida por Modelos (MDA) [10], entre otras, que realizan aportes significativos en el desarrollo de software. Existen aspectos que podrían complementarse a las citadas metodologías como son la estimación de los recursos computacionales y por otra parte se hace necesaria una metodología específica para el diseño de soluciones de gobierno electrónico.

Teniendo en cuenta lo expuesto anteriormente resulta necesario establecer una metodología al momento de decidir incorporar una TIC y en especial GE, asegurando que el proceso tendrá en cuenta aspectos relacionados no sólo a la tecnología informática, sino también a la teoría de las organizaciones, la gestión pública, como así también a la utilización de los conceptos relacionados a la Calidad y aspectos sociales; como por ejemplo la participación ciudadana.

3. metodo_SCGE: una metodología basada en SCGE

La metodología metodo_SCGE se encuentra basada en servicios computacionales, por lo tanto se definirá lo que es un servicio computacional de gobierno electrónico (SCGE) en el marco conceptual de este trabajo. Un SCGE es una unidad de servicio que se pone a disposición de una solución de gobierno electrónico. Una solución de gobierno electrónico puede estar compuesta por varios SCGE y estos a su vez conformar varias soluciones de gobierno electrónico. A continuación se describirán los tres componentes que conforman a un SCGE.

3.1 Componentes

Un SCGE está compuesto por:

- Una parte dirigida al análisis, diseño, prueba e implementación del software de gobierno electrónico; denominada Componente Software. Este componente puede poseer una o más funcionalidades específicas del dominio de gobierno electrónico, también contiene la descripción del software de base utilizado, el lenguaje o herramienta de desarrollo utilizada, el sistema Gestor de Base de Datos y la arquitectura de software implementada.
- Un elemento relacionado con la estimación de los recursos computacionales necesarios para soportar el software de gobierno electrónico, denominado Componente Hardware y Conectividad. Este componente se refiere a los recursos de hardware y conectividad asociados a la solución diseñada. Aunque este componente puede considerarse como uno solo, el mismo se dividió en dos, lo que permite separar los recursos necesarios para efectuar el procesamiento y almacenamiento de datos por una parte y la transferencia de datos a través de medios de comunicación por otra.

En consecuencia un SCGE podría graficarse de la siguiente manera (Ver Figura 1):

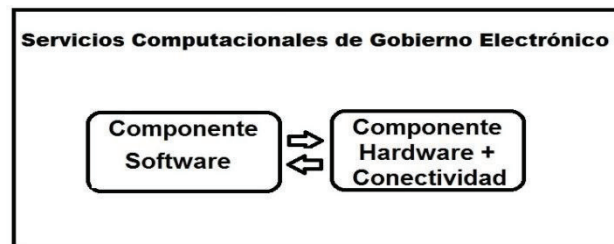


Figura 1. Componentes de un Servicio computacional de Gobierno Electrónico

3.1.1 Componente Software

Este componente es de suma importancia debido a que contiene las funcionalidades concretamente definidas para el SCGE. Estas funcionalidades se encuentran directamente relacionadas con las especificadas en el trabajo “Análisis de las propiedades y atributos propios de sitios de gobierno electrónico” [11] y representan el núcleo del SCGE. También en este elemento se deben especificar el Software de base que utiliza el SCGE, es decir el sistema operativo nativo sobre el cual se encuentra soportado el servicio, el lenguaje de programación utilizado para generar el código fuente o generado por un entorno integrado de desarrollo (IDE), o una herramienta de ayuda para el desarrollo de aplicaciones (CASE) utilizada para el desarrollo del servicio. Conjuntamente se deberá determinar, el sistema

gestor de Base de Datos que se utilizará para soportar el almacenamiento, modificación y extracción de la información en la base de datos elegida, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos contenidos en la misma. Por último se deberá describir la Arquitectura de Software que soportará el SCGE, es decir si se tratará de una arquitectura host-terminal, cliente servidor o multicapa. En la Figura 2 puede observarse la estructura del Componente Software.

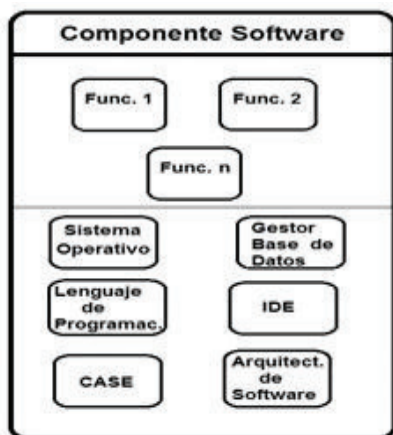


Figura 2. Estructura del Componente Software

3.1.2 Componente Hardware

Aquí se deberá definir el equipamiento que soportará el SCGE diseñado. En primer lugar resulta indispensable consignar la unidad central de procesamiento (CPU) que se utilizará, como así también el tipo y la capacidad de la memoria RAM requerida, el tipo y tamaño del soporte de almacenamiento magnético que resulte necesario. Por último es necesario establecer el tipo y características de las interfaces que se usarán. Para poder definir convenientemente este componente resultará adecuado realizar una estimación de los recursos computacionales necesarios, tales como tipo y capacidad de procesamiento y almacenamiento. En la Figura 3 se puede visualizar la estructura del componente descrito.

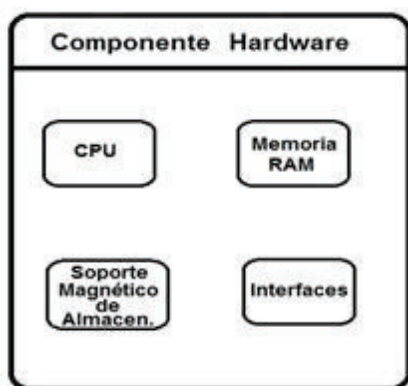


Figura 3. Estructura del Componente Hardware

3.1.3 Componente Conectividad

En este componente se define la manera en que el SCGE se encuentra conectado con otros sistemas y servicios computacionales, se deberá definir el tipo de enlace físico a utilizar, la velocidad de conexión, el protocolo de comunicación y el tipo de red. En la figura 4 se puede observar la estructura del componente de conectividad.

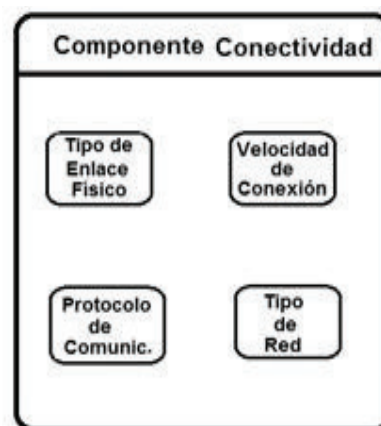


Figura 4. Estructura del Componente Conectividad

3.2 Características de la metodología basada en SCGE (metodo_SCGE)

La metodología metodo_SCGE, se trata de un método iterativo e incremental de desarrollo basado en SCGE. Al igual que las metodologías ágiles tradicionales para el desarrollo de software, como Scrum y XP, metodo_SCGE realiza incrementos e iteraciones para agregar funcionalidades y características de software, hardware y conectividad para el SCGE que se está desarrollando.

La metodología propone desarrollar un SCGE no como una sola unidad, sino como una serie de incrementos y cada una de ellos incluye una nueva característica de software, hardware y conectividad para el SCGE. En metodo_SCGE, los procesos de especificación, diseño e implementación están entrelazados, y el SCGE se desarrolla en diferentes versiones.

Al igual que las metodologías ágiles metodo_SCGE, considera al diseño y la implementación como actividades centrales, incorporando a éstas, otras actividades como la adquisición de requerimientos y las pruebas. La iteración ocurre a través de las actividades. En consecuencia la adquisición de requerimientos y el diseño se desarrollan en conjunto, no por separado.

3.3 Etapas de metodo_SCGE

La metodología metodo_SCGE posee cuatro etapas, las cuales se describen a continuación:

1. Seleccionar características de Componentes SCGE de la versión a liberar: en esta etapa se seleccionarán las características de software, hardware y conectividad que se desean incorporar en la nueva versión que se liberará.

2. Desglosar características de cada componente: en esta etapa se deben determinar claramente las características que se desean desarrollar para cada componente.

3. Desarrollar, utilizar, integrar y probar cada característica por componente: aquí se realizará el desarrollo, la integración y las pruebas de cada característica correspondiente a cada uno de los componentes de un SCGE.

4. Liberar y evaluar versión del SCGE: durante esta etapa se procederá a la liberación y evaluación de la versión desarrollada para el SCGE.

En la figura 5 se puede observar un gráfico de las etapas de metodo_SCGE.

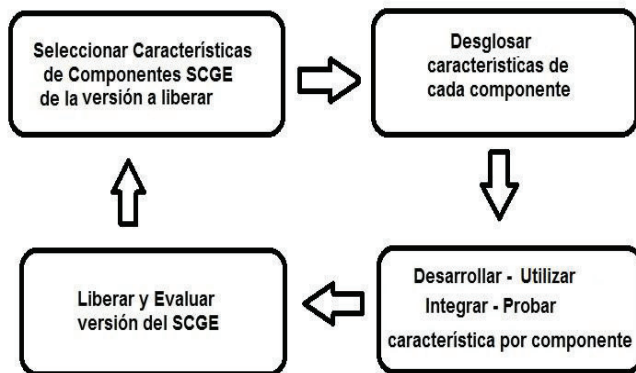


Figura 5. Etapas de metodo_SCGE

3.4 Actividades de metodo_SCGE

En la tabla 1, se detallan las actividades y sub-actividades correspondientes a los componentes de software, hardware y conectividad, que se encuentran incluidos en cada una de las etapas de la metodología propuesta:

Etapa	Actividad	Componente	Sub-Actividad
I- Seleccionar características de Componentes SCGE de la versión a liberar	I.1-Seleccionar característica CS	Software	I.1.1-Seleccionar y priorizar funcionalidades específicas del dominio.
			I.1.2-Seleccionar una herramienta Case y/o entorno IDE
			I.1.3-Seleccionar un lenguaje de programación en que se generará el código fuente del SCGE
			I.1.4-Seleccionar una Arquitectura de Software a utilizar
			I.1.5-Seleccionar un Gestor de Base de Datos
	I.2-Seleccionar característica CH	Hardware	I.2.1-Seleccionar una Unidad Central de Procesamiento
			I.2.2-Seleccionar un Tipo y capacidad de memoria RAM
			I.2.3-Seleccionar un tipo y una capacidad de almacenamiento magnético
	I.3-Seleccionar característica CC	Conectividad	I.2.4-Seleccionar un tipo de interfaz
			I.3.1-Seleccionar un tipo de enlace físico
			I.3.2-Seleccionar una velocidad de conexión
II-Desglosar características de cada componente	II.1-Desglosar característica CS	Software	I.3.3-Seleccionar un protocolo de comunicación
			I.3.4-Seleccionar un tipo de red
			II.1.1-Especificar funcionalidades del SCGE.
			II.1.2-Especificar características de la herramienta Case y/o entorno IDE, seleccionadas.
			II.1.3-Especificar características del lenguaje de programación seleccionado
	II.1-Desglosar característica CH	Hardware	II.1.4-Especificar características de la Arquitectura de Software seleccionada
			II.1.5-Especificar características del Gestor de Base de Datos

			seleccionado
	II.2-Desglosar característica CH	Hardware	II.2.1- Especificar características de la Unidad Central de Procesamiento
			II.2.2- Especificar características de la memoria RAM
			II.2.3- Especificar características del almacenamiento magnético
			II.2.4- Especificar características de las interfaces
	II.3-Desglosar característica CC	Conectividad	II.3.1- Especificar características del enlace físico
			II.3.2- Especificar características de la velocidad de conexión
			II.3.3- Especificar características del protocolo de comunicación
			II.3.4- Especificar características del tipo de red
III-Desarrollar, utilizar, integrar y probar cada característica por componente	III.1- Desarrollar, utilizar, integrar y probar CS	Software	III.1.1- Desarrollar, integrar y probar funcionalidades del SCGE.
			III.1.2-Utilizar y probar la herramienta Case y/o entorno IDE, seleccionadas.
			III.1.3- Utilizar y probar el lenguaje de programación seleccionado
			III.1.4- Utilizar y probar la Arquitectura de Software seleccionada
			III.1.5- Utilizar y probar Gestor de Base de Datos seleccionado
	III.2-Desarrollar, utilizar, integrar y probar CH	Hardware	III.2.1- Utilizar y probar Unidad Central de Procesamiento
			III.2.2- Utilizar y probar la memoria RAM
			III.2.3- Utilizar y probar el almacenamiento magnético
			III.2.4- Utilizar y probar las interfaces
	III.3-Desarrollar, utilizar, integrar y probar CC	Conectividad	III.3.1- Utilizar y probar el enlace físico
			III.3.2- Utilizar y probar la velocidad de conexión
			III.3.3- Utilizar y probar el protocolo de comunicación
			III.3.4- Utilizar y probar el tipo de red
IV-Liberar y evaluar versión del SCGE	IV.1 – Implementar y evaluar la versión liberada del SCGE, para los tres componentes	Software, Hardware y Conectividad	IV.1.1 - Implementar y evaluar la versión liberada para el componente Software
			IV.1.2 - Implementar y evaluar la versión liberada para el componente Hardware
			IV.1.3 - Implementar y evaluar la versión liberada para el componente Conectividad

Tabla1. Actividades por etapas de metodo_SCGE

4. Validación de la metodología propuesta aplicándola a un caso de estudio en un organismo gubernamental.

4.1 Introducción

La metodología metodo_SCGE será validada a través de su aplicación en la Dirección Provincial de Rentas de la Provincia de Jujuy, la cual depende de la Secretaría de Ingresos Públicos perteneciente al Ministerio de Hacienda.

Para la aplicación de metodo_SCGE se utilizarán algunos de los servicios a brindar al ciudadano y que se encuentra detallados en el Árbol de requerimientos consignados en el trabajo “Análisis de las propiedades y atributos propios de sitios de gobierno electrónico”[11] para la característica “Funcionalidad” de los sitios de GE,

y específicamente para la sub-característica “Funcionalidad específica del dominio y contenido”, atributo “Poder Ejecutivo” y sub-atributo “Servicios públicos, finanzas y producción”.

4.2 Desarrollo de la etapa I

Teniendo en cuenta lo consignado en la Tabla 1, a continuación se desarrollarán las actividades previstas para la etapa I.

Etapa I: Seleccionar características de Componentes SCGE de la versión a liberar

I.1.Actividad: Seleccionar característica CS

I.1.1-Seleccionar y priorizar funcionalidades específicas del dominio:

- Declaración Jurada de Impuestos
- Pago de impuestos y tarifas en línea

- Denuncias de evasión Tributaria
 - Consultar Informe de Deuda Tributaria
- I.1.2-Seleccionar una herramienta Case y/o entorno IDE
- El entorno de desarrollo integrado (IDE) seleccionado es Eclipse SDK 4.15 (64-bit)
- I.1.3-Seleccionar un lenguaje de programación en que se generará el código fuente del SCGE
- El lenguaje de programación seleccionado es Java EE versión 8.0.
- I.1.4-Seleccionar una Arquitectura de Software a utilizar
- La arquitectura seleccionada es una orientada a servicios (SOAP).
- I.1.5-Seleccionar un Gestor de Base de Datos
- El gestor de Base de Datos seleccionado es Oracle versión 19c
- I.2-Seleccionar característica Hardware CH
- I.2.1-Seleccionar una Unidad Central de Procesamiento
- CPU seleccionada Intel® Core™ i7 de 10ma Generación
- I.2.2-Seleccionar un Tipo y capacidad de memoria RAM
- Memoria RAM 16GB, 2666MHz
- I.2.3-Seleccionar un tipo y una capacidad de almacenamiento magnético
- Disco Rígido interno, 1TB, Sata III.
- I.2.4-Seleccionar un tipo de interfaz
- Pantalla táctil LED 18.5 “,TFT, 1366 px x 768 px
- I.3-Seleccionar característica Conectividad CC
- I.3.1-Seleccionar un tipo de enlace físico
- Inalámbrico
- I.3.2-Seleccionar una velocidad de conexión
- 1000Mbps o superior
- I.3.3-Seleccionar un protocolo de comunicación
- Protocolo TCP/IP
- I.3.4-Seleccionar un tipo de Red
- Red de área local (LAN).

En la figura 6 se puede visualizar el detalle de los ítems seleccionados para los componentes de software, hardware y conectividad.

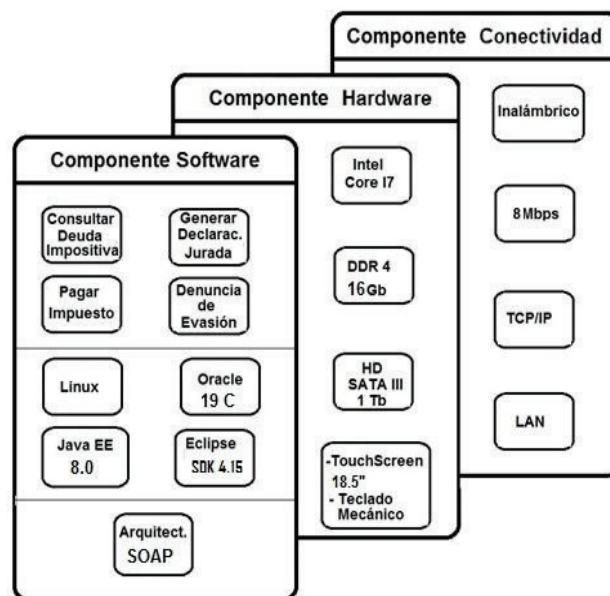


Figura 6. Valores seleccionados para los componentes de software, hardware y conectividad en la etapa I

4.3 Desarrollo de la etapa II

Teniendo en cuenta lo consignado en la Tabla 1, a continuación se desarrollarán las actividades previstas para la etapa II.

Etapa II: Desglosar características de cada componente

II.1. Desglosar característica de Software (CS)

II.1.1- Especificar funcionalidades del SCGE:

- Declaración Jurada de Impuestos: Se debe consignar información de los plazos para la presentación de declaraciones de impuestos. Además de deberá permitir descargar los formularios a presentar. El objetivo final será efectuar la declaración jurada de impuestos totalmente en línea.
- Pago de impuestos y tarifas en línea: Estas funcionalidades de suma importancia ya que es un servicio indispensable para el ciudadano o empresa y debe permitir realizar la transacción de pago, totalmente en tiempo real; además requiere complementarse con técnicas de pago electrónico a través de diferentes medios.
- Denuncias de evasión Tributaria: Esta funcionalidad permite registrar las denuncias realizadas por los ciudadanos a través de correo electrónico y todos los canales de comunicación que posea el organismo gubernamental.
- Consultar Informe de Deuda Tributaria: esta funcionalidad deberá permitir realizar consulta en línea e impresión del detalle de la deuda de un contribuyente.

II.1.2- Especificar características de la herramienta Case y/o entorno IDE, seleccionadas.

El entorno de desarrollo integrado (IDE) Eclipse SDK 4.15 (64-bit) cuenta con las siguientes características principales:

- Desarrollador: Fundación Eclipse
- Autor: ObjectTechnology International
- Fecha de Lanzamiento: 07/11/2001 -
Última versión estable: 17/06/2020 (Versión 4.16)
- Género: IDE, Java SDK, C/C++ -Lenguaje nativo: Java
- Sistema operativo: Multiplataforma -
Plataforma: Máquina virtual Java
- Licencia: Publica de Eclipse -Idioma: Español

II.1.3- Especificar características del lenguaje de programación seleccionado

El lenguaje de programación seleccionado Java EE versión 8.0, posee algunas de las características que se mencionan a continuación:

- Versión: Java™ SE Development Kit 8, Update 251 (JDK 8u251)
- Fecha actualización: 14/04/2020
- Full versión string: 1.8.0_251-b08
- Número de versión: 8u251.
- Actualización JDK:8u251 implement JSR 337

II.1.4- Especificar características de la Arquitectura de Software seleccionada

La arquitectura orientada a servicios SOA es la que mejor se adapta al caso de estudio elegido, debido a que existen algunas partes de software existente que puede ser utilizado. Al tratarse de aplicaciones que se encuentran en un entorno y dominio específico, en el que las operaciones no sufrirán cambios significativos, la implementación de los web services basados en SOAP puede resultar muy conveniente.

II.1.5- Especificar características del Gestor de Base de Datos seleccionado

El gestor de Base de Datos seleccionado Oracle posee las siguientes características:

- Versión: Oracle Database 19c
- Fecha actualización: 10/01/2019
- Sistema operativo: Multiplataforma
- Desarrollador: Oracle Corporation
- Cantidad de licencias a adquirir: 300

II.2- Desglosar característica de Hardware (CH)

II.2.1- Especificar características de la Unidad Central de Procesamiento

La CPU seleccionada Intel® Core™ i7 de 10ma Generación posee las siguientes características:

- Procesador: Intel® Core™ i7-10700K
- Cantidad de núcleos: 8
- Frecuencia básica del procesador: 3.80 GHz
- Frecuencia turbo máxima: 5.10 GHz
- Tipos de memoria soportada: DDR4
- Caché: 16 Mb Intel Smart

- Subprocesos: 16
- Gráficos del procesador: Gráficos HD Intel® 630

II.2.2- Especificar características de la memoria RAM

La memoria seleccionada posee las siguientes características:

- Tipo de memoria interna: DDR4
- Capacidad: 16 GB
- Diseño de memoria: 1 x 16 GB
- Velocidad de memoria del reloj: 2666 MHz
- Factor de Forma: 288-pin DIMM

II.2.3- Especificar características del almacenamiento magnético

Las características del Disco Rígido seleccionado son:

- Disco duro interno
- Capacidad de almacenamiento: 1 TB.
- Caché 64 MB.
- Interfaz: SATA III
- Velocidad de transferencia interna: 210 Mbps.
- Consumo: 5,3 W.
- Dimensiones: 146,99 x 101,6 x 20,17 mm.
- Peso aproximado: 400 g.

II.2.4- Especificar características de las interfaces

Las características de la Pantalla táctil seleccionada son:

- Tipo de panel: Aluminio de magnesio.
- Screen: LCD de 18,5", retroillum. por LED.
- Resolución: 1366*768.
- Procesadores: Intel Baytrail J1900 2.0Ghz.
- Tacto proyectivo: capacitivo multipunto.
- Grado de protección IP65.
- soporte de montaje: VESA 75/100.
- Entrada de 9~24V DC.
- I/O: 4x USB, 2x GLAN, 2x COM, 1x HDMI, Audio.

II.3- Desglosar característica de Conectividad (CC)

II.3.1- Especificar características del enlace físico

Las características del enlace físico seleccionado son:

- Tecnología: inalámbrica
- Medio: ondas de radiofrecuencia de baja potencia y banda específica
- Uso: privado.

II.3.2- Especificar características de la velocidad de conexión

Para lograr una buena velocidad se utilizarán las siguientes características:

- Frecuencia: 2,4 GHz
- Velocidad máxima teórica: 11 Mbps
- Estándar inalámbrico: IEEE 802.11b

II.3.3- Especificar características del protocolo de comunicación

El Protocolo TCP/IP seleccionado es el que mejor se adapta a las velocidades y configuración de la red utilizada.

II.3.4- Especificar características del tipo de red

La LAN utilizará estructura cableada existente y se adicionarán tres puntos de acceso (AP) Ubiquiti Dual Band 802.11ac 2.4/5.8ghz, para conformar una configuración inalámbrica que soporte el diseño especificado.

4.4 Desarrollo de la etapa III

En función a lo establecido en la Tabla 1, a continuación se desarrollarán las actividades previstas para la etapa III.

Etapa III: Desarrollar, utilizar, integrar y probar cada característica por componente

III.1- Desarrollar, utilizar, integrar y probar Software

III.1.1- Desarrollar, integrar y probar funcionalidades del SCGE.

Para el desarrollo se utilizó el IDE Eclipse, generando un modelo con generación de código Java. A continuación se pueden visualizar algunas de las pantallas principales de la aplicación desarrollada en las figuras 7,8,9,10,11 y 12.

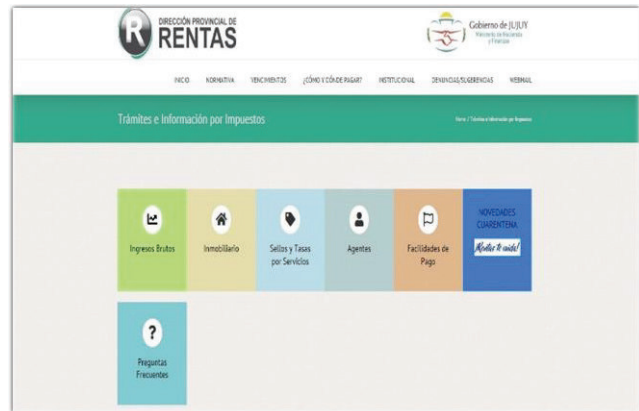


Figura 9. Menú Tarifas e información de impuestos

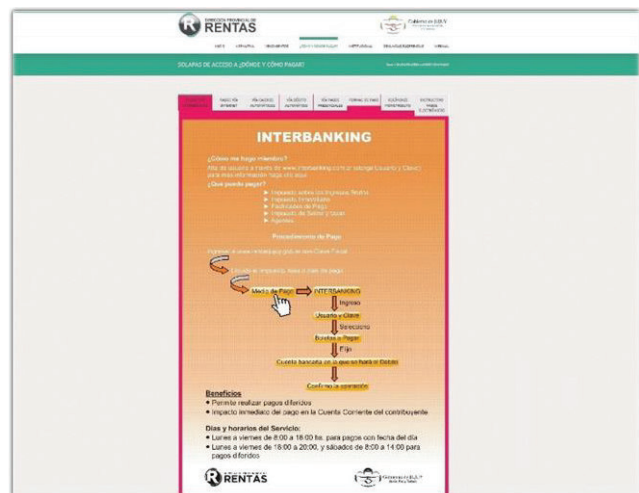


Figura 10. Pago de impuestos en línea

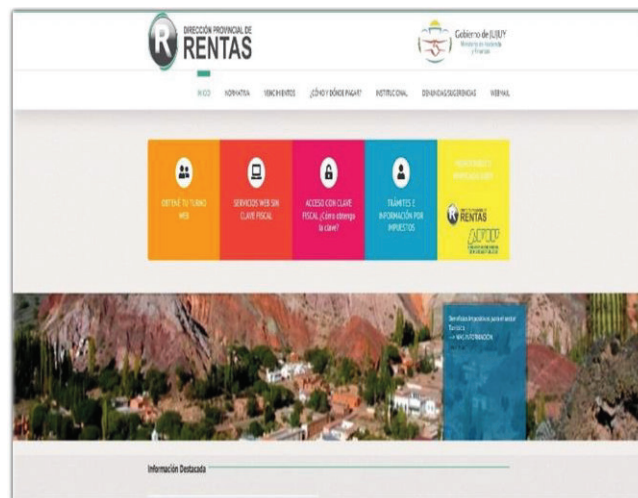


Figura 7. Menú principal

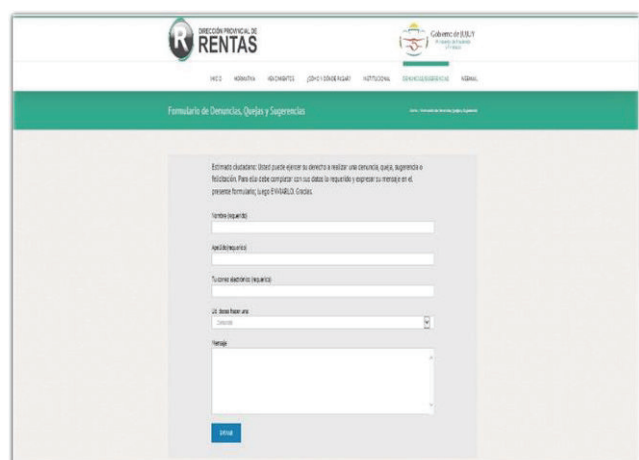


Figura 11. Formulario de Denuncias

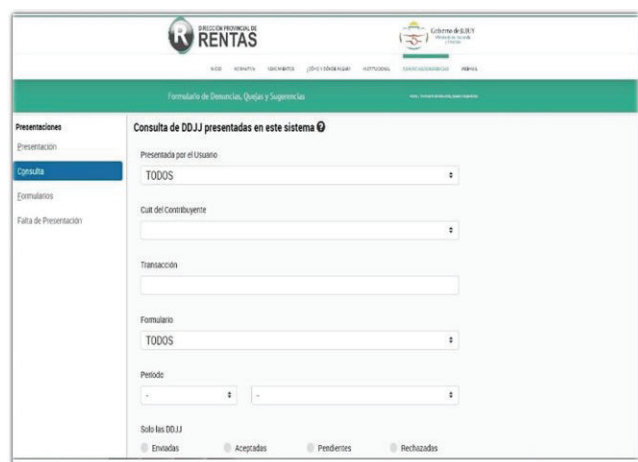


Figura 8. Declaración Jurada



Figura 12. Menú Consultas

III.1.2-Utilizar y probar la herramienta Case y/o entorno IDE, seleccionadas.

Se utilizó el entorno IDE, para desarrollar las aplicaciones correspondientes a las funcionalidades especificadas oportunamente. Aunque el entorno IDE se comportó adecuadamente, se tuvieron que agregar algunos plugins para generar gráficos.

II.1.3- Utilizar y probar el lenguaje de programación seleccionado

Al tratarse de una generación automática de código a partir del IDE, el lenguaje seleccionado resultó estable y no presentó problemas nativos. Los inconvenientes surgieron con la necesidad de instalar algunos plugins.

II.1.4- Utilizar y probar la Arquitectura de Software seleccionada

A través de la arquitectura SOA, se propuso utilizar Servicios Web SOAP para acceder a las funcionalidades que no se encuentran desarrolladas para entorno web. Sin embargo, si una determinada repartición quisiera ejecutar algunos procesos no lo podría hacer; en consecuencia se propone que a través de Servicios Web SOAP se permitan a las distintas reparticiones ejecutar las funcionalidades que se encuentran desarrolladas en Java.

Para lo cual se hace necesario efectuar una invocación de las siguientes funcionalidades que se encuentran alojadas en el Application Server.

En la Fig. 13 se puede apreciar el esquema propuesto.

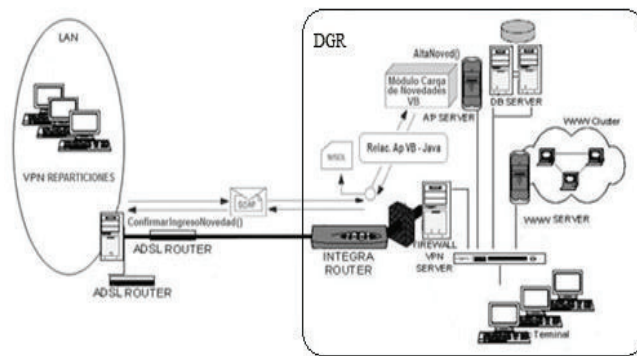


Figura 13. Arquitectura Web SOAP propuesta

II.1.5- Utilizar y probar Gestor de Base de Datos seleccionado

El DBMS seleccionado se comportó estable y adecuado para el volumen de las transacciones previstas. No presentó inconvenientes al momento de generar la base de datos correspondiente al modelo de datos desarrollado.

III.2-Desarrollar, utilizar, integrar y probar característica de Hardware

III.2.1-Utilizar y probar Unidad Central de Procesamiento

El procesador Intel® Core™ i7 de 10ma Generación se comportó adecuadamente en las pruebas realizadas, brindando un excelente tiempo de respuesta teniendo en cuenta la carga de trabajo a la cual fue sometida. Se realizó una prueba de estrés con usuarios accediendo simultáneamente a las distintas funcionalidades, obteniéndose una prestación adecuada en función a los núcleos, frecuencia y subprocesos del procesador seleccionado.

III.2.2- Utilizar y probar la memoria RAM

Si bien el tipo de memoria resultó adecuada, se podría pensar en duplicar la capacidad para tener un margen de holgura óptimo. Por otro lado la Velocidad de memoria del reloj es conveniente para el tipo y cantidad de transacciones utilizadas.

III.2.3- Utilizar y probar el almacenamiento magnético

El Disco Rígido seleccionado resulta adecuado para el volumen de transacciones utilizadas, aunque puede pensarse en una ampliación en un tiempo de dos años. Sin embargo podría utilizarse una estrategia de resguardo de archivos históricos que permitan remover aquellos menos utilizados o teniendo en cuenta los que poseen escasos movimientos.

III.2.4- Utilizar y probar las interfaces

La pantalla seleccionada resulto adecuada en cuanto a la resolución, y fundamentalmente el tipo de tacto proyectivo, esa decir capacitivo multipunto.

III.3-Desarrollar, utilizar, integrar y probar Característica de Conectividad

III.3.1- Utilizar y probar el enlace físico

La tecnología inalámbrica seleccionada presentó algunos inconvenientes de inestabilidad de la señal en relación a las ondas de radiofrecuencia

III.3.2- Utilizar y probar la velocidad de conexión

La velocidad de conexión resultó adecuada aunque podría pensarse en cambiar la frecuencia de transmisión.

III.3.3- Utilizar y probar el protocolo de comunicación

El Protocolo seleccionado resultó adecuado a las velocidades y configuración de la red utilizada.

III.3.4- Utilizar y probar el tipo de red

El diseño del tipo de red LAN mixta utilizando estructura cableada y con puntos de acceso (AP) Ubiquiti Dual Band, resultó adecuada. En la figura 14 puede observarse el diseño de la red utilizada.

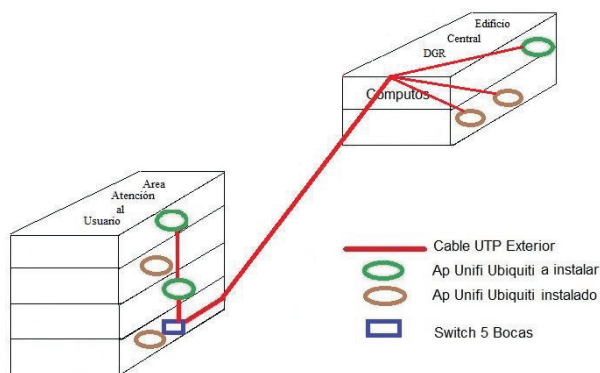


Figura 14. Diseño Red LAN

5. Conclusiones

El proceso de validación utilizado, permitió demostrar que la metodología metodo_SCGE resulta de fácil aplicación y sumamente intuitiva para su aplicación.

Teniendo en cuenta el caso de estudio elegido, se puede concluir que la metodología para las etapas I, II y III, se podrá utilizar sin ninguna dificultad en casos de complejidad alta, tanto en aspectos relacionados a la funcionalidad específica del dominio, como así también en los aspectos relacionados a la estimación de los recursos computacionales. Como trabajo futuro, se verificarán herramientas que permitan realizar una adecuada estimación de los recursos computacionales requeridos por la aplicación a desarrollar. Por lo expuesto precedentemente, se puede concluir que se cumplió con el objetivo propuesto en el presente trabajo.

El producto final que se obtuvo es un conjunto de especificaciones referidas a los servicios que se deben implementar, los procesos, el equipamiento y la conectividad necesaria para una unidad de organización gubernamental

Por otra parte se pudo visualizar que las actividades previstas en las etapas I, II y III, requieren de un conocimiento relativamente simple al momento de especificar los aspectos relacionados tanto al software como al hardware y la conectividad.

6. Referencias

1. Diseño, desarrollo e implementación de los modelos y estrategias de Gobierno Electrónico para una gestión abierta, cercana, eficiente y eficaz de la Administración Pública. Secretaría Nacional de la Administración Pública, Gobierno de Ecuador. Disponible en <http://www.administracionpublica.gob.ec/wp-content/uploads/downloads/2016/04/Gobierno-Electronico-23-12-2015.pdf>, 2015.
2. Booch, G., Rumbaugh, J., Jacobson, I.: El Lenguaje Unificado de Modelado. Addison Wesley, 2004.
3. Erl, T.: SOA Principles of Service Design. Prentice Hall, 2008.
4. Pacific Council of International Policy: Roadmap for e-government in the developing world. Disponible en <http://unpan1.un.org/intradoc/groups/public/documents/apc/ity/unpan005030.pdf>, 2002.
5. Affisco, J., Soliman, K.: E-government: a strategic operations management framework for service delivery. Business Process Management Journal. Vol.12 Issue:1. <https://doi.org/10.1108/14637150610643724>, 13-21, 2006.
6. Hsin-Pin, F., Tien-Hsiang, C., Pei, C., Chyou-Huey, C.: A collaborative model for service provision by multiple public-sector agencies. Internet Research, Vol. 16 Iss: 4. Disponible en <http://www.emeraldinsight.com/doi/abs/10.1108/10662240610690007>, 365 – 379, 2006.
7. Hreño, J., Bednár, P., Furdik, K., Sabol, T.: Integration of Government Services using Semantic Technologies. Journal of theoretical and applied electronic commerce research, Res. vol.6 no.1. Disponible en http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-18762011000100010, 2011.
8. Chhabra, S., Kumar, M.: Integrating E-Business Models for Government Solutions: Citizen-Centric Service Oriented Methodologies and Processes., Advisory Board, 2009.
9. Dzhusupova, Z., Shareef, M., Ojo, A., Janowski, T.: Methodology for e-government readiness assessment-models, instruments, implementation, - Proceedings of the International Conference on Society and Information Technologies, 6-9, 2010.
10. Mellor, S., Scott, K., Uhl, A., Weise, D.: MDA Distilled, Principles of Model Driven Architecture, Addison-Wesley, Professional, 2004.
11. Castro, M.: Análisis de las propiedades y atributos propios de sitios de gobierno electrónico. Tesis de Maestría en Ingeniería de Software, Universidad Nacional de San Luis, 2009.