



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA
COMPUTACIÓN**

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: DESARROLLO DE UNA APLICACIÓN WEB DE
FACTURACIÓN ELECTRÓNICA BAJO MODELO SAAS PARA
LA EMPRESA SOFYA SYSTEMS S.A.**

**AUTORES: GUERRA RUIZ ERNESTO ANDRÉS
MATUS CORONEL ABRAHAM ISRAEL**

DIRECTORA: ING. GÓMEZ SALAZAR MÓNICA ELIZABETH

SANGOLQUÍ

2016

CERTIFICADO



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIONES

CARRERA DE INGENIERIA EN SISTEMAS E INFORMÁTICA

CERTIFICACIÓN

Certifico que el trabajo de titulación, “DESARROLLO DE UNA APLICACIÓN WEB DE FACTURACIÓN ELECTRÓNICA BAJO MODELO SAAS PARA LA EMPRESA SOFYA SYSTEMS S.A.” realizado por los señores ABRAHAM ISRAEL MATUS CORONEL y ERNESTO ANDRÉS GUERRA RUIZ, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a los señores ABRAHAM ISRAEL MATUS CORONEL y ERNESTO ANDRÉS GUERRA RUIZ para que lo sustente públicamente.

Sangolquí, 11 de agosto del 2016

Atentamente,

Ing. Mónica Gómez S., MIS
Directora de Tesis

AUTORÍA DE RESPONSABILIDAD



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

AUTORÍA DE RESPONSABILIDAD

Nosotros, MATUS CORONEL ABRAHAM ISRAEL, con cédula de identidad N° 1717379257, y GUERRA RUIZ ERNESTO ANDRÉS, con cédula de identidad N° 1003341508, declaramos que este trabajo de titulación “DESARROLLO DE UNA APLICACIÓN WEB DE FACURACIÓN ELECTRÓNICA BAJO MODELO SAAS PARA LA EMPRESA SOFYA SYSTEMS S.A.” ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 10 de octubre de 2016

MATUS CORONEL ABRAHAM
ISRAEL
1717379257

GUERRA RUIZ ERNESTO
ANDRÉS
1003341508

AUTORIZACIÓN



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

AUTORIZACIÓN

Nosotros, MATUS CORONEL ABRAHAM ISRAEL y GUERRA RUIZ ERNESTO ANDRÉS, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación “DESARROLLO DE UNA APLICACIÓN WEB DE FACURACIÓN ELECTRÓNICA BAJO MODELO SAAS PARA LA EMPRESA SOFYA SYSTEMS S.A.” cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Sangolquí, 10 de octubre de 2016

MATUS CORONEL ABRAHAM
ISRAEL
1717379257

GUERRA RUIZ ERNESTO
ANDRÉS
1003341508

DEDICATORIA

Dedico este trabajo a mis padres y hermanos, quienes con sacrificio y esfuerzo me han apoyado a través de mis años de estudios de manera incondicional, siempre animándome para alcanzar mis sueños; a mis abuelos, quienes siempre confiaron en mí y me motivaron a ser una persona de bien para la sociedad, quienes estarían orgullosos de mí al ver en lo que me he convertido; a los profesores que marcaron mi vida, quienes dejaron de lado su rol e hicieron mucho más por mí para satisfacer mi interés y curiosidad, quienes notaron potencial en mí y me animaron para ser más; a todas aquellas personas, conocidas o anónimas, pertenecientes a la comunidad de software libre, quienes gracias a su sincero interés por compartir y difundir el conocimiento no sólo me ayudaron a mí, sino a millones de personas alrededor del mundo.

Abraham Matus Coronel

AGRADECIMIENTO

El resultado de este trabajo está dedicado a aquellas personas e instituciones que me ayudaron a culminar con esta etapa. Mis más sinceros agradecimientos a la empresa SOFYA S.A., a los Ingenieros Bruno Luzuriaga y Juan Cabrera, por su apoyo y apertura para desarrollar este proyecto bajo su tutela, por confiar en mis capacidades y profesionalismo. A mis compañeros de trabajo, que no dudaron en darme su punto de vista o una mano cuando lo necesitaba. Agradezco la Ing. Mónica Gómez por su tutoría y guía en el desarrollo de esta tesis, por su conocimiento y paciencia, por su amabilidad y apoyo aún en los momentos más difíciles. Muchísimas gracias a mi familia, que a pesar del largo camino me han apoyado siempre en todos los aspectos, y a mi novia, quien con su compasión y optimismo me ayudó a ver el lado positivo ante cada adversidad, los amo con todo mi corazón.

Abraham Matus Coronel

Quisiera agradecer a todas las personas que me han apoyado a lo largo de este proyecto, especialmente a Abraham ya que con él pudimos completar todo esto. Agradezco de manera muy grande a mis padres por su cariño y comprensión que me han mostrado siempre, principalmente en estos días

Ernesto Guerra

ÍNDICE

CERTIFICADO	II
AUTORÍA DE RESPONSABILIDAD.....	III
AUTORIZACIÓN	IV
DEDICATORIA.....	V
AGRADECIMIENTO	VI
ÍNDICE	VII
ÍNDICE DE TABLAS	X
ÍNDICE DE FIGURAS	XII
RESUMEN	XIV
ABSTRACT.....	XV
CAPÍTULO I – INTRODUCCIÓN.....	1
1.1 ANTECEDENTES	1
1.2 PLANTEAMIENTO DEL PROBLEMA.....	2
1.3 JUSTIFICACIÓN	3
1.4 OBJETIVOS	4
1.4.1 OBJETIVO GENERAL	4
1.4.2 OBJETIVOS ESPECÍFICOS.....	4
1.5 ALCANCE.....	4
CAPÍTULO II – MARCO TEÓRICO.....	6
2.1 METODOLOGÍA UWE	6
2.1.1 CICLO DE VIDA.....	6
2.1.2 FASES	9
2.1.3 ROLES.....	9
2.1.4 PRODUCTOS ENTREGABLES	10
2.2 COMPROBANTES ELECTRÓNICOS	19
2.2.1 BASE LEGAL SRI	20
2.2.2 TIPOS DE COMPROBANTES	21

	viii
2.2.3 INFORMACIÓN TÉCNICA	21
2.3 SOFTWARE COMO SERVICIO	25
2.3.1 DEFINICIÓN.....	25
2.3.2 CARACTERÍSTICAS	25
2.3.3 ALTERNATIVAS	25
CAPÍTULO III – INICIACIÓN Y ELABORACIÓN.....	28
3.1 INICIACIÓN	28
3.1.1. MODELO DE CASOS DE USO.....	28
3.1.2. DIAGRAMA DE ACTIVIDAD – AUTENTICAR USUARIO.....	29
3.1.3. DIAGRAMA DE ACTIVIDAD – SUBIR COMPROBANTE	29
3.1.4. DIAGRAMA DE ACTIVIDAD – GENERAR XML	30
3.1.5. DIAGRAMA DE ACTIVIDAD – EMITIR COMPROBANTE SRI.....	31
3.1.6. DIAGRAMA DE ACTIVIDAD – OBTENER AUTORIZACIÓN SRI	32
3.1.7. DIAGRAMA DE ACTIVIDAD – ENVIAR CORREO.....	33
3.1.8. DIAGRAMA DE ACTIVIDAD – BUSCAR COMPROBANTES.....	33
3.1.9. DIAGRAMA DE ACTIVIDAD – DESCARGAR COMPROBANTES	34
3.1.10. DIAGRAMA DE ACTIVIDAD – PERSONALIZAR CORREOS.....	34
3.1.11. DIAGRAMA DE ACTIVIDAD – PERSONALIZAR RIDE.....	35
3.1.12. DIAGRAMA DE ACTIVIDAD – SALIR USUARIO.....	35
3.2 ELABORACION	36
4.2.1. MODELO DE CONTENIDO	36
4.2.2. MODELO DE NAVEGACIÓN	37
4.2.3. MODELO DE PRESENTACIÓN	37
4.2.4. MODELO DE PROCESO	40
3.3 ARQUITECTURA DE SOFTWARE	48
CAPÍTULO IV – CONSTRUCCIÓN Y TRANSICIÓN.....	49
4.1 CONSTRUCCIÓN	49
4.1.1 DIAGRAMA FÍSICO DE BASE DE DATOS.....	51
4.1.2 SCRIPTS, PAQUETES Y MÉTODOS.....	52
4.2 TRANSICIÓN	55
4.2.1. CASOS DE PRUEBA	55
4.2.2. MANUAL DE USUARIO	63

	ix
4.2.3. CARTA DE ACEPTACIÓN.....	64
CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES.....	65
5.1 CONCLUSIONES	65
5.1 RECOMENDACIONES.....	66
BIBLIOGRAFÍA	67

ÍNDICE DE TABLAS

Tabla 1. Tipos de nodos en el diagrama de actividades.....	12
Tabla 2. Estereotipos en un modelo de navegación.....	16
Tabla 3. Estereotipos en un modelo de presentación.....	17
Tabla 4. WSDL de los servicios del ambiente de pruebas.....	23
Tabla 5. WSDL de los servicios del ambiente de producción.....	23
Tabla 6. Herramientas utilizadas para el desarrollo de SEBS.....	49
Tabla 7. Librerías necesarias para el desarrollo de SEBS.....	50
Tabla 8. Clases de paquete model.....	53
Tabla 9. Paquete de los DAO.....	53
Tabla 10. Paquete de emisor de comprobante.....	53
Tabla 11. Paquete con las clases de autorización de comprobantes.....	54
Tabla 12. Paquete con las clases clientes de los servicios web.....	54
Tabla 13. Paquete con las clases de firma electrónica.....	54
Tabla 14. Paquete con los generadores de PDF.....	54
Tabla 15. Paquete con las clases de envío de email.....	55
Tabla 16. Paquete de generación de XML.....	55
Tabla 17. Clases de equivalencia de Autenticar Usuario.....	56
Tabla 18. Casos de prueba de Autenticar Usuario.....	56
Tabla 19. Clases de equivalencia de Subir Comprobante.....	57
Tabla 20. Casos de prueba de Subir Comprobante.....	57
Tabla 21. Clases de equivalencia de Emitir comprobante no procesado.....	58
Tabla 22. Caso de uso de emisión de comprobante no procesado.....	58
Tabla 23. Clases de equivalencia de autorización de comprobante pendiente.....	58
Tabla 24. Caso de uso de autorización de comprobante pendiente.....	59
Tabla 25. Clases de equivalencia de búsqueda de comprobantes.....	59
Tabla 26. Casos de uso de búsqueda de comprobantes.....	59
Tabla 27. Clases de equivalencia de descarga de formato RIDE.....	60
Tabla 28. Casos de prueba de descarga de formato RIDE.....	60
Tabla 29. Clases de equivalencia de descarga de formato XML.....	60
Tabla 30. Casos de prueba de descarga de formato XML.....	61
Tabla 31. Clases de equivalencia de descarga de archivo ZIP.....	61

	xi
Tabla 32. Casos de uso de descarga de archivo ZIP.	61
Tabla 33. Clases de equivalencia de personalización de correos electrónicos.	62
Tabla 34. Casos de uso de personalización de correos electrónicos.	62
Tabla 35. Clases de equivalencia de personalización de RIDE.	63
Tabla 36. Casos de uso de personalización de RIDE.	63

ÍNDICE DE FIGURAS

Figura 1. Modelo lineal de Actividades.....	7
Figura 2. Modelo Incremental.....	7
Figura 3. Modelo Iterativo.....	8
Figura 4. Actor.....	11
Figura 5. Caso de Uso.....	11
Figura 6. Relaciones del Diagrama de Casos de Uso.....	11
Figura 8. Flujo.....	13
Figura 9. Representación de una Clase.....	13
Figura 10. Relaciones en un Diagrama de Clases.....	14
Figura 11. Representación de la Agregación y la Composición.....	15
Figura 12. Ejemplo de un Modelo de Presentación.....	18
Figura 13. Ejemplo de un Diagrama de Estructura de Procesos.....	18
Figura 14. Ejemplo de Diagrama de Flujo de Procesos.....	19
Figura 15. Ejemplo de factura en formato RIDE.....	24
Figura 16. Diagrama de casos de uso de la aplicación.....	28
Figura 17. Diagrama de actividad del caso de uso Autenticar Usuario.....	29
Figura 18. Diagrama de actividad del caso de uso Subir Comprobante.....	29
Figura 19. Diagrama de actividad del caso de uso Generar XML.....	30
Figura 20. Diagrama de actividad del caso de uso Emitir Comprobante.....	31
Figura 21. Diagrama de actividad del caso de uso Obtener Autorización SRI.....	32
Figura 22. Diagrama de actividad del caso de uso Enviar Correo.....	33
Figura 23. Diagrama de actividad del caso de uso Buscar Comprobantes.....	33
Figura 24. Diagrama de actividad del caso de uso Descargar Comprobantes.....	34
Figura 25. Diagrama de actividad del caso de uso Personalizar Correos.....	34
Figura 26. Diagrama de actividad del caso de uso Personalizar RIDE.....	35
Figura 27. Diagrama de actividad del caso de uso Salir Usuario.....	35
Figura 28. Diagrama de clases que define al modelo de contenido.....	36
Figura 29. Diagrama de navegación de la aplicación.....	37
Figura 30. Modelo de presentación de la pantalla de autenticación.....	37
Figura 31. Modelo de presentación del menú.....	38
Figura 32. Modelo de presentación de la pantalla principal.....	38

Figura 33. Modelo de presentación de la pantalla de subida de comprobantes.	39
Figura 34. Modelo de presentación de la pantalla de comprobantes emitidos.	39
Figura 35. Modelo de presentación de la pantalla de configuración de RIDE.	40
Figura 36. Modelo de presentación de la pantalla de configuración de correos electrónicos.	40
Figura 37. Diagrama de clases que representa el modelo de Estructura de Procesos.	41
Figura 38. Diagrama de actividad que define el flujo de procesos para Login.	41
Figura 39. Diagrama de actividad que define el flujo de procesos para Logout.	42
Figura 40. Diagrama de actividad que define el flujo de procesos para Subir Comprobantes.	43
Figura 41. Diagrama de actividad que define el flujo de procesos para Comprobantes Emitidos.	44
Figura 42. Diagrama de actividad que define el flujo de procesos para Descargar Comprobantes.	45
Figura 43. Diagrama de actividad que define el flujo de procesos para Configurar Correo.	46
Figura 44. Diagrama de actividad que define el flujo de procesos para Configurar RIDE.	47
Figura 45. Arquitectura en múltiples capas de la aplicación.	48
Figura 46. Diagrama físico de la base de datos de SEBS.	51

RESUMEN

La facturación electrónica es un nuevo paradigma de tributación que aprovecha las ventajas de las tecnologías en la información. En Ecuador, el Servicio de Rentas Internas (SRI) impuso el uso obligatorio de la facturación electrónica desde 2015 para ciertos sectores comerciales como, por ejemplo, empresas que realizan exportaciones. SOFYA S.A. posee clientes que se desenvuelven en estas áreas, por lo que se ve en la necesidad de brindar una solución de facturación electrónica sencilla fácil de usar a los mismos, así como empresarialmente se ve la oportunidad de aprovechar el nicho de mercado de soluciones informáticas de esta naturaleza para ofrecer a terceros. Para esto se desarrollo una aplicación de facturación electrónica que permita la generación, emisión, autorización y descarga de comprobantes electrónicos, distribuida bajo modelo de software como servicio o SaaS que aprovecha las bondades de la nube, utilizando la metodología de desarrollo de aplicaciones web UWE, apoyada de entregables adicionales que validen las etapas del ciclo de vida de software y de una arquitectura de software adecuada que permita su despliegue en la nube, obteniendo como resultado un proceso robusto y bien documentado del desarrollo de una aplicación web con una metodología formal y ágil, a la vez que se entrega al cliente final un producto que satisface sus necesidades.

Palabras clave:

- **FACTURACIÓN ELECTRÓNICA**
- **UWE**
- **SAAS**
- **APLICACIÓN WEB**
- **UML**

ABSTRACT

Electronic invoicing is a new paradigm of taxation that takes advantage of information technologies. In Ecuador, the Internal Revenue Service (SRI) imposed the mandatory use of electronic invoicing since 2015 for certain business sectors, for example, companies that export products or services. SOFYA S.A. has clients who operate in these areas, so is the need to provide them an easy solution for electronic invoicing, as well as entrepreneurially opportunity to tap the niche market of solutions of this nature to offer to third parties. For this, it was developed an electronic invoicing application that allows the creation, issuance, authorization and download electronic vouchers, distributed under software as a service or SaaS that leverages the benefits of the cloud, using the methodology of web application development UWE, supported on additional deliverables that validate the stages of the life cycle of software and architecture appropriate software that allows its deployment in the cloud, resulting in a robust and well documented development of a web application with a formal methodology and agile process , while the final customer is delivered a product that meets their needs.

Keywords:

- **ELECTRONIC INVOICING**
- **UWE**
- **UML**
- **WEB APPLICATION**
- **SAAS**

CAPÍTULO I – INTRODUCCIÓN

1.1 ANTECEDENTES

La facturación electrónica es la evolución y nueva normativa del Servicio de Rentas Internas (SRI), expedida bajo el Art. 2 de la Resolución No. NAC-DGERCGC12-00105 en 2012, del proceso tradicional de emisión de facturas y documentos tributarios que reemplaza medios físicos, como: especies y copias en papel por documentos en formato digital que hacen uso de herramientas y estándares informáticos para garantizar la autenticidad y proporcionar validez legal y tributaria.

El término “Facturación Electrónica” no se limita a este tipo de documentos tributarios, sino que abarca también otros tipos de comprobantes electrónicos como lo son: notas de débito, notas de crédito, guías de remisión y comprobantes de retención; de acuerdo a lo estipulado en el Art. 2. La emisión de comprobantes electrónicos podrá ser realizada por personales naturales o jurídicas conforme a normativas entre las que se puede destacar:

- Ley de Comercio Electrónico, Firmas y Mensajes de Datos publicado en el Suplemento del Registro Oficial No. 557 de 17 de abril de 2002.
- Reglamento a la Ley de Comercio Electrónico, Firmas y Mensajes de Datos, publicado en el Registro Oficial No. 735 de 31 de diciembre de 2002.
- Decreto No. 181 publicado en el Registro Oficial No. 553 de 11 de octubre de 2011.
- Resolución No. NAC-DGERCGC12-00105 de 09 de marzo de 2012, publicada en Registro Oficial No. 666 de 21 de marzo de 2012.
- Resolución NAC-DGERCGC14-00790, publicada en el Registro Oficial 346 de 2 de octubre de 2014.
- Resolución NAC-DGERCGC13-00236, Registro Oficial 956 de 17 de mayo de 2013.

El SRI dispuso de un programa piloto de facturación electrónica en 2012, permitiendo que entidades participen del mismo de manera voluntaria. La emisión de comprobantes electrónicos se establece como obligatoria desde 2014, cumpliendo

con el siguiente cronograma según el Registro Oficial 956 No. NAC-DGERCGC13-00236:

- Desde el 01 de junio de 2014 las instituciones financieras exceptuando mutualistas de ahorro, y contribuyentes especiales correspondientes al sector de las telecomunicaciones y televisión pagada.
- Desde el 01 de agosto de 2014 sociedades emisoras y administradoras de tarjetas de crédito.
- Desde el 01 de octubre de 2014 exportadores calificados como contribuyentes especiales.
- Desde el 01 de enero de 2015 todos los demás contribuyentes que no correspondan a los grupos señalados con anterioridad, incluyendo aquellos que realicen actividades de exportación.

Por lo tanto, desde enero de 2015, todas las empresas, ya sean grandes multinacionales o Pequeñas y Medianas Empresas (PYMES), están obligadas a realizar emisión de comprobantes electrónicos.

1.2 PLANTEAMIENTO DEL PROBLEMA

SOFYA SYSTEMS S.A. es una empresa de desarrollo de software enfocado en sistemas web y con funcionamiento de los mismos para una variedad de clientes. Debido a la disposición del Servicio de Rentas Internas, SOFYA SYSTEMS S.A. se ve en la necesidad de adaptar a sus sistemas en producción, y en desarrollo, la capacidad de emitir comprobantes electrónicos para proporcionar a sus clientes una herramienta que ayude a solventar esta situación apremiante, y a su vez comercializar esta solución a terceros.

Hasta finales de enero de 2015 más de 5000 empresas en todo el país deberán adoptar obligatoriamente la emisión de comprobantes electrónicos. SOFYA SYSTEMS S.A. cuenta con clientes que no disponen de una herramienta para la emisión de comprobantes electrónicos, por lo que se ve en la oportunidad y necesidad de desarrollar una solución que cumpla con esta disposición obligatoria y que satisfaga a quienes estén interesados, incluyendo a terceros, sobre todo en aspectos como las facturas correspondientes a las actividades comerciales que

involucran exportaciones, ya que es una característica que la herramienta gratuita del SRI no contempla ni brinda solución.

1.3 JUSTIFICACIÓN

Debido a la nueva normativa del SRI, empresas de diversas actividades comerciales están sujetas a un cronograma de obligatoriedad descrito en la Resolución NAC-DGERCGC13-00236 definido entre 2014 y 2015. Las empresas que realizan actividades económicas que involucran la exportación están sujetas a esta obligatoriedad. La herramienta gratuita de emisión de comprobantes electrónicos que dispone el SRI no contempla las necesidades de empresas de esta naturaleza, por lo que es necesario desarrollar un sistema que se adapte a estas necesidades. Asimismo, se requiere que conocimientos técnicos informáticos para poder configurar y usar esta herramienta gratuita correctamente, lo que se evita en un modelo de software como servicio como el sistema dispuesto a desarrollar, además de reducir el riesgo de error humano.

La empresa de desarrollo de software SOFYA SYSTEMS S.A., con su solución web de facturación electrónica bajo modelo software como servicio suplirá la necesidad de clientes actuales de la empresa, así como atraer clientes potenciales, brindándoles los siguientes beneficios:

- Disminución de los costos operacionales y tributarios.
- Eliminación de almacenamiento de comprobantes en formato físico.
- Robustez y seguridad de documentos electrónicos a través de firmas digitales.
- Facilidad de consulta y acceso a comprobantes electrónicos emitidos.
- Cálculo de impuestos, valores y retenciones.
- Acceso a la plataforma desde cualquier parte del país.
- Alta disponibilidad de servicio.
- Generación de Representación Impresa de Documentos Electrónicos (RIDE) para los comprobantes electrónicos.
- Validación automatizada contra los servicios web del Servicio de Rentas Internas.
- Comunicación de datos cifrada y segura.

- Integración a sistemas de terceros de manera no intrusiva a partir de Interfaces de Programación de Aplicaciones (Aplicación Programming Interface - API).

1.4 OBJETIVOS

1.4.1 Objetivo general

Desarrollar una aplicación web de facturación electrónica bajo modelo Software as a Service (SaaS), respetando la normativa legal pertinente emitida por el Estado y organismos de control, para la empresa SOFYA SYSTEMS S.A., integrándolo con su sistema de gestión de usuarios.

1.4.2 Objetivos específicos

1. Investigar la normativa legal y técnica para la emisión de comprobantes electrónicos.
2. Analizar la situación actual de la Empresa en relación a la facturación electrónica y especificar requisitos de software.
3. Diseñar y desarrollar la aplicación web de facturación electrónica.
4. Realizar un plan pruebas para verificar y validar el producto software.
5. Validar el marco de trabajo de Software como servicio SaaS para la implementación de la aplicación web.

1.5 ALCANCE

El alcance de la aplicación a desarrollar para la empresa SOFYA SYSTEMS S.A. está definido por los siguientes módulos y funcionalidades:

- Uso del core de gestión de usuarios de SOFYA SYSTEMS S.A. para controlar las operaciones de los mismos.
- Carga de facturas locales, facturas de exportación, notas de crédito, notas de débito, guías de remisión y comprobantes de retención en formato Excel.

- Carga de firmas electrónicas en formato archivo PKCS #12 por parte de los contribuyentes.
- Generación de impuestos y detalles a partir del comprobante cargado en el sistema.
- Generación y validación del comprobante electrónico en formato XML y firmado del mismo.
- Emisión y obtención del estado de autorización del comprobante electrónico contra los servicios web del SRI.
- Emisión de comprobantes electrónicos para ambientes de prueba y producción.
- Visualización de facturas, notas de crédito, notas de débito, guías de remisión y comprobantes de retención emitidos.
- Generación y descarga del formato de RIDE de los comprobantes electrónicos.
- Almacenamiento de comprobantes electrónicos en la nube.
- Envío de correo electrónico con el comprobante en formato XML y PDF/RIDE al cliente.
- Ser accesible vía Internet a través del Protocolo Seguro de Transferencia de Hipertexto HTTPS.

El sistema no contemplará:

- Gestión de inventarios.
- Generación de informes.
- Generación de Anexos Transaccionales Simplificados (ATS)

CAPÍTULO II – MARCO TEÓRICO

2.1 METODOLOGÍA UWE

UWE, por sus siglas en inglés de UML-based Web Engineering, es una metodología de desarrollo de software para aplicaciones Web que utiliza gran cantidad de los diagramas UML, aunque podría hacer uso de todos ellos.

Las principales características de esta metodología son el uso de una notación estándar y la definición de los modelos de manera muy detallada. La ventaja de tener una sola notación, en este caso UML, es que se estandarizan los modelos a seguir y se los puede leer de una manera fácil.

2.1.1 CICLO DE VIDA

El ciclo de vida de un software se define como “una aproximación lógica a la adquisición, el suministro, el desarrollo, la explotación y el mantenimiento del software” (The Institute of Electrical and Electronics Engineers, 1998).

El ciclo de vida del software viene dado desde que se tiene la idea de crear algo hasta el momento en el que el software deja de funcionar. También es un conjunto de Actividades que se dan en orden secuencial y que están definidas en modelos de ciclos de vida. Estas Actividades son “cuerpos definidos de trabajo a ser realizado, incluyendo su información de Entrada y de Salida” (The Institute of Electrical and Electronics Engineers, 1998).

- **Modelo Incremental**

Este modelo permite la ejecución de procesos lineales en varias iteraciones, es incremental debido a que en cada iteración se obtienen incrementos del software. Este modelo también está basado en la construcción de prototipos. Cada proceso lineal generalmente está compuesto por varias actividades en forma secuencial que tienen la siguiente estructura, ilustrada en la Figura 1:



Figura 1. Modelo lineal de Actividades.

Al final de cada iteración se obtiene un producto software operacional, como se muestra en la Figura 2. “Los primeros incrementos son versiones incompletas del producto final, pero proporcionan al usuario la funcionalidad que necesita y una plataforma para evaluarlos” (Pressman, 2005).

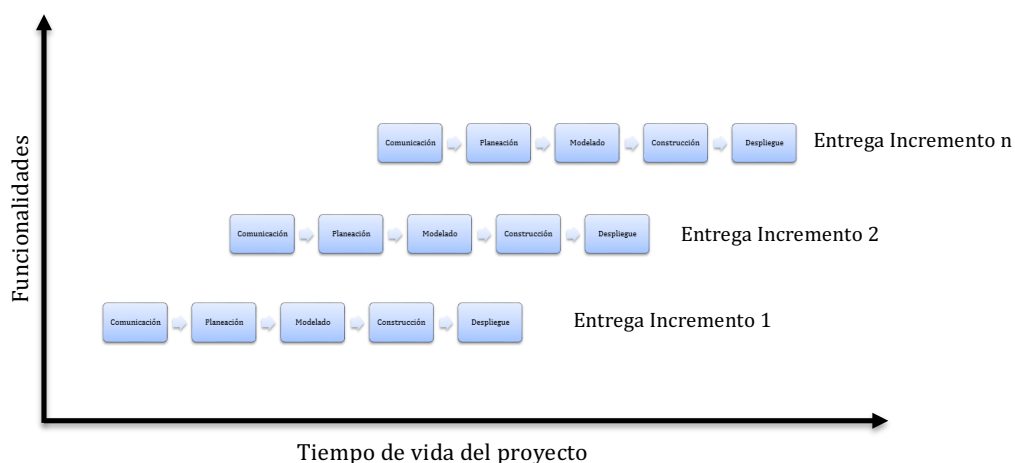


Figura 2. Modelo Incremental.

Cada iteración puede iniciar antes de que se tenga el resultado de cada incremento, de este modo se puede trabajar de forma paralela en varias Actividades de diferentes iteraciones; esto sirve cuando no se tiene mucho personal o el personal puede hacer varias actividades.

- **Modelo Iterativo**

Es un modelo en el cual se realizan varias iteraciones de un modelo lineal de Actividades pero que para que comience una iteración debe terminar completamente la iteración anterior. Al final de cada iteración se obtiene un producto final que es una mejora del anterior producto software, como se muestra en la Figura 3.

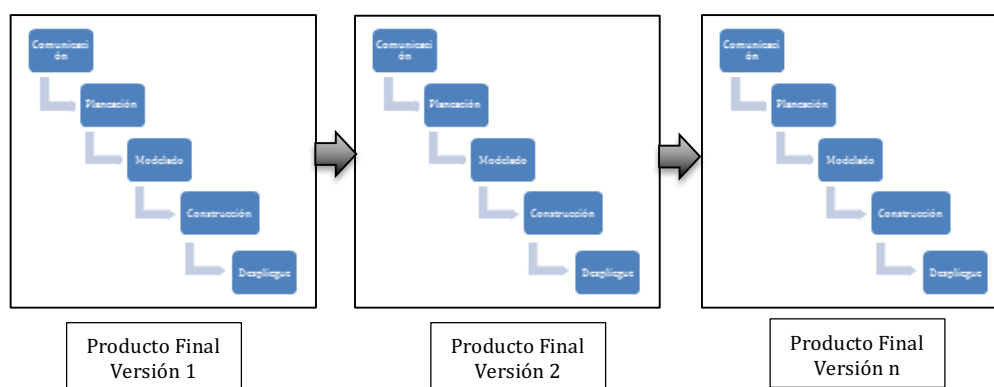


Figura 3. Modelo Iterativo.

En este modelo el cliente puede evaluar el producto luego de cada iteración y propone mejoras en el mismo con las cuales se comienza la siguiente iteración.

UWE no es una metodología creada desde cero, sino una aproximación a utilizar elementos existentes en otras metodologías de desarrollo de aplicaciones multimedia, así como el Proceso Unificado de Desarrollo (RUP), teniendo gran parecido por su agilidad al Proceso Unificado Ágil o Agile Unified Process (AUP) por su nombre en inglés, por lo que su ciclo de vida se basa en este siendo iterativo-incremental.

2.1.2 FASES

UWE utiliza un modelo iterativo-incremental que presenta fases parecidas a las presentadas en el Proceso Unificado de Modelado, pero se hace más énfasis a la parte de análisis y diseño. Estas fases son:

1. **Iniciación:** fase en la que se adquieren y especifican las características funcionales y no funcionales de la aplicación web. Se utilizan los diagramas de casos de uso y de actividad correspondiente al modelo de requerimientos.
2. **Elaboración:** se define la estructura que deberá tener la aplicación para cumplir con las especificaciones de requisitos adquiridas en la fase anterior. Se realizan el modelo de contenido representado por el diagrama de clases, modelo de navegación, modelo de presentación y modelo de proceso.
3. **Construcción:** se lleva a cabo la construcción del producto por medio de iteraciones.
4. **Transición:** validar y entregar el sistema en un ambiente de producción.

2.1.3 ROLES

En UWE se define como roles a los expertos responsables de las actividades efectuadas en cada una de las fases de cada, se los conoce como trabajadores o workers en inglés. La metodología no define explícitamente cuáles son estos roles o trabajadores involucrados, sin embargo, se hace hincapié en los siguientes roles:

- **Analista de sistema:** Es el encargado de diseñar el sistema en una forma global en base a los requerimientos obtenidos.
- **Especificador de requisitos:** Es el encargado de elicitar la información del cliente referente al sistema y plasmarla en papel.
- **Diseñador de interfaz de usuario:** Es el encargado de diseñar la parte visual de la aplicación.
- **Diseñador de base de datos:** Es el encargado de la creación de los diferentes modelos de la base de datos en base a los requerimientos obtenidos.

- **Implementador:** Es el encargado de desarrollar los componentes del sistema y realizar las pruebas de integración del mismo.
- **Jefe de proyecto:** Es el encargado de la comunicación entre el equipo de desarrollo y el cliente, además se encarga de la coordinación de los materiales necesarios para el equipo y el control de las etapas del desarrollo.
- **Especialista en Pruebas (tester):** Es el encargado de diseñar, ejecutar e informar sobre las pruebas realizadas al sistema.

2.1.4 PRODUCTOS ENTREGABLES

La metodología determina puntualmente los artefactos o productos entregables que se deben generar, agrupando las actividades en modelos los cuales están conformados por uno o más diagramas:

1. **Modelo de Requerimientos:** representado por diagramas de casos de uso y diagramas de actividad.
2. **Modelo de contenido:** representado por el diagrama de clases.
3. **Modelo de navegación:** representado por el diagrama de navegación.
4. **Modelo de presentación:** representado por el diagrama de presentación.
5. **Modelo de proceso:** representado por el modelo de estructura de proceso (diagrama de clases) y el modelo de flujo de proceso (diagrama de actividades).

Modelo de Requerimientos – Diagrama de Casos de Uso

Este modelo permite representar gráficamente los casos de uso que se obtuvieron de la elicitación de requerimientos, este modelo pertenece a UML y tiene los siguientes componentes gráficos:

- a. **Actor:** Alguien o algo externo al sistema que interactúa con él desempeñando un rol.



Figura 4. Actor.

- b. **Caso de uso:** es la representación de la acción a ejecutarse, siempre debe ser llamada por alguien o algo. Los casos de uso pueden ejecutar otros casos de uso.



Figura 5. Caso de Uso.

- c. **Relaciones/Asociaciones:** muestran la relación entre los otros elementos del modelo, pueden ser inclusión, extensión o generalización.

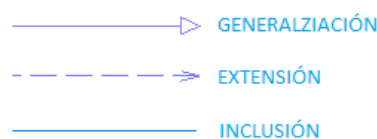


Figura 6. Relaciones del Diagrama de Casos de Uso.

- La *inclusión* se la usa cuando un actor llama directamente a un caso de uso, o cuando un caso de uso llama a otro caso de uso.
- La *extensión* consiste en una llamada a otro caso de uso, pero que tiene características similares a dicho caso de uso.
- La *generalización* indica que un caso de uso es muy similar a otro, solo difiere en ciertas características; este concepto se lo puede asociar con la

herencia en el paradigma de programación Orientado a Objetos. Existen dos casos de usos especificados en esta relación, el caso generalizado que es el que tiene las funcionalidades comunes y el especializado que tiene una diferenciación del general.







- d. **Límite:** Agrupa casos de uso dentro de un mismo sistema, Útil cuando tenemos varios sistemas/subsistemas

Modelo de Requerimientos – Diagrama de Actividad

Este modelo muestra los procesos o acciones que realizará el sistema, o casos de uso, por medio de un flujo de trabajo; pertenece a UML y tiene los siguientes componentes gráficos:

- a. **Nodo:** es un símbolo que permite identificar si un proceso inicia o termina, se presentan varias formas de nodos en la Tabla 1:

Tabla 1.
Tipos de nodos en el diagrama de actividades.

Símbolo	Nombre	Descripción
	Nodo Inicial	Muestra el inicio del proceso o flujo de acciones.
	Nodo Final	Muestra la terminación del proceso.
	Nodo Decisión	Muestra un punto del flujo en el que se debe tomar una decisión y seguir con una actividad.
	Nodo Unión	Muestra un punto al cual llegan las actividades después de varias decisiones y el proceso continua cuando cualquiera de los flujos llega a este nodo.
	Fork	Muestra el comienzo de varias actividades que se realizan en paralelo.
	Join	Indica que todas las actividades unidas a esta deben haber terminado para poder seguir con el proceso.

- b. **Acción:** Es una tarea o actividad que deben realizar los usuarios o el sistema.

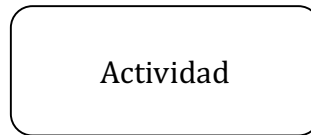


Figura 7. Acción.

- c. **Flujo:** Muestra el orden de ejecución de las actividades.



Figura 8. Flujo.

Modelo de contenido

“El objetivo del modelo de contenido es proporcionar una especificación visual de la información en el dominio relevante para la aplicación Web.” (Nieves Guerrero, Ucán Pech, & Menéndez Domínguez, 2014)

El modelo de contenido es un diagrama de clases UML estructurado para la aplicación Web. Los diagramas de clases son herramientas que permiten visualizar las relaciones entre dichas clases en la aplicación. Existen solo dos tipos de elementos en este diagrama:

- a. **Clase:** es la abstracción de un elemento del mundo, contiene atributos y métodos.

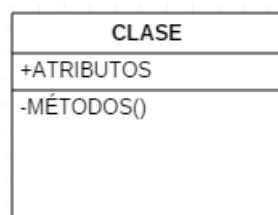


Figura 9. Representación de una Clase.

Los *atributos* son las características de la clase, en la POO un atributo esta denotado por un tipo de dato y por la visibilidad entre otras clases de dicha característica. Por otro lado, las visibilidades de estos atributos son comunes en los lenguajes orientados a objetos y estos son:

- PUBLIC (+): indica que los atributos serán vistos por cualquier clase que se relacione con el objeto instanciado.
- PROTECTED (#): indica que los atributos serán vistos por las clases heredadas de dicha clase.
- PRIVATE (-): indica que solo la clase que tiene los atributos puede verlos.

Los métodos son las funciones o acciones que ejecutan dichas clases, estos tienen datos de entrada y un dato de salida. Estos datos pueden ser de diferentes tipos, pero se los debe especificar en el diagrama. También tienen visibilidad, que fue especificada en los atributos.

- b. **Relación:** es la forma en que interactúan las clases, se tiene herencia, asociación, composición, agregación y uso.

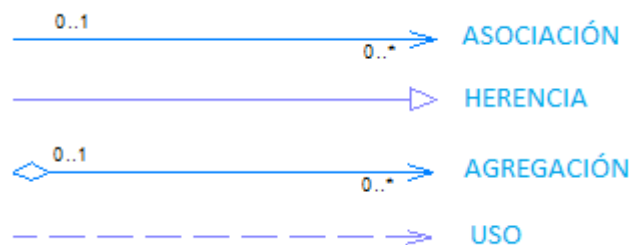




Figura 10. Relaciones en un Diagrama de Clases.

- **Herencia:** indica que una clase hereda los atributos y métodos de otra clase que se la denomina padre. La clase hija, en este caso, puede tener más atributos y métodos que la diferencien de la clase padre.
- **Asociación:** permite asociar clases que colaboran entre si y que no están estrechamente ligadas, esto quiere decir que no hay una dependencia muy fuerte entre estas clases. En la asociación se tiene una especificación para ver cuantos objetos de las clases dependen de los demás como por

ejemplo: la multiplicidad (1 a 1) que indica que un solo objeto de la primera clase está relacionado con un solo objeto de la segunda clase. Tenemos las siguientes multiplicidades:

- 0 ... * indica que pueden o no existir varios objetos
 - 1 indica que solo existe un objeto
 - 1 ... * indica que existen uno o más objetos
- **Agregación:** este tipo de relación indica que una clase es parte de otra clase; esto se lo puede interpretar como si una clase instanciara varios objetos de otra clase. En la agregación se tienen dos formas de uso de esta relación: por valor y por referencia.

Una *agregación por valor* indica que una clase va a estar atada a la clase que la agregó, comúnmente se lo llama composición y se lo representa por .

Una *agregación por referencia* indica que la clase agregada no tiene un vínculo tan estrecho con la clase que lo agregó, comúnmente llamada agregación y representada por .

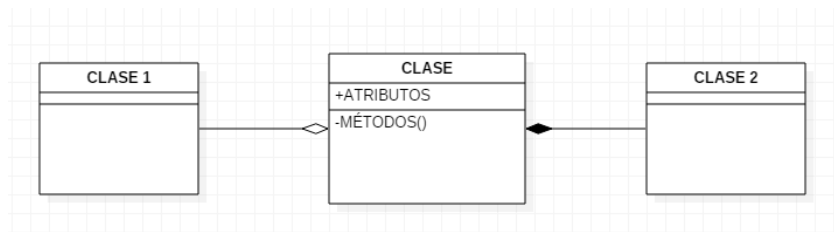


Figura 11. Representación de la Agregación y la Composición.

Por ejemplo, una clase A tiene agregada una clase B por valor y a una C por referencia; cuando se instancia A se crean objetos B pero los objetos C no se los crea, ya deben estar creados en otro lado del programa. De la misma manera cuando se destruye A, se destruye también B pero C no se destruye.

- **Uso:** es una relación de dependencia que indica que una clase solo puede ser instanciada si la clase que está en relación de uso esta instanciada. Un ejemplo es cuando se tiene un programa de ventanas, se debe instanciar la

clase que ejecuta la aplicación para luego instanciar la ventana de la otra clase.

Modelo de navegación – Diagrama de Navegación








Este modelo permite ver la forma en que las páginas de la aplicación web interactúan entre ellas. Este modelo está conformado por nodos y enlaces.

Un *nodo* es una unidad de navegación, esto quiere decir que puede ser una página web externa o un elemento de una página.

Los *enlaces* son las formas en cómo se van a llamar a los nodos.

En este diagrama se tienen estereotipos de los nodos, estos estereotipos indican la funcionalidad de cada nodo en el diagrama, en la Tabla 2 se especifican estos estereotipos. (Kroiß & Koch, 2008)

Tabla 2.
Estereotipos en un modelo de navegación.










Símbolo	Estereotipo	Descripción
	Clase de Navegación	Indican los nodos que pueden ser navegados.
	Menú	Indica las rutas alternativas de navegación.
	Índice	Permiten seleccionar una instancia de un nodo de una serie de instancias del mismo.
	Pregunta	Permite obtener contenido de una base de datos.
	Visita Guiada	Permite obtener contenido de una base de datos.
	Clase de proceso	Indica la ejecución de un proceso.
	Nodo Externo	Indica la ejecución de una acción en un sistema diferente.

Modelo de presentación

Este modelo permite tener una vista abstracta de las interfaces de usuario de la aplicación web, es basado en el diagrama de navegación. Describe que elementos

(texto, imágenes, video, etc.) van en que nodo del diagrama de navegación y cómo estos nodos son agrupados en las páginas web. Con esto se pueden ver los posibles diseños de las páginas y hacer cambios a los mismos antes de programar. Se utilizan estereotipos para mostrar los diferentes elementos del modelo: (Ludwig-Maximilians-Universität, 2006)

Tabla 3.
Estereotipos en un modelo de presentación.

Símbolo	Estereotipo	Descripción
	Grupo de presentación	Define a un grupo de clases cuyos contenidos se muestran en un mismo lugar de la página
	Página de presentación	Indica que la clase representa una página en la que se va a ubicar el resto de clases.
	Texto	Permite la visualización de texto plano.
	Entrada de Texto	Permite al usuario el ingreso de texto.
	Ancla	Permite la transición a un enlace especificado.
	File Upload	Permite al usuario la subida de un archivo al servidor.
	Botón	Permite al usuario iniciar una acción en la página Web
	Imagen	Permite la visualización de una imagen estática.
	Formulario	Indica que esta clase puede contener clases que sirvan para el manejo de datos.

En la construcción de este modelo se debe tomar en cuenta como se quiere mostrar la página, que acciones se requieren, que formularios se deben mostrar y como están estructurados todos los elementos; un ejemplo de esto se muestra a continuación: (Ludwig-Maximilians-Universität, 2006)

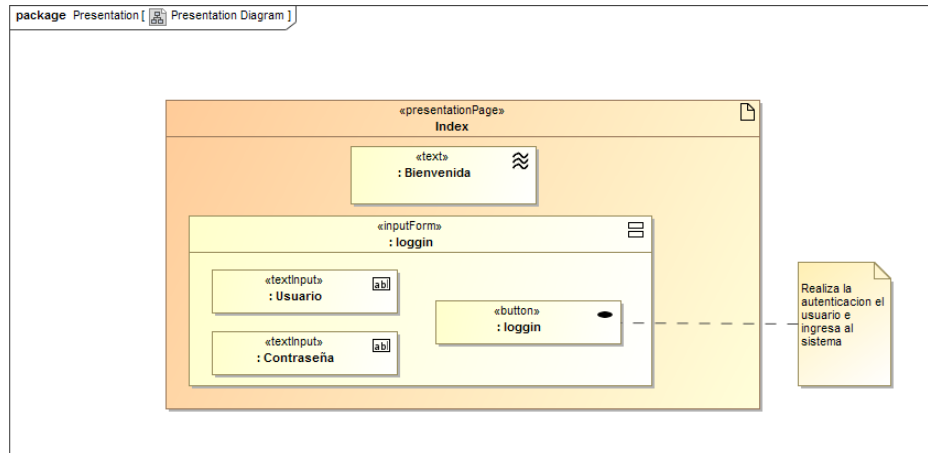


Figura 12. Ejemplo de un Modelo de Presentación.

Modelo de proceso

Este modelo parte del modelo de presentación y permite la visualización de los procesos inmersos en la aplicación Web y su interacción entre ellos. Se definen dos tipos de modelos de procesos, los de estructura y los de flujo.

- **Modelo de estructura de procesos:** gracias a este modelo podemos definir las interacciones de los procesos en la aplicación web. Este diagrama se basa en un diagrama de clases UML, pero cada clase representa un proceso y viene representado por un \gg dentro de la clase. Se muestra un ejemplo: (Ludwig-Maximilians-Universität, 2006)

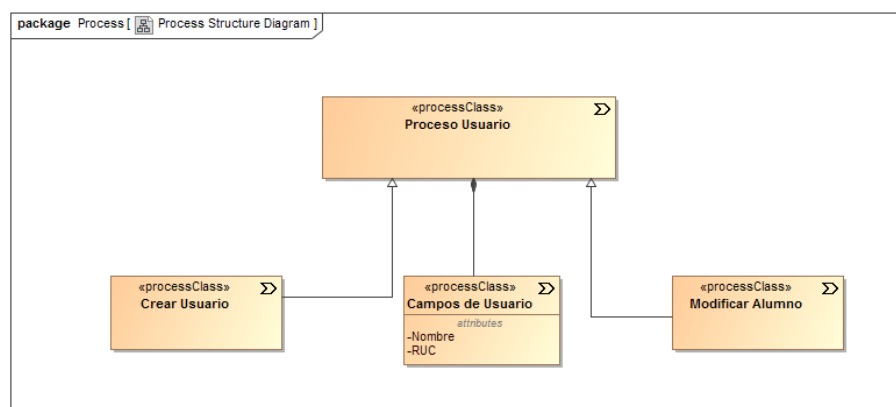


Figura 13. Ejemplo de un Diagrama de Estructura de Procesos.

- Modelo del flujo de procesos:** este diagrama permite ver las actividades que se van a realizar y que están ligadas a los procesos. En este modelo se puede observar que actores van a hacer que actividades y que procesos se generan a partir de dichas actividades. (Ludwig-Maximilians-Universität, 2006) Este diagrama está basado en el diagrama de Actividades UML pero difiere en su forma de presentación. Se muestra un ejemplo: (Nieves Guerrero, Ucán Pech, & Menéndez Domínguez, 2014)

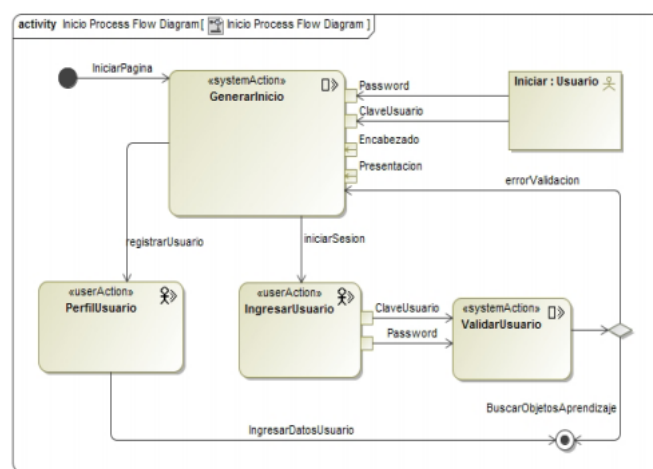


Figura 14. Ejemplo de Diagrama de Flujo de Procesos.

2.2 COMPROBANTES ELECTRÓNICOS

Los comprobantes electrónicos son la representación de las especies tributarias tradicionales como comprobantes de venta, retención y documentos complementarios como mensajes de datos (Ministerio de Coordinación de Seguridad, 2015). Al proceso de emisión de comprobantes electrónicos también se lo conoce como proceso de facturación electrónica.

2.2.1 BASE LEGAL SRI

La base legal sobre la que se fundamenta la emisión de comprobantes electrónicos como reemplazo a la facturación tradicional se encuentra determinada por las siguientes leyes, reglamentos y resoluciones:

- **Ley de Comercio Electrónico, firmas electrónicas y mensajes de datos publicado en el Suplemento del Registro Oficial No. 557 de 17 de abril de 2002:** que brinda todo el marco legal para el uso de firmas electrónicas y la generación de mensajes de datos descrito a lo largo de toda la extensión del documento. (Congreso Nacional, 2002).
- **Reglamento a la Ley de Comercio Electrónico, Firmas y Mensajes de Datos, publicado en el Registro Oficial No. 735 de 31 de diciembre de 2002:** reglamento aplicado a la Ley de Comercio Electrónico expuesto con anterioridad (Noboa Bejarano, 2002).
- **Resolución No. NAC-DGERCGC12-00105 de 09 de marzo de 2012, publicada en Registro Oficial No. 666 de 21 de marzo de 2012:** que define los documentos tributarios a ser adaptados al esquema de comprobantes electrónicos, así como los requisitos y proceso que deben seguir los sujetos pasivos para obtener la autorización de emisión de comprobantes electrónicos (Servicio de Rentas Internas, 2002), sustentado en toda la resolución.
- **Resolución NAC-DGERCGC13-00236, Registro Oficial 956 de 17 de mayo de 2013:** que dispone el calendario de obligatoriedad de emisión de comprobantes electrónicos a empresas privadas (Directora General del Servicio de Rentas Internas, 2014).
- **Resolución NAC-DGERCGC14-00157, Registro Oficial 215 de 31 de marzo de 2014:** que dispone el calendario de obligatoriedad de emisión de comprobantes electrónicos para instituciones y entidades públicas (Director General del Servicio de Rentas Internas, 2014).
- **Resolución NAC-DGERCGC14-00366, Registro Oficial 257 de 30 de mayo de 2014:** que establece la ampliación de los cronogramas de adopción de emisión de comprobantes electrónicos (Servicio de Rentas Internas, 2014).
- **Resolución NAC-DGERCGC14-00790, Registro Oficial 346 de 02 de octubre de 2014:** que expone la normativa para la emisión y autorización de

comprobantes electrónicos por parte de los sujetos pasivos (Servicio de Rentas Internas, 2014).

- **Resolución NAC-DGRCGC14-00788, Registro Oficial 351 de 9 de octubre de 2014:** que define las normas para la transmisión de los comprobantes electrónicos (Servicio de Rentas Internas, 2014).
- **Resolución NAC-DGERCGC15-00000004 publicada en el Registro Oficial 414 de 12 de enero de 2015:** que define la obligatoriedad de emisión de comprobantes electrónicos por parte de notarios (Servicio de Rentas Internas, 2012).

2.2.2 TIPOS DE COMPROBANTES

La resolución No. NAC-DGERCGC12-00105 de 09 de marzo de 2012 define que los comprobantes electrónicos a ser emitidos por los sujetos pasivos serán los siguientes:

1. Facturas
2. Comprobantes de retención
3. Notas de crédito
4. Notas de débito
5. Guías de remisión

Estos comprobantes electrónicos deberán respetar los esquemas XSD dispuestos en el portal del Servicio de Rentas Internas www.sri.gob.ec. Asimismo, se dispone esquemas XML de ejemplo.

2.2.3 INFORMACIÓN TÉCNICA

Firma Electrónica

En Ley de comercio electrónico, firmas electrónicas y mensajes de datos (Ley No. 2002-67), especificada en el Título II, Capítulo 1, Artículo 13 se define a la firma electrónica la siguiente forma: “Son los datos en forma electrónica consignados en un mensaje de datos, adjuntados o lógicamente asociados al mismo, y que puedan ser utilizados para identificar al titular de la firma en relación con el mensaje de datos, e

indicar que el titular de la firma aprueba y reconoce la información contenida en el mensaje de datos.” (Congreso Nacional, 2002)

Firmado electrónico de comprobantes

Los comprobantes electrónicos deben ser firmados bajo la normativa Xades BES. Xades BES -XML Advanced Electronic Signature, Basic Electronic Signature- es el formato de firmado electrónico más básico que cumple con las características y requisitos de la Directiva de Firma Electrónica Avanzada. La versión del esquema a utilizar es la 1.3.2, con codificación UTF-8 y la firma debe ser enveloped o envuelta, es decir, que envuelva a todo el comprobante electrónico XML (Servicio de Rentas Internas, 2015).

Esquemas XSD

Los XML Schema Definition o esquemas XSD son archivos XML en los cuales se define un formato que un documento XML debe cumplir, puntualizando qué valores debe poseer, condiciones, cantidades y restricciones de los mismos. Estos esquemas XSD a menudo son utilizados a modo de plantilla para validar la información contenida en un documento XML.

De acuerdo a la Ficha Técnica de Comprobantes Electrónicos, actualizada a diciembre de 2014, existen dos versiones para los esquemas XML de los comprobantes electrónicos: la versión 1.0.0, versión por omisión de todos los comprobantes y que soporta hasta cantidades numéricas con 2 decimales; y el esquema 1.1.0, que permite la utilización de hasta 6 decimales para precios y cantidades unitarias en las facturas, notas de crédito y comprobantes de retención. Las versiones de los esquemas se aplican para comprobantes electrónicos individuales, así como para la emisión de comprobantes electrónicos en lote.

Los esquemas XSD y XML correspondientes a las versiones 1.0.0 y 1.1.0 pueden ser descargados de la página de Información Técnicas de Comprobantes Electrónicos del SRI: <http://www.sri.gob.ec/de/10116>.

Servicios web

La Administración Tributaria pone a disposición de los contribuyentes servicios web los cuáles se pueden consumir para emitir y autorizar los comprobantes electrónicos. Estos servicios están disponibles para los ambientes de prueba y producción, y definidos en formato Web Service Description Language (WSDL).

El ambiente de pruebas, como su nombre lo indica, debe ser utilizado para probar las aplicaciones de los contribuyentes. Los comprobantes electrónicos emitidos y autorizados bajo este ambiente no poseen validez tributaria. En la Tabla1, se muestra la dirección de las funciones de recepción y autorización del WSDL.

Tabla 4.

WSDL de los servicios del ambiente de pruebas.

FUNCIÓN	WSDL
Recepción	https://celcer.sri.gob.ec/comprobantes-electronicos-ws/RecepcionComprobantes?wsdl
Autorización	https://celcer.sri.gob.ec/comprobantes-electronicos-ws/AutorizacionComprobantes?wsdl

El ambiente de producción permite la emisión y autorización de comprobantes electrónicos con validez tributaria. Este ambiente debe ser consumido una vez que el contribuyente haya realizado las pruebas respectivas de su aplicación. En la Tabla1, se muestra la dirección de las funciones de recepción y autorización del WSDL.

Tabla 5.

WSDL de los servicios del ambiente de producción.

FUNCIÓN	WSDL
Recepción	https://celcer.sri.gob.ec/comprobantes-electronicos-ws/RecepcionComprobantes?wsdl
Autorización	https://celcer.sri.gob.ec/comprobantes-electronicos-ws/AutorizacionComprobantes?wsdl

Formato RIDE

El formato RIDE corresponde a la representación impresa del documento electrónico. Este formato, si bien no posee validez legal ni tributaria al no estar firmado electrónicamente como su versión en XML, posee la clave de acceso y número de autorización del comprobante electrónico. El formato RIDE debe ser enviado por correo electrónico junto a su versión en XML al responsable de la actividad comercial e impreso en los siguientes escenarios:

1. Cuando no se proporcione un correo electrónico o consentimiento del receptor.
2. Cuando el receptor del comprobante electrónico sea identificado como consumidor final.
3. Cuando se requiera de manera explícita el mismo por parte del receptor.

Los campos y formato que debe respetar el formato RIDE está expresa en la ficha técnica de comprobantes electrónicos, tal como se evidencia en la Fig.15 (Servicio de Rentas Internas, 2015)

 <p>SOFYA Sistemas Operativos Fáciles y Amigables</p>		<p>RUC: 1715960363001</p> <p>FACTURA</p> <p>No. 001-002-00000036</p> <p>NÚMERO DE AUTORIZACIÓN: 3001201510501217159603630010212076272</p> <p>FECHA Y HORA DE AUTORIZACIÓN: 2015-01-30 10:50:12.434-05:00</p> <p>AMBIENTE: PRUEBAS</p> <p>EMISIÓN: NORMAL</p> <p>CLAVE DE ACCESO:</p>  <p>300120150117159603630011001002000000361980868217</p>					
<p>JUAN ESTEBAN CABRERA</p> <p>DIR. MATRIZ: AV. AMAZONAS Y NNUU</p> <p>DIR. SUCURSAL: AV. AMAZONAS Y NNUU.</p> <p>CONTRIBUYENTE ESPECIAL No. 5/N</p> <p>OBLIGADO A LLEVAR LA CONTABILIDAD: SI</p>		<p>RAZON SOCIAL / NOMBRES Y APELLIDOS: MATUS CORONEL ABRAHAM ISRAEL</p> <p>FECHA EMISIÓN: 30/01/2015</p> <p>IDENTIFICACIÓN: 1717379257</p> <p>GUÍA DE REMISIÓN: 001-001-000000001</p>					
COD. PRINCIPAL	COD. AUXILIAR	CANT.	DESCRIPCIÓN	DETALLE ADICIONAL	PRECIO UNITARIO	DESC.	PRECIO TOTAL
CAP-001	000	20.00	Capacitación JSF	Color: Rojo; Talla: 1 metro;	15.00	0.00	300.00
CAP-001	000	20.00	Capacitación JSF	Color: Rojo; Talla: 1 metro;	15.00	0.00	300.00
CAP-001	000	20.00	Capacitación JSF	Color: Rojo; Talla: 1 metro;	15.00	0.00	300.00
<p>INFORMACIÓN ADICIONAL</p> <p>TELEFONO: 0994148769</p> <p>DIRECCION COMPRADOR: CONOCOTO</p>					<p>SUBTOTAL 12% 900.00</p> <p>SUBTOTAL 0% 0.00</p> <p>SUBTOTAL NO OBJETO DE IVA% 0.00</p> <p>SUBTOTAL SIN IMPUESTOS 900.00</p> <p>DESCUENTO 0.00</p> <p>ICE 0.00</p> <p>IVA 12% 108.00</p> <p>IRBPNR 0.00</p> <p>PROPINA 0.00</p> <p>VALOR TOTAL 1008.00</p>		

Figura 15. Ejemplo de factura en formato RIDE.

2.3 SOFTWARE COMO SERVICIO

2.3.1 DEFINICIÓN

El Software como Servicio, más comúnmente conocido como SaaS, es un modelo de distribución de computación en la nube en el cual se proporciona a los usuarios el acceso a aplicaciones web utilizando el protocolo HTTP/S.

2.3.2 CARACTERÍSTICAS

El Software como Servicio se diferencia de los servicios de desarrollo de software tradicionales debido a las siguientes características:

- El software consumido se encuentra alojado en servidores ya sean locales o públicos en lugar de las estaciones de trabajo de cada usuario.
- Se necesita de acceder al mismo a través de una red (intranet o Internet).
- Se reduce el costo en licenciamiento por máquinas y se adopta el pago de suscripciones mensuales o anuales por el alquiler del software.
- Se reduce el costo en infraestructura de hardware, así como los costos operativos de la misma.
- El servicio consumido es proporcionado de manera transparente al usuario, pero este no tiene acceso al código fuente, ni puede realizar modificaciones.
- Las mejoras continuas y parches del software son implementadas de manera transparente al usuario sin la necesidad que este o equipo de TICs intervenga.

2.3.3 ALTERNATIVAS

Para la implantación de un proyecto SaaS es necesario contar con la infraestructura adecuada para su correcto, eficaz y eficiente servicio. Para lograr esto se pueden utilizar nubes públicas, privadas e híbridas.

Nube Pública

La nube pública es aquella que es ofrecida por terceros a una amplia gama de usuarios como un servicio producto de una actividad comercial. Este tipo de nube es implementada y mantenida por los proveedores, mientras los clientes y usuarios simplemente hacen uso de la misma, mas no tienen control sobre su funcionamiento interno. Los modelos de distribución de computación en la nube brindados por estas empresas son Infraestructura como Servicio (IaaS), plataforma como servicio (PaaS) y Software como Servicio (SaaS). Entre los beneficios de la nube pública podemos listar los siguientes:

- Fácil acceso de la infraestructura deseada.
- Reducción de costos infraestructura física.
- Reducción de costos operacionales y de mantenimiento.

Entre las nubes públicas más populares y que dominan el mercado según el cuadrante mágico de Gartner (Leong, Toombs, & Gill, 2015) encontramos:

- **Amazon Web Services:** es el servicio de computación de la empresa estadounidense Amazon. Provee IaaS y PaaS ya sea con sus instancias EC2, RDS o su servicio de balanceo de carga. Así mismo cuenta con asistentes para la instalación de soluciones informáticas de terceros como servidores web, aulas virtuales, tiendas online, etc., con un simple clic, aproximándose a un SaaS.
- **Google Cloud:** es la plataforma de computación en la nube de Google. Provee asimismo de IaaS con su Google Compute Engine y de PaaS con su Google App Engine. A diferencia de Amazon Web Services, Google Cloud factura cada 10 minutos de consumo de sus servicios.
- **Microsoft Azure:** es la plataforma de computación en la nube de Microsoft. Como es de suponer, Microsoft ofrece el servicio de IaaS para máquinas virtuales Windows y GNU/Linux. Un punto que destaca a Microsoft Azure es que se factura por cada minuto de consumo de su servicio.
- **Digital Ocean:** es otra plataforma de computación en la nube IaaS ampliamente conocida y caracterizada por enfocarse en distribuciones GNU/Linux/*Nix y aplicaciones de software libre y código abierto. Así como

Amazon Web Services posee instalaciones de aplicaciones y plataformas con un solo clic, lo que facilita el SaaS.

Nube privada

La nube privada es aquella nube que está disponible dentro de una Intranet o WAN privada para el uso interno de los usuarios autorizados. Comúnmente este tipo de nube es usada dentro de las instituciones y entidades para compartir software y servicios entre áreas y departamentos, así como para mantener centralizado la información de la misma. En este tipo de nube, la institución es quien la implementa y se encarga del mantenimiento de la misma, lo que permite que todos los procesos e información estén bajo su control.

Entre los beneficios de una nube privada encontramos:

- Control total sobre los procesos y operaciones.
- Centralización de contenidos e información.
- Fácil acceso a los recursos informáticos por parte de todas las áreas empresariales.

Nube Híbrida

La nube híbrida, como su nombre sugiere, es la combinación de la nube privada con la pública, en la cual un porcentaje de servicios se encuentran bajo el control de la institución privada y el resto es provisto por una tercera parte. Esta nube permite distribuir la carga de funciones, siendo aquellos procesos críticos para la empresa los que se encuentran en la parte privada mientras el restante es administrado y mantenido por una tercera parte.

CAPÍTULO III – INICIACIÓN Y ELABORACIÓN

3.1 INICIACIÓN

En la fase de levantamiento de requerimientos para la aplicación web se utilizó la técnica lluvia de ideas o brainstorming, apoyados en la documentación legal y técnica proporcionada por el SRI para la comprensión de los requerimientos. La lluvia de ideas se encuentra respaldada por un formato de historias de usuarios utilizado por SOFYA S.A. para documentar los requerimientos proporcionada en el Anexo 1.

El modelo de requerimientos establecido por la metodología UWE define un modelo de casos de uso general y modelos de actividad los cuales se especifican en las figuras 16 – 27.

3.1.1. MODELO DE CASOS DE USO

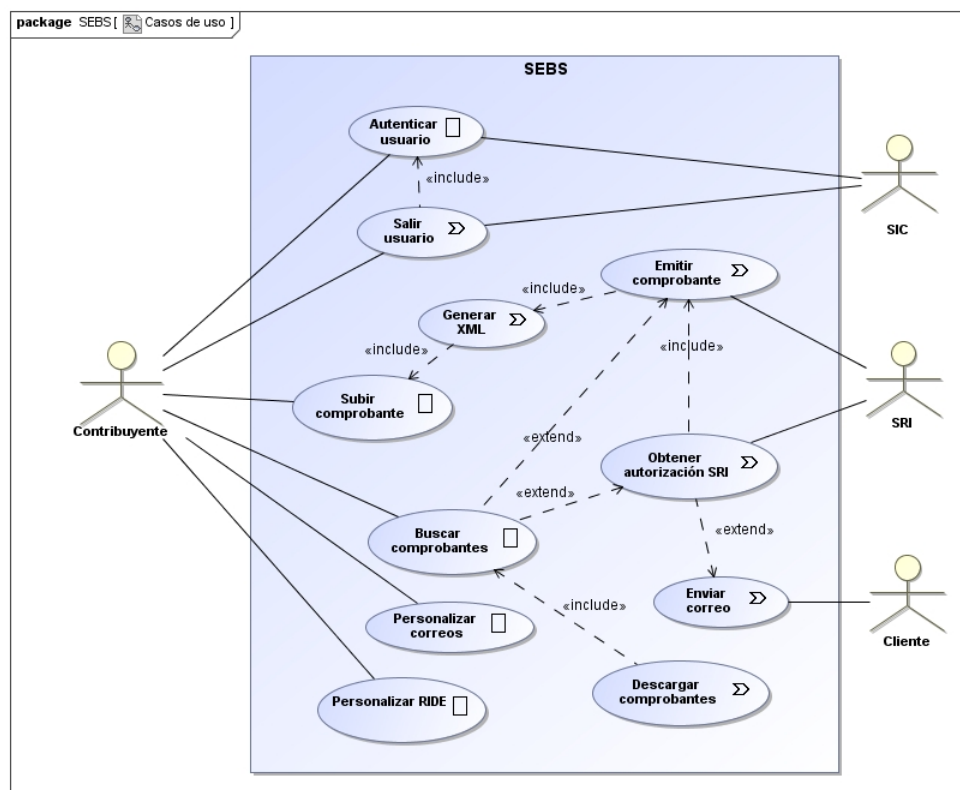


Figura 16. Diagrama de casos de uso de la aplicación.

3.1.2. DIAGRAMA DE ACTIVIDAD – AUTENTICAR USUARIO

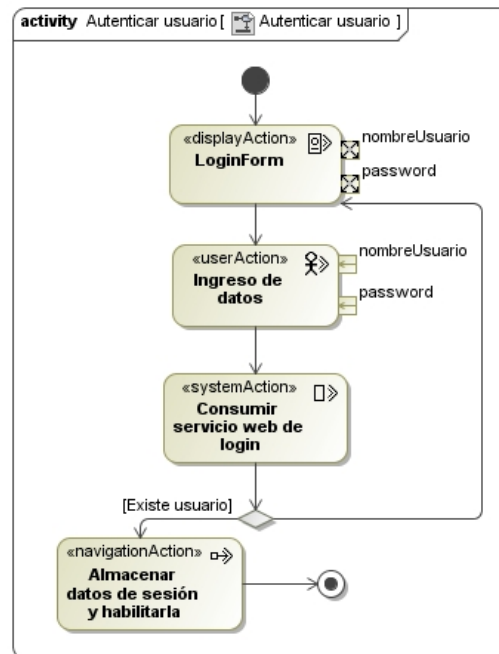


Figura 17. Diagrama de actividad del caso de uso Autenticar Usuario.

3.1.3. DIAGRAMA DE ACTIVIDAD – SUBIR COMPROBANTE

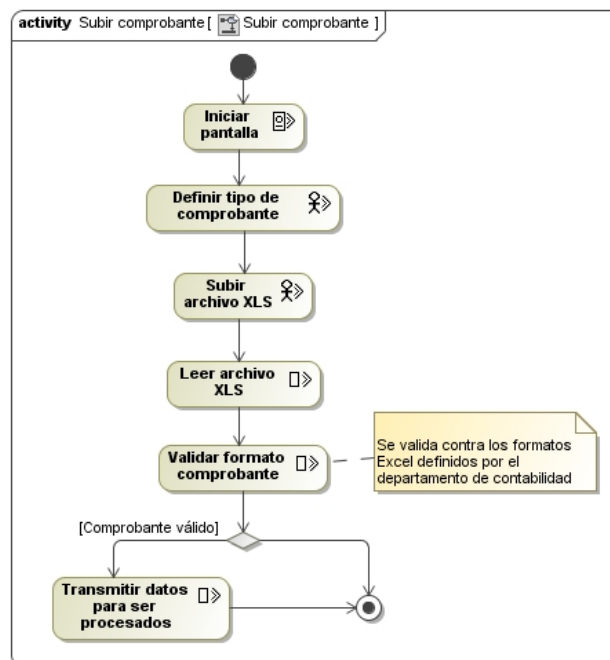


Figura 18. Diagrama de actividad del caso de uso Subir Comprobante.

3.1.4. DIAGRAMA DE ACTIVIDAD – GENERAR XML

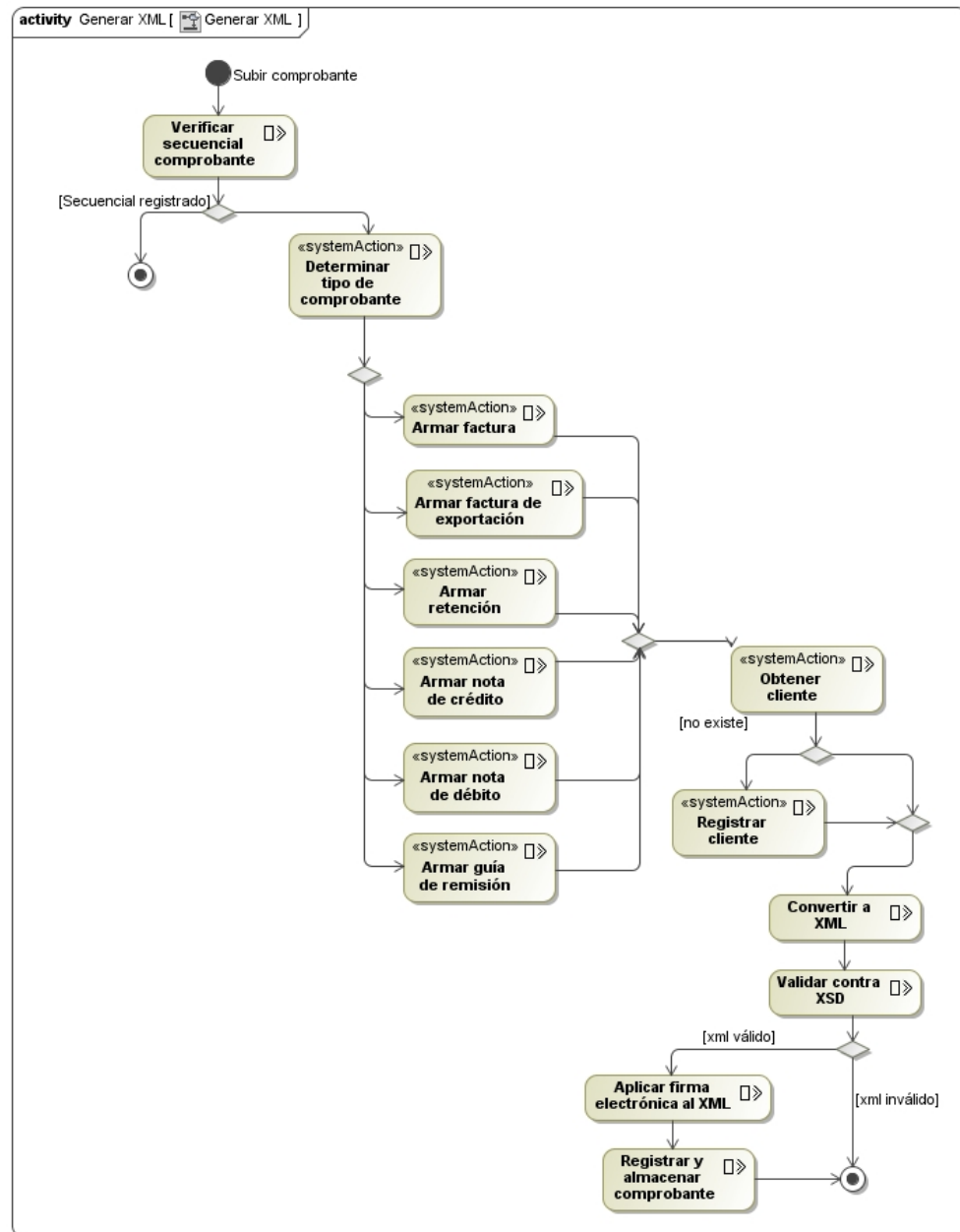


Figura 19. Diagrama de actividad del caso de uso Generar XML.

3.1.5. DIAGRAMA DE ACTIVIDAD – EMITIR COMPROBANTE SRI

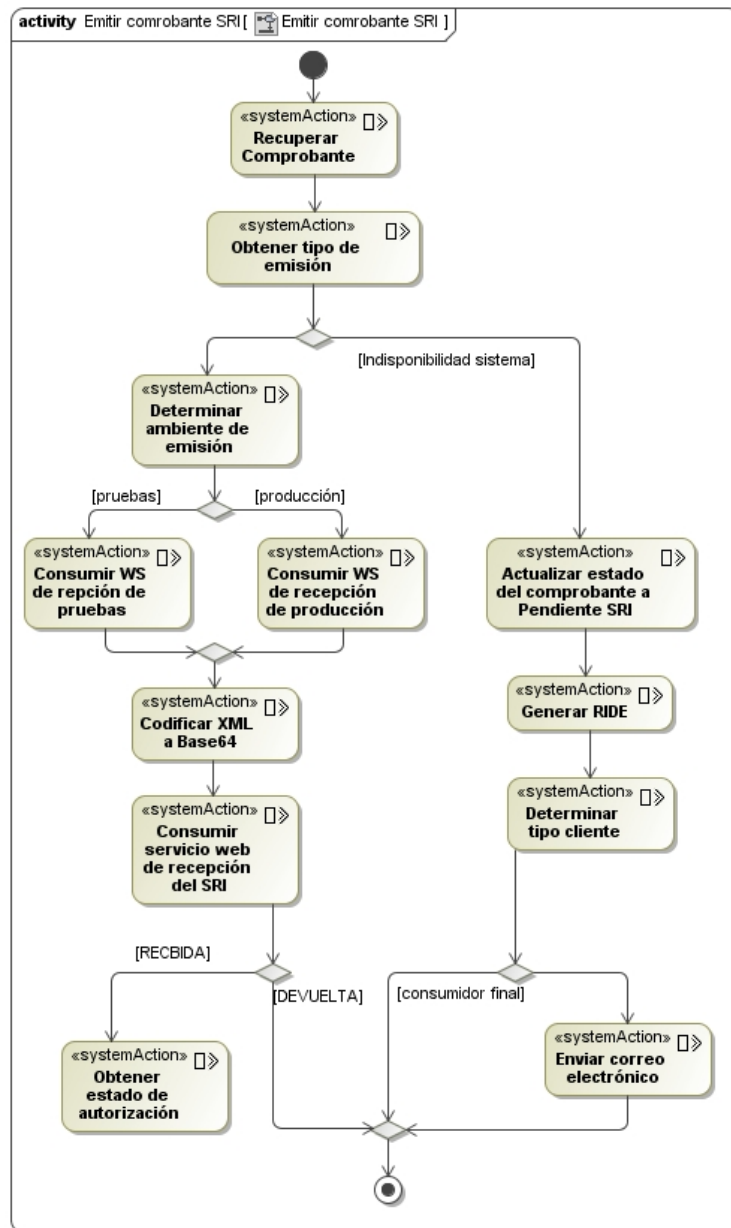


Figura 20. Diagrama de actividad del caso de uso Emitir Comprobante.

3.1.6. DIAGRAMA DE ACTIVIDAD – OBTENER AUTORIZACIÓN SRI

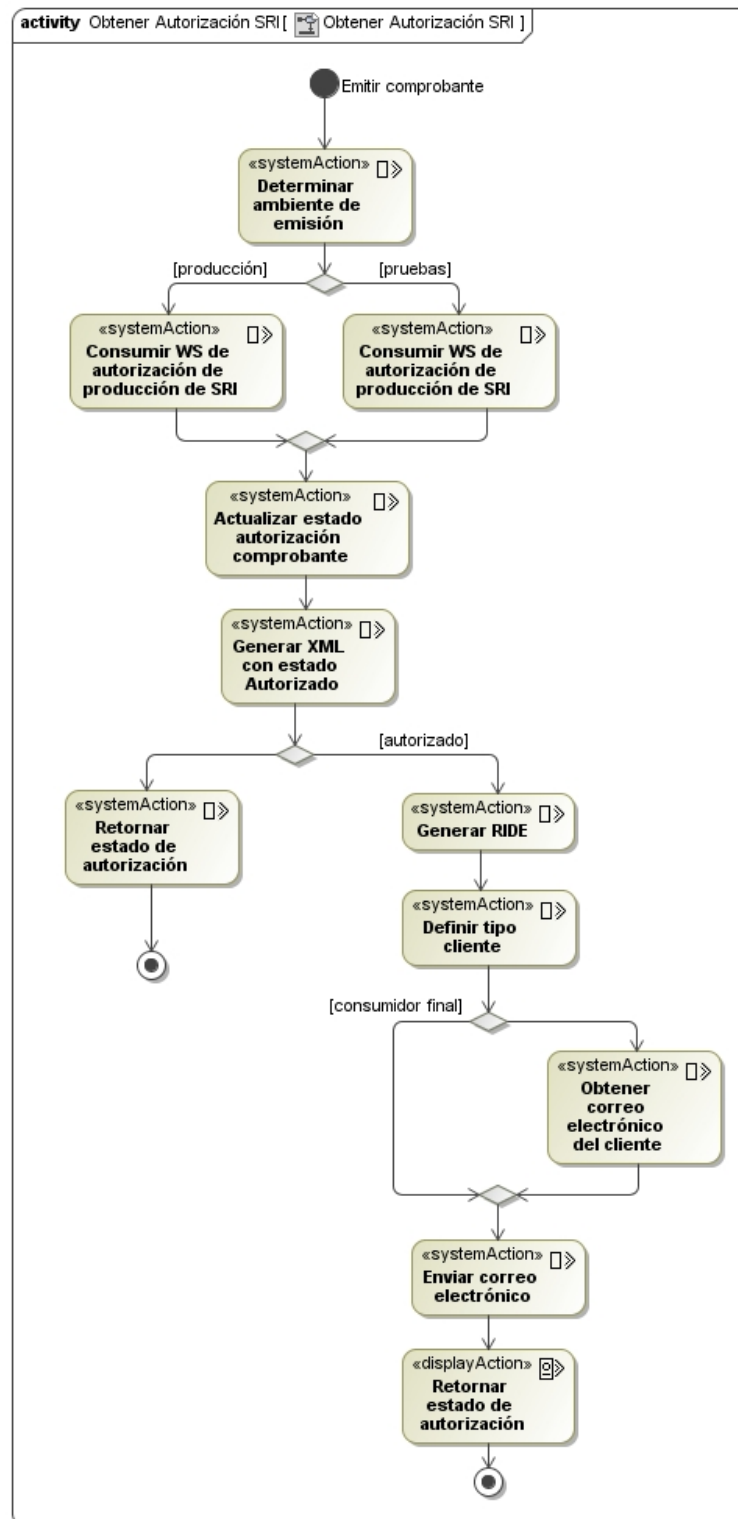


Figura 21. Diagrama de actividad del caso de uso Obtener Autorización SRI.

3.1.7. DIAGRAMA DE ACTIVIDAD – ENVIAR CORREO

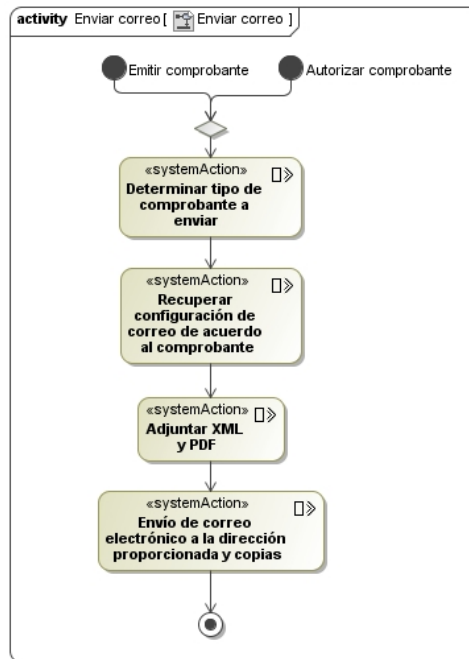


Figura 22. Diagrama de actividad del caso de uso Enviar Correo.

3.1.8. DIAGRAMA DE ACTIVIDAD – BUSCAR COMPROBANTES

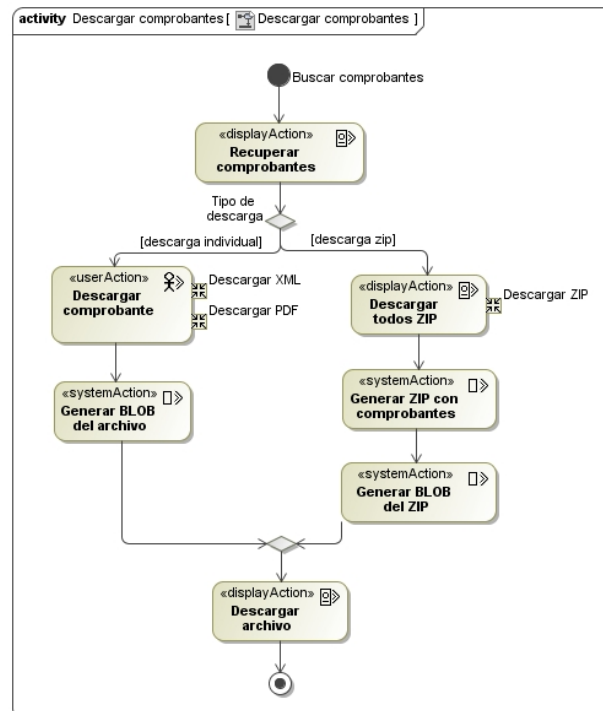


Figura 23. Diagrama de actividad del caso de uso Buscar Comprobantes.

3.1.9. DIAGRAMA DE ACTIVIDAD – DESCARGAR COMPROBANTES

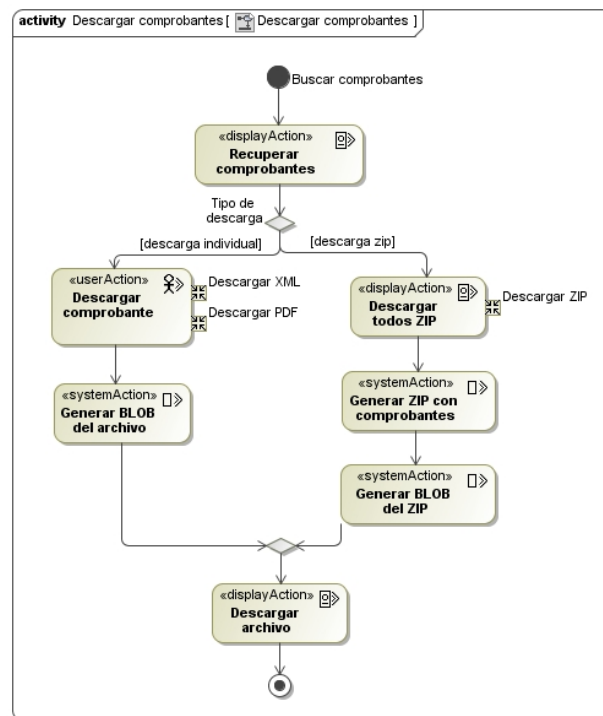


Figura 24. Diagrama de actividad del caso de uso Descargar Comprobantes.

3.1.10. DIAGRAMA DE ACTIVIDAD – PERSONALIZAR CORREOS

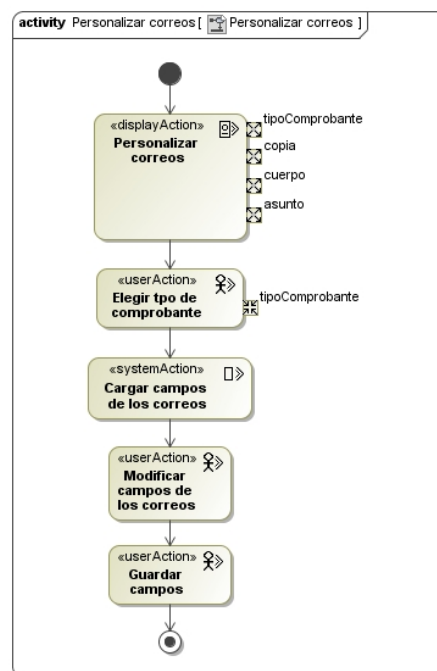


Figura 25. Diagrama de actividad del caso de uso Personalizar Correos.

3.1.11. DIAGRAMA DE ACTIVIDAD – PERSONALIZAR RIDE

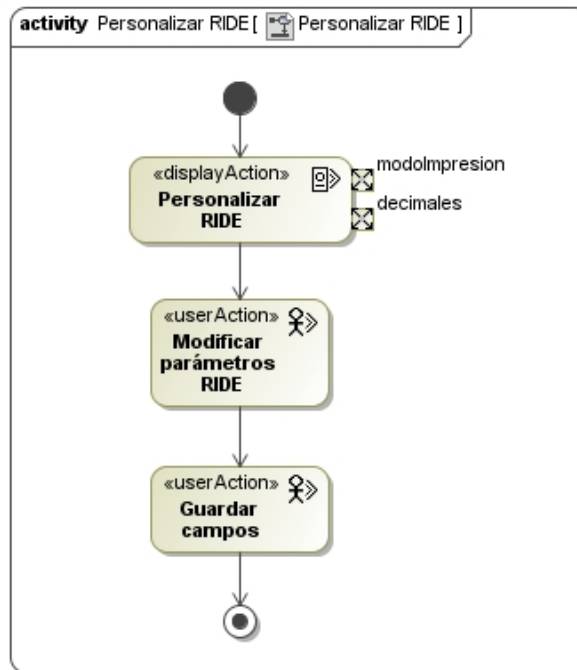


Figura 26. Diagrama de actividad del caso de uso Personalizar RIDE.

3.1.12. DIAGRAMA DE ACTIVIDAD – SALIR USUARIO

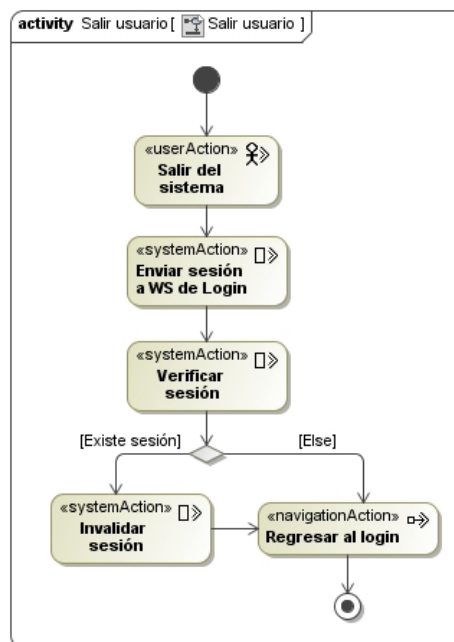


Figura 27. Diagrama de actividad del caso de uso Salir Usuario.

Como lineamiento de SOFYA S.A., los requerimientos deben ser documentados bajo la norma establecida por la empresa, para la Especificación de requisitos de software (SRS) basados en el estándar IEEE 830. Este documento se proporciona como Anexo 2.

3.2 ELABORACION

La elaboración o diseño de la aplicación web se la realizó de acuerdo a los entregables establecidos por la metodología UWE, los cuales son: modelo de contenido, modelo de navegación, modelo de presentación y modelo de proceso.

4.2.1. MODELO DE CONTENIDO

El modelo de contenido, como lo define la metodología UWE, está representado en la figura 28 por un diagrama de clases sin elementos ni notaciones adicionales o específicas como en los diagramas de uso expuestos con anterioridad.

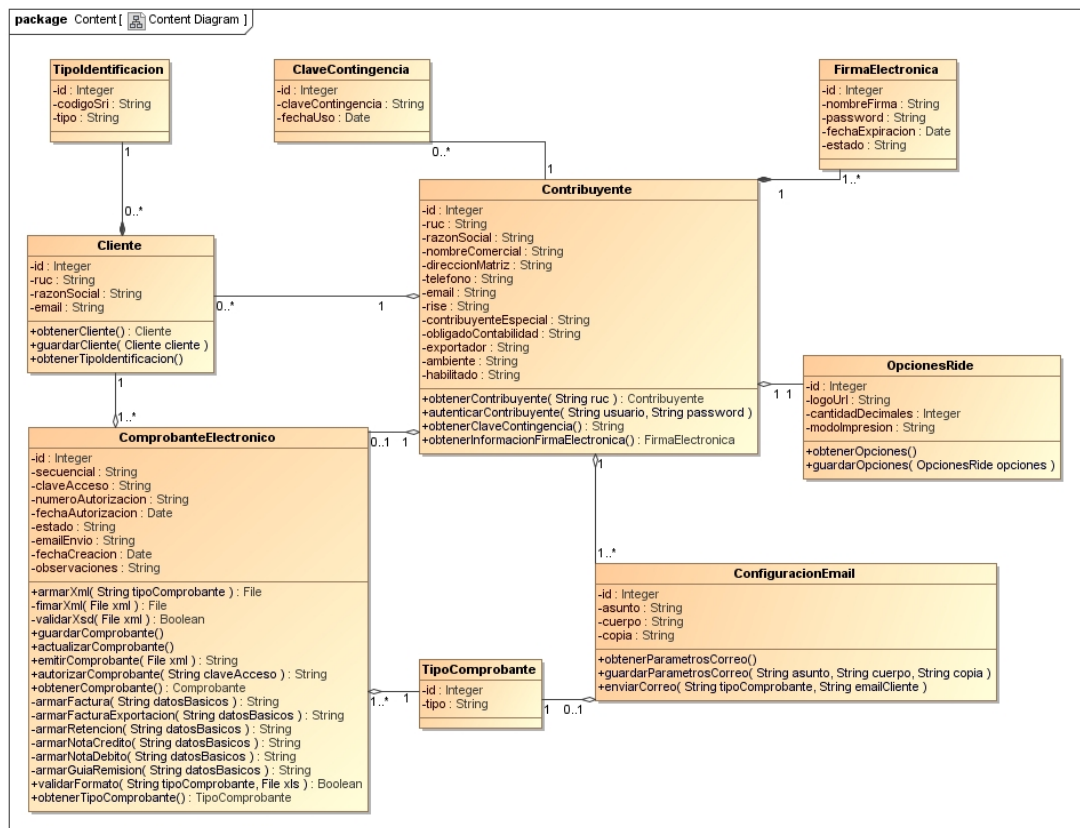


Figura 28. Diagrama de clases que define al modelo de contenido.

4.2.2. MODELO DE NAVEGACIÓN

El modelo de navegación determina la forma en que el usuario navega por la aplicación web y que procesos se involucran en las actividades realizadas.

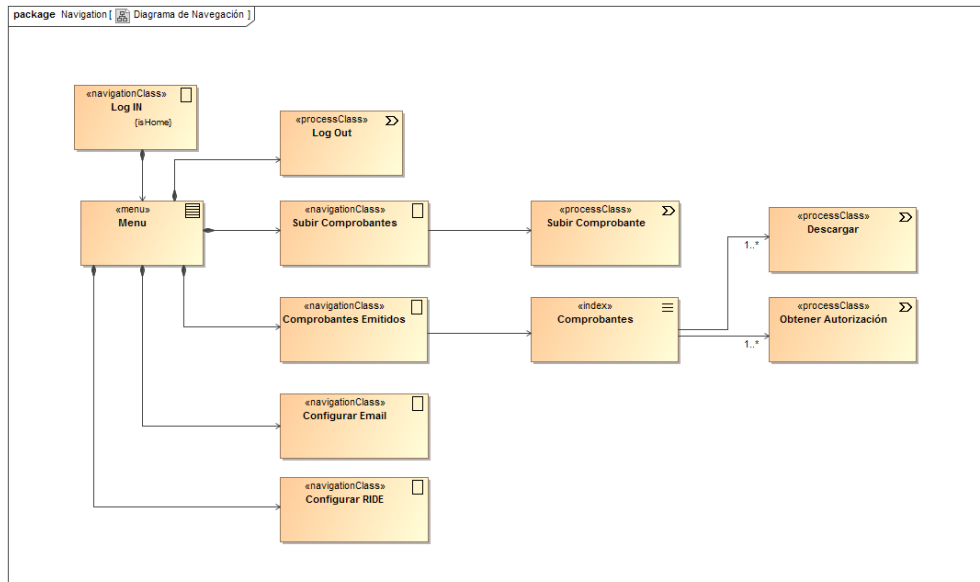


Figura 29. Diagrama de navegación de la aplicación.

4.2.3. MODELO DE PRESENTACIÓN

El modelo de presentación se encarga de exhibir los elementos gráficos de la aplicación por cada una de las pantallas diseñadas en el modelo de navegación.

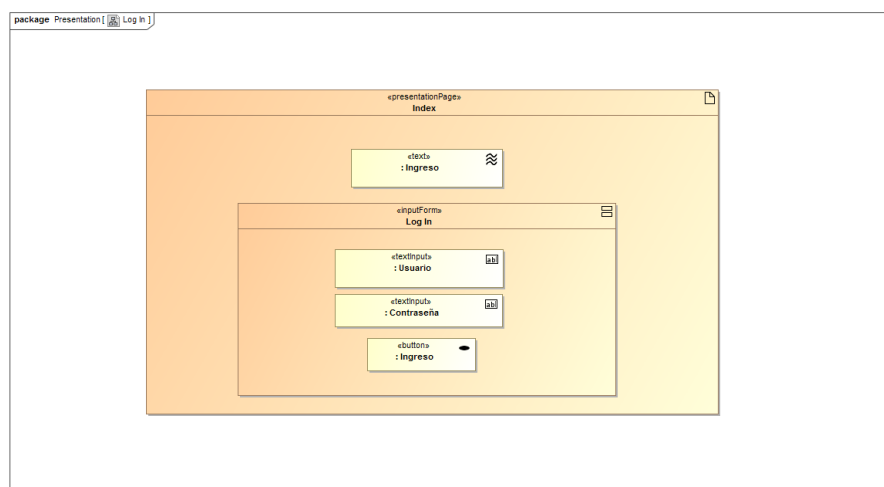


Figura 30. Modelo de presentación de la pantalla de autenticación.

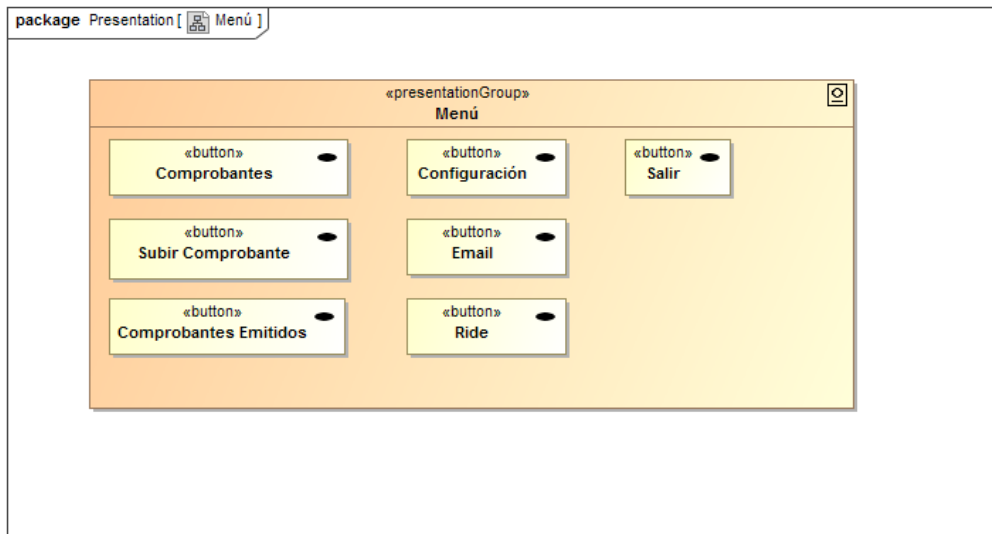


Figura 31. Modelo de presentación del menú.

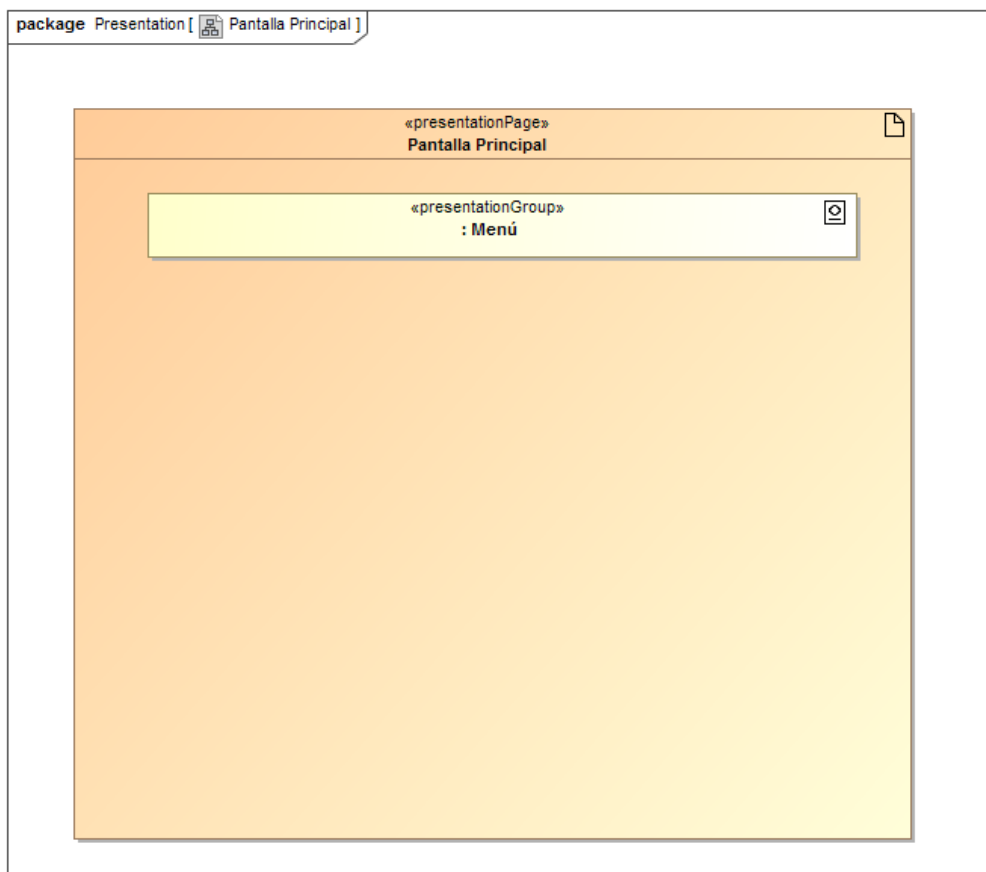


Figura 32. Modelo de presentación de la pantalla principal.

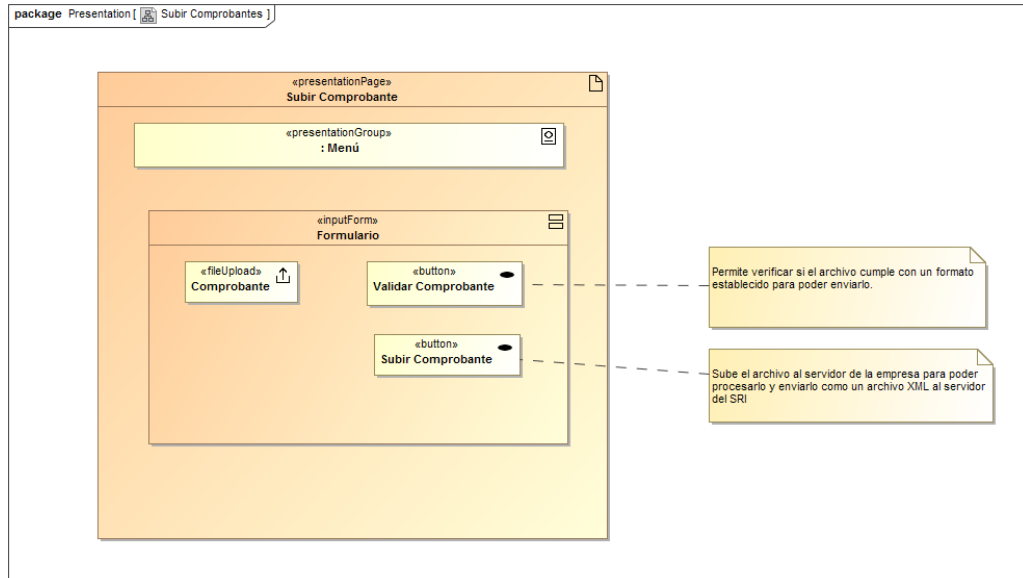


Figura 33. Modelo de presentación de la pantalla de subida de comprobantes.

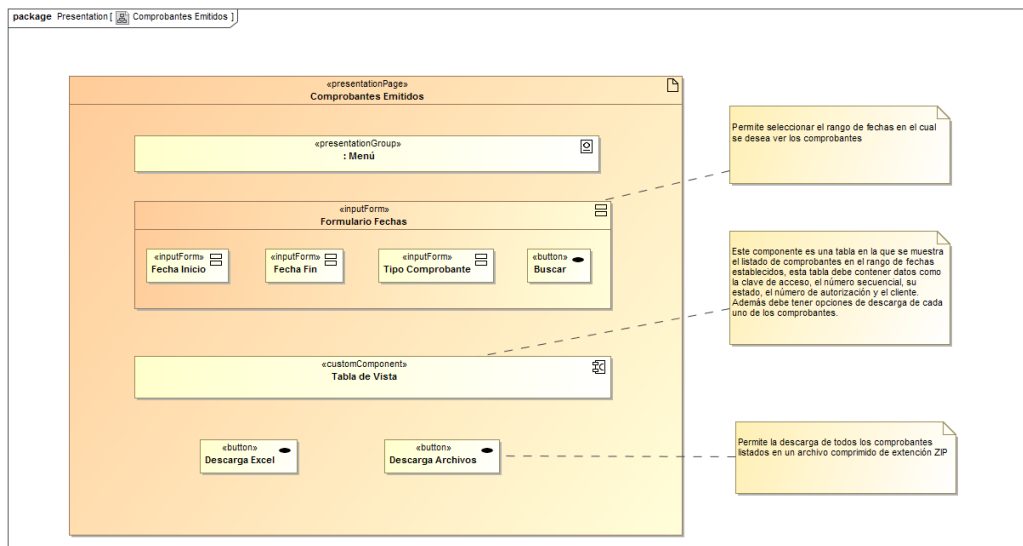


Figura 34. Modelo de presentación de la pantalla de comprobantes emitidos.

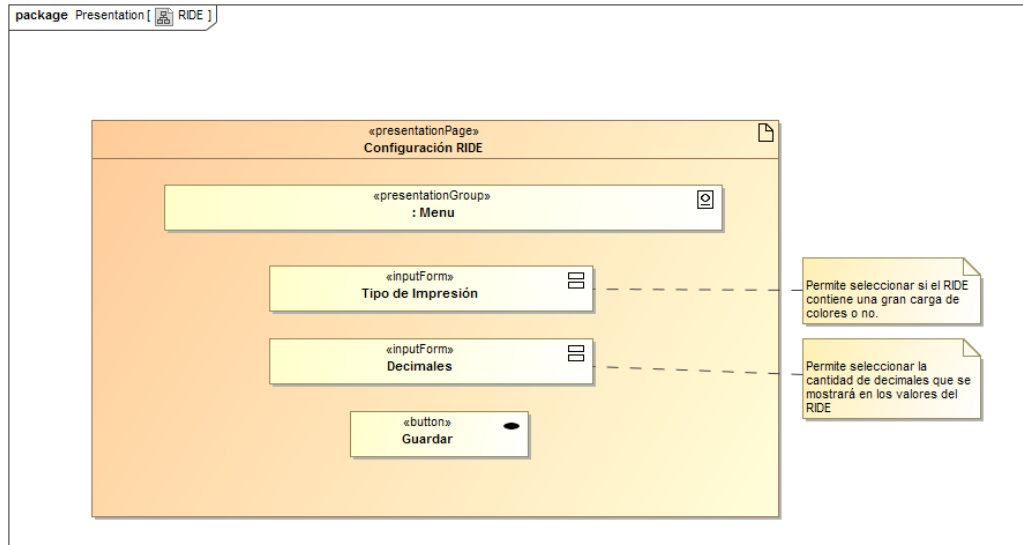


Figura 35. Modelo de presentación de la pantalla de configuración de RIDE.

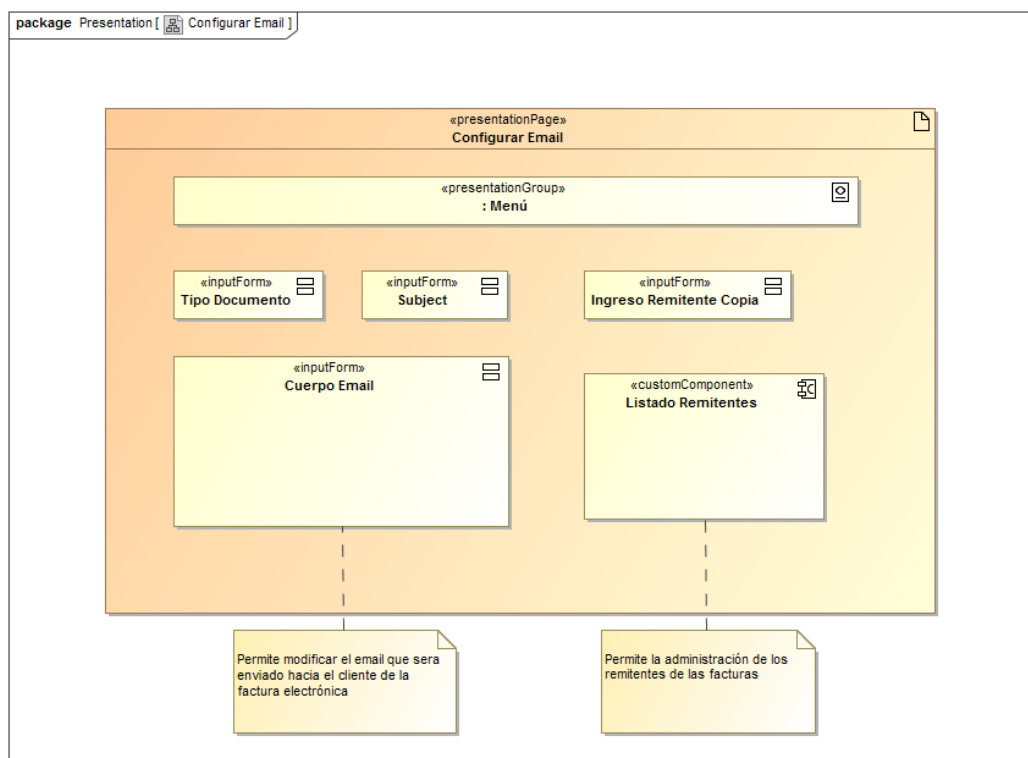


Figura 36. Modelo de presentación de la pantalla de configuración de correos electrónicos.

4.2.4. MODELO DE PROCESO

El modelo de proceso detalla a más profundidad las acciones y procesos involucrados en cada una de las actividades y funcionalidad de la aplicación web,

como, por ejemplo, sus flujos principales y alternativos. El modelo de estructura de procesos nace del modelo de navegación y cada una de sus clases es representada a profundidad por un modelo de flujo de proceso.

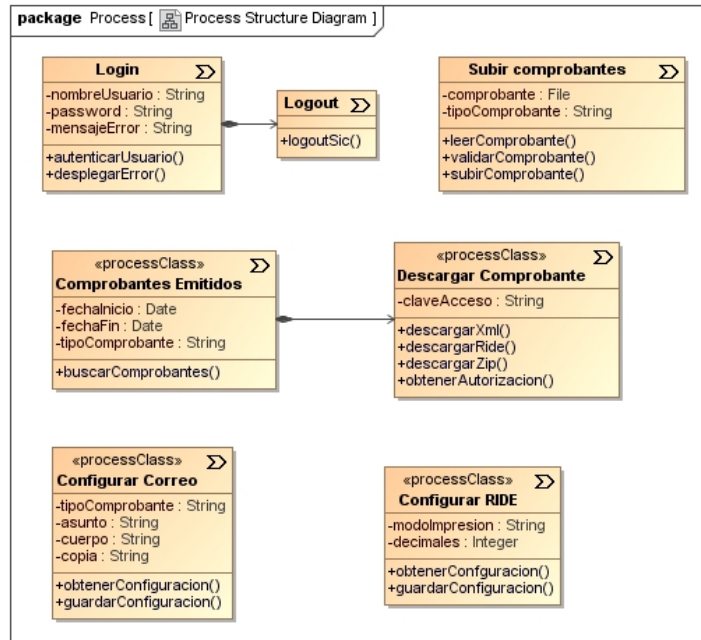


Figura 37. Diagrama de clases que representa el modelo de Estructura de Procesos.

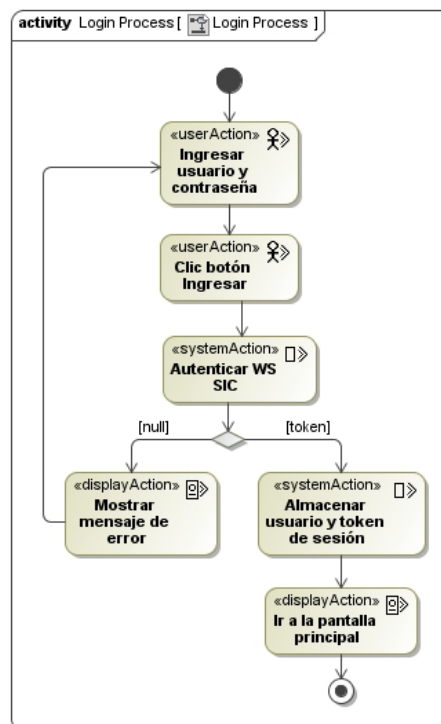


Figura 38. Diagrama de actividad que define el flujo de procesos para Login.

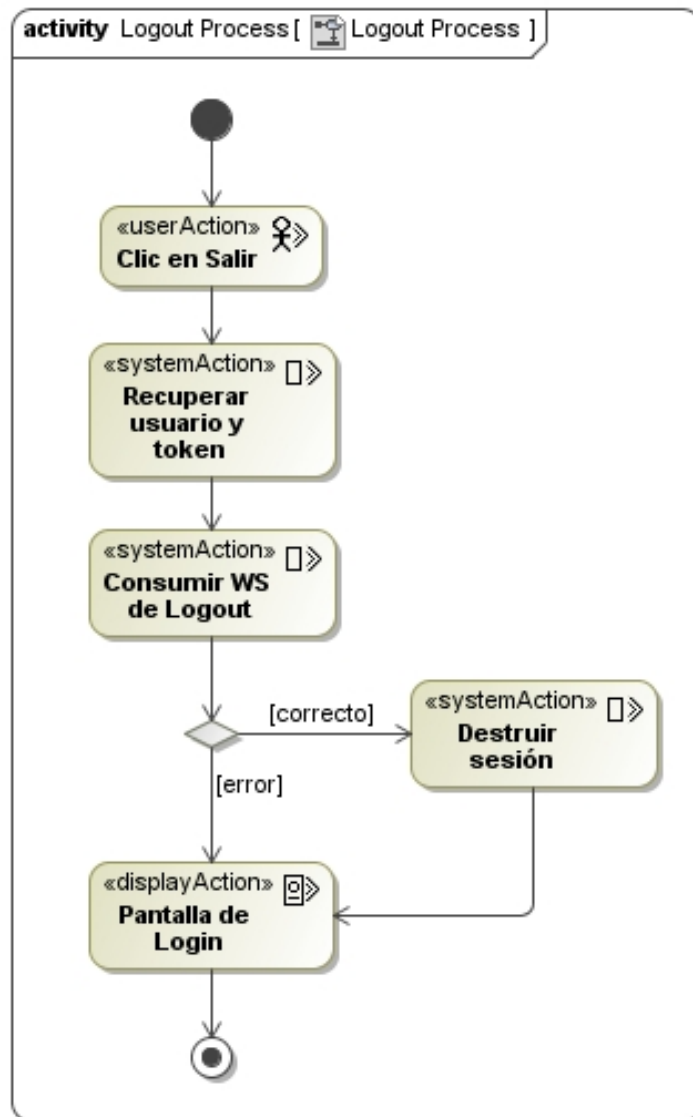


Figura 39. Diagrama de actividad que define el flujo de procesos para Logout.

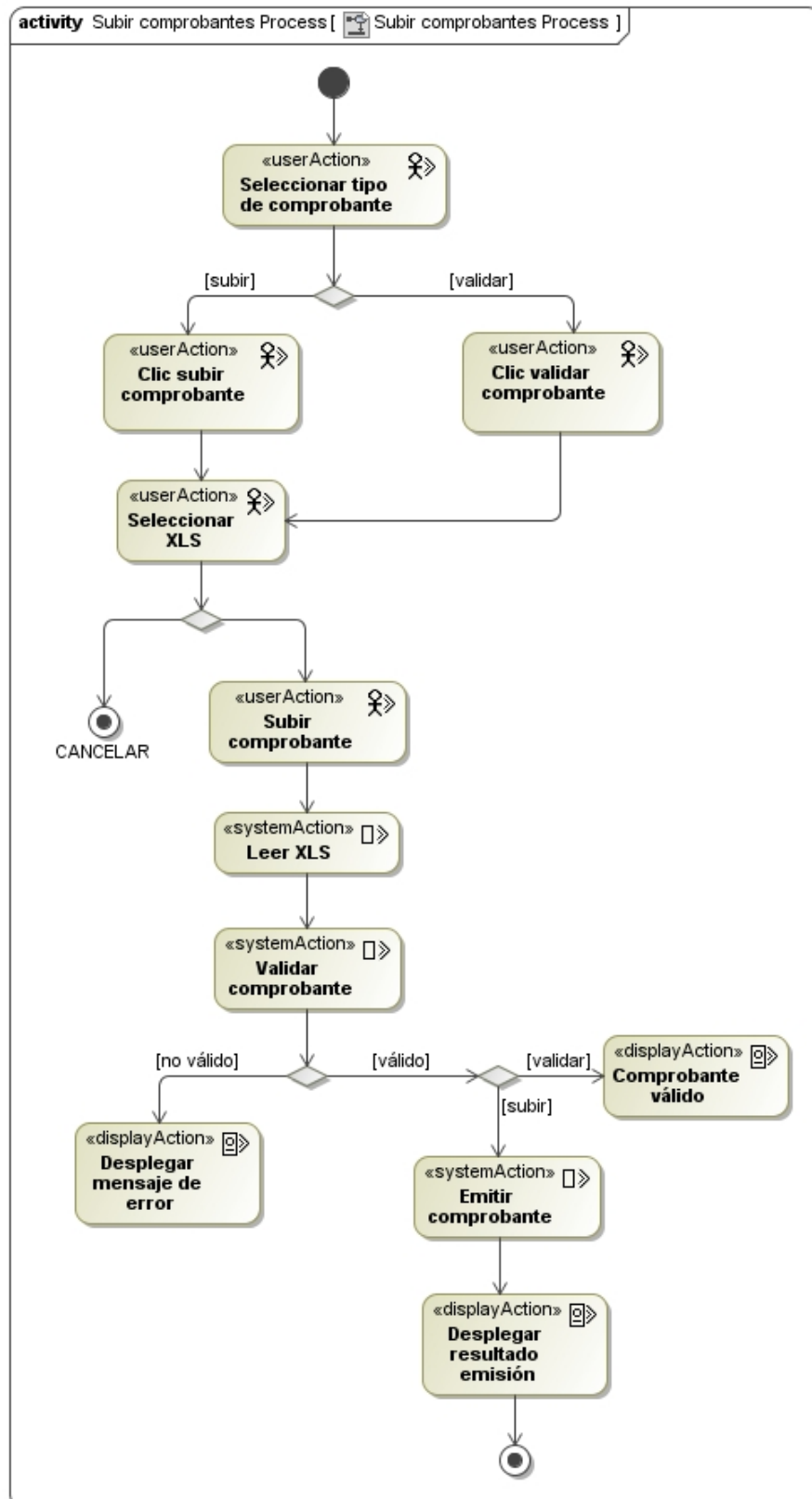


Figura 40. Diagrama de actividad que define el flujo de procesos para Subir Comprobantes.

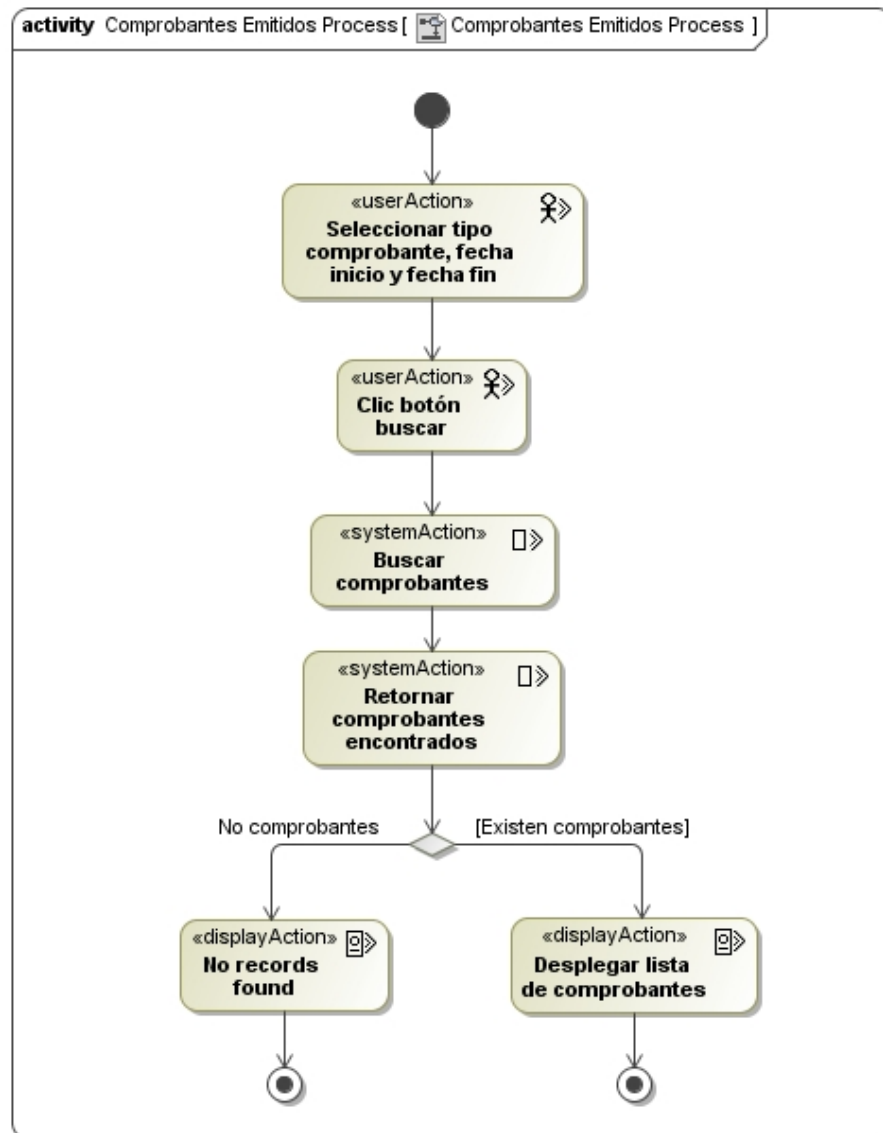


Figura 41. Diagrama de actividad que define el flujo de procesos para Comprobantes Emitidos.

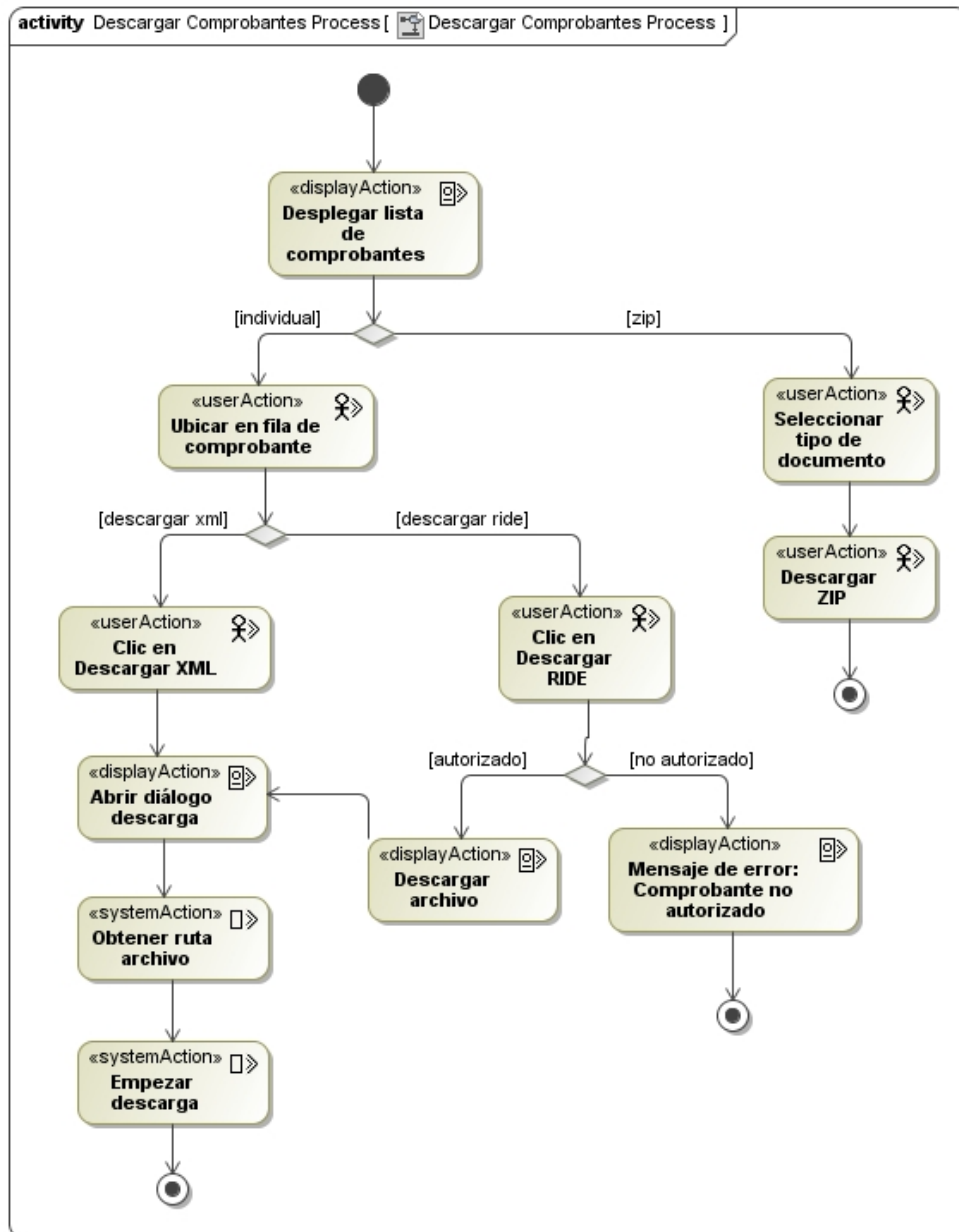


Figura 42. Diagrama de actividad que define el flujo de procesos para Descargar Comprobantes.

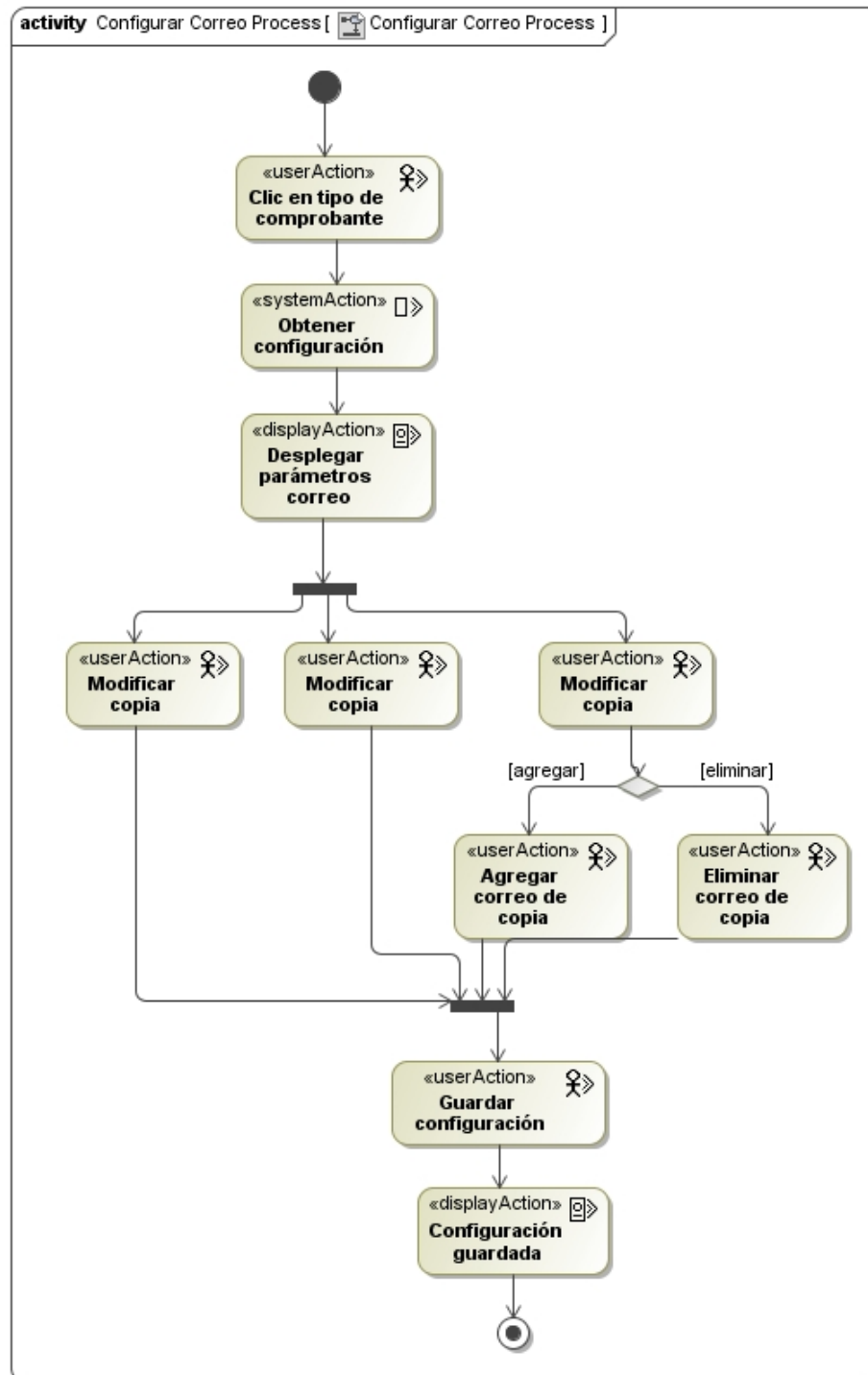


Figura 43. Diagrama de actividad que define el flujo de procesos para Configurar Correo.

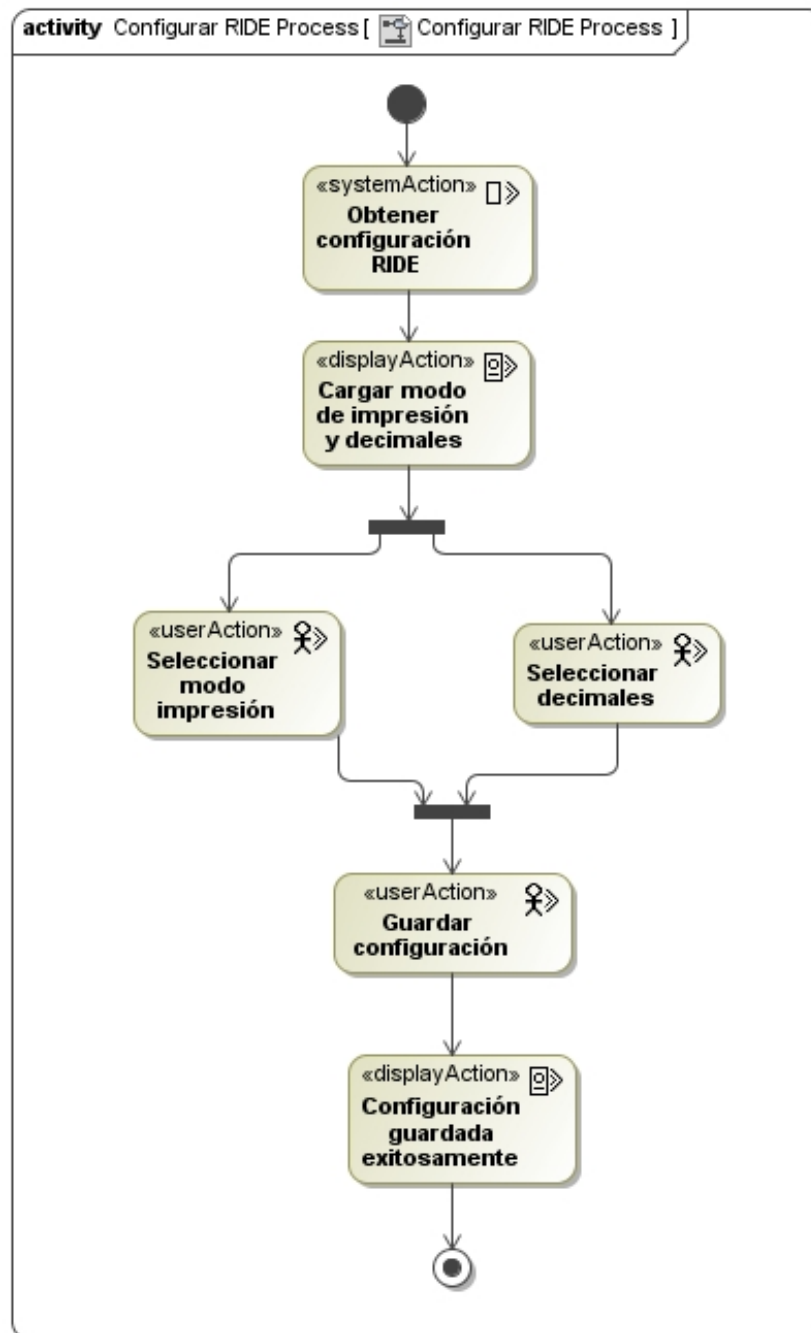


Figura 44. Diagrama de actividad que define el flujo de procesos para Configurar RIDE.

3.3 ARQUITECTURA DE SOFTWARE

Como se establece en la especificación de requisitos de software, la aplicación web será un sistema distribuido en múltiples capas, las cuales se detallan en la siguiente figura:

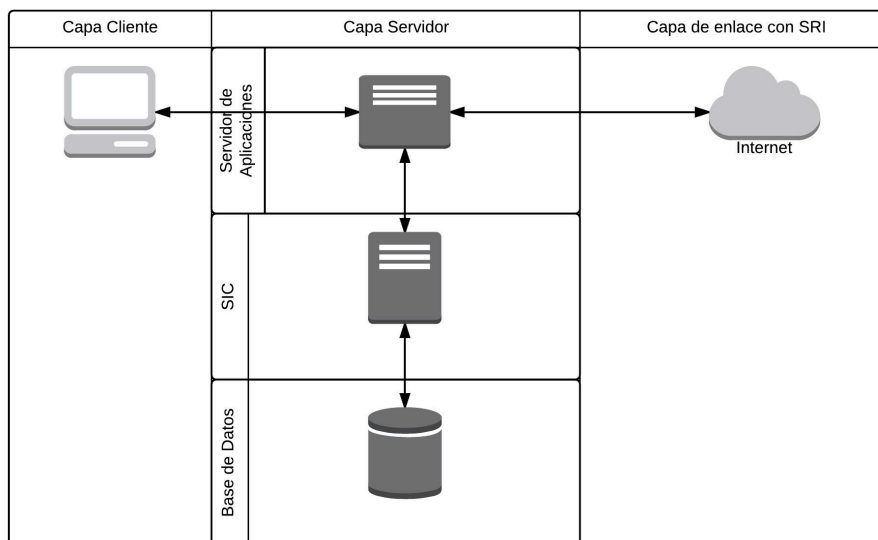


Figura 45. Arquitectura en múltiples capas de la aplicación.

La capa de cliente corresponde a los contribuyentes que acceden a la aplicación desde su navegador web.

La capa de servidor comprende al servidor de aplicaciones JEE en donde será desplegada la aplicación web, la capa de SOFYA Integrated Core (SIC) que actúa como un middleware entre las aplicaciones y la persistencia de datos, y la capa de base de datos, comprendida por el servidor MySQL.

Existe una capa externa de la cual depende la aplicación que corresponde a los servicios del SRI, los cuales proporcionan los mecanismos para la emisión y autorización de comprobantes electrónicos.

CAPÍTULO IV – CONSTRUCCIÓN Y TRANSICIÓN

4.1 CONSTRUCCIÓN

En la metodología de desarrollo de software UWE, la etapa de construcción de software no define los artefactos ni actividades a desarrollar y ejecutar, por lo que se siguieron las guías internas de codificación limpia de la empresa SOFYA SYSTEMS S.A. Esta guía se especifica en el Anexo 3.

El lenguaje utilizado para desarrollar la aplicación web es Java debido a la afinidad de la empresa auspiciante, SOFYA S.A., y las tecnologías utilizadas por el SRI para el firmado y validación de comprobantes electrónicos. Y por último, las herramientas y librerías utilizadas para el desarrollo de este proyecto se las detalla a continuación:

Tabla 6.

Herramientas utilizadas para el desarrollo de SEBS.

HERRAMIENTA	DESCRIPCIÓN
Netbeans	Entorno de desarrollo de código abierto para aplicaciones Java, PHP, C++, HTML5, Javascript, entre otros.
MySQL	Sistema de gestión de base de datos de código abierto.
JSF	Java Server Faces. Framework para el desarrollo de aplicaciones web utilizando Java EE.
Primefaces	Librería de elementos web visuales e interactivos para JSF.
Apache TomEE	Servidor de aplicaciones basado en Apache Tomcat, pero compatible con la especificación Java EE 6.
Glassfish	Servidor de aplicaciones de código abierto desarrollado por Sun Microsystems y posteriormente adquirido por Oracle. Compatible con el estándar Java EE.
Ubuntu	Sistema operativo de código abierto basado en Linux y Debian. Desarrollado por la empresa Canonical.

Tabla 7.
 Librerías necesarias para el desarrollo de SEBS.

LIBRERÍAS	DESCRIPCIÓN
MitYCLiBXades	Librería de código abierto para la firma y validación de documentos electrónicos desarrollada por el Ministerio de Industria, Turismo y Comercio de España.
XStream	Librería de código abierto para la generación de archivos XML.
iText	Librería de código abierto bajo la Licencia Pública Menor GNU (LGPL) que permite generar archivos PDF.
JUnit	Librería para el testeo de código. Utilizada comúnmente en pruebas de caja blanca y negra.
MySQL JDBC	Conector para la comunicación de aplicaciones Java con el DBMS MySQL.
Java Mail	Librería para el envío de correos electrónicos a través de la implementación del protocolo Simple Mail Transport Protocol (SMTP).
Apache POI	Librería de código abierto para la lectura de archivos Excel .XLS y .CSV.
Apache Maven	Librería para el manejo de dependencias de aplicaciones en Java.

4.1.1 DIAGRAMA FÍSICO DE BASE DE DATOS

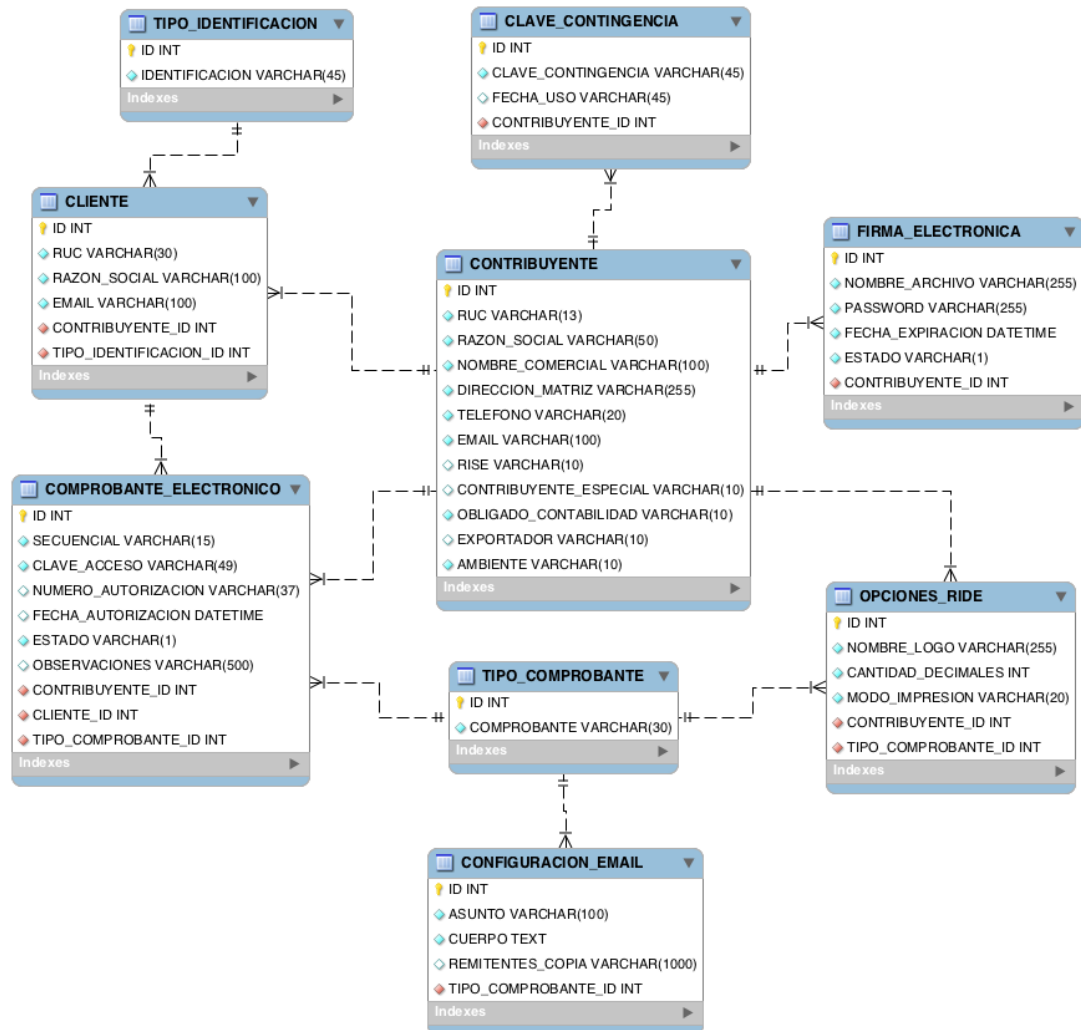


Figura 46. Diagrama físico de la base de datos de SEBS.

4.1.2 SCRIPTS, PAQUETES Y MÉTODOS

a. SCRIPT DE CREACIÓN DE LA BASE DE DATOS

El script de creación de la base de datos se adjunto en el Anexo 4.

b. CRUD DE COMPROBANTES ELECTRÓNICOS

A continuación se muestran sentencias SQL utilizadas en el ORM de SEBS.

Obtener todos los comprobantes electrónicos por empresa

```
SELECT * FROM COMPROBANTE_ELECTRONICO WHERE
CONTRIBUYENTE_ID = ¿ AND ACTIVO = 'S' AND TIPO_COMPROBANTE =
¿ AND CREATED_AT BETWEEN ¿ AND DATE_ADD(¿, INTERVAL 1 DAY)
ORDER BY ID DESC;
```

Actualizar comprobante electrónico

```
UPDATE COMPROBANTE_ELECTRONICO SET FECHA_AUTORIZACION =
¿, ESTADO = ¿, NUMERO_AUTORIZACION = ¿ WHERE CLAVE_ACCESO = ¿
AND ACTIVO = 'S';
```

Insertar un nuevo comprobante electrónico en la base de datos

```
INSERT INTO COMPROBANTE_ELECTRONICO (TIPO_COMPROBANTE,
SECUENCIAL, CLAVE_ACCESO, ESTADO, NUMERO_AUTORIZACION,
FECHA_AUTORIZACION, OBSERVACIONES, CREATED_AT, CLIENTE_ID,
CONTRIBUYENTE_ID) VALUES (¿, ¿, ¿, ¿, ¿, ¿, NOW(), ¿, ¿, ¿);
```

Eliminar comprobante electrónico (Eliminación lógica)

```
UPDATE COMPROBANTE_ELECTRONICO SET ACTIVO = 'N' WHERE
CLAVE_ACCESO = ¿ AND ACTIVO = 'S';
```

c. DESCRIPCIÓN DE PAQUETES

Tabla 8.

Clases de paquete model.

PAQUETE	ec.com.sofyasa.sebs.model
DESCRIPCIÓN	Contiene los modelos de las entidades
CLASES	Contribuyente.java FirmaElectronica.java Cliente.java ComprobanteElectronico.java Iva.java Ice.java

Tabla 9.

Paquete de los DAO.

PAQUETE	ec.com.sofyasa.sebs.dao
DESCRIPCIÓN	Contiene los accesos a objetos (Data Access Object) correspondientes a las entidades.
CLASES	ContribuyenteDao.java FirmaElectronicaDao.java ClienteDao.java ComprobanteElectronicoDao.java IvaDao.java IceDao.java

Tabla 10.

Paquete de emisor de comprobante.

PAQUETE	ec.com.sofyasa.sebs.emisor
DESCRIPCIÓN	Clases que permiten la emisión de los comprobantes
CLASES	EmisorComprobante.java
MÉTODOS	emitirComprobante()

Tabla 11.

Paquete con las clases de autorización de comprobantes.

PAQUETE	ec.com.sofyasa.sebs.autorizador
DESCRIPCIÓN	Clases que permiten la emisión de los comprobantes
CLASES	AutorizadorComprobante.java

Tabla 12.

Paquete con las clases clientes de los servicios web.

PAQUETE	ec.com.sofyasa.sebs.wsclient
DESCRIPCIÓN	Paquete con el cliente de los servicios web del SRI.
CLASES	EnvioComprobantesWs.java AutorizacionComprobantesWs.java WSClient.java WSUtil.java

Tabla 13.

Paquete con las clases de firma electrónica.

PAQUETE	ec.com.sofyasa.sebs.signer
DESCRIPCIÓN	Paquete con las clases e implementaciones de la librería de firmado electrónico.
CLASES	CertificadosSSL.java GenericXMLSignature.java PassStoreKS.java Signer.java

Tabla 14.

Paquete con los generadores de PDF.

PAQUETE	ec.com.sofyasa.sebs.pdf
DESCRIPCIÓN	Clases que contiene las clases para la generación de los formatos RIDE en PDF.
CLASES	FacturaGenerator.java FacturaExportacionGenerator.java GenericDocumentGenerator.java GuiaRemisionGenerator.java NotaCreditoGenerator.java NotaDebitoGenerator.java RetencionGenerator.java

Tabla 15.

Paquete con las clases de envío de email.

PAQUETE	ec.com.sofyasa.sebs.email
DESCRIPCIÓN	Clases que contiene las clases para la generación de los formatos RIDE en PDF.
CLASES	DocumentMailer.java

Tabla 16.

Paquete de generación de XML.

PAQUETE	ec.com.sofyasa.sebs.xml
DESCRIPCIÓN	Clases para la generación de archivos XML.
CLASES	FacturaXml.java FacturaExportacionXml.java NotaCreditoXml.java NotaDebitoXml.java RetencionXml.java GuiaRemisionXml.java

4.2 TRANSICIÓN

La metodología UWE no proporciona una guía de los entregables para esta etapa del ciclo de vida de desarrollo, por lo que en base a las necesidades de la empresa SOFYA S.A. se generaron los siguientes artefactos:

1. Casos de prueba de caja negra.
2. Manual de usuario.
3. Carta de aceptación.

4.2.1. CASOS DE PRUEBA

Para la etapa de pruebas, se aplicaron pruebas de caja negra apoyándolas en clases de equivalencia, para la validación y verificación del software, siguiendo un formato utilizado por SOFYA S.A., el cual se muestra a continuación:

a. PRUEBA 001 – Autenticar usuario

Clases de equivalencia

Tabla 17.

Clases de equivalencia de Autenticar Usuario.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Nombre de usuario	Texto alfanumérico	1. Entre 1 y 100caracteres.	2. En blanco. 3. Más de 10 caracteres.
Contraseña	Texto alfanumérico	4. Cualquier texto alfanumérico.	5. En blanco.

Casos de prueba

Tabla 18.

Casos de prueba de Autenticar Usuario.

Caso N.	Clases de equivalencia	Nombre de usuario	Contraseña	Resultado
1	1, 4	“matus”	“password”	Ingreso exitoso al sistema.
2	2, 4		“password”	ERROR: Usuario/Contraseña inválido

b. PRUEBA 002 – Subir comprobante

Para el caso de prueba de subir comprobante la clase de equivalencia varía un poco debido a que no se prueban campos independientes, sino su conjunto como tal. Estos campos son los requeridos por los formatos de los comprobantes electrónicos, así como su formato.

Clases de equivalencia

Tabla 19.

Clases de equivalencia de Subir Comprobante.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Archivo Excel	Valor	1. Extensión .xls	2. Extensión diferente a .xls
Cédula	Valor	3. Número de cédula válido.	4. Cualquier otro valor.
Precio	Valor	5. Número decimal o entero.	6. Cualquier otro valor.
Archivo Excel	Valor	7. Archivo con formato de factura local.	8. Archivo con formato de factura para exportadores. 9. Archivo con formato inválido

Casos de prueba

Tabla 20.

Casos de prueba de Subir Comprobante.

Caso N.	Clases de equivalencia	Condición de entrada	Resultado
1	1	FACTURA_001.xls	Comprobante autorizado.
2	2	FACTURA_002.xlsx	Tipo de archivo inválido. Invalid file typeFACTURA_002.xlsx 15.8 KB
3	3	1717379157	Comprobante generado exitosamente.
4	4	171739159	Cédula inválida. Comprobante no generado.
5	6	0 (Cero)	Comprobante autorizado.
6	8, 9	FACTURA_EXP_01.xls	Error: Archivo y/o campos inválidos.

c. PRUEBA 003 – Emitir comprobante no procesado

Clases de equivalencia

Tabla 21.

Clases de equivalencia de Emitir comprobante no procesado.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Archivo XML	Valor	1. Comprobante no procesado	2. Comprobante emitido con anterioridad.

Casos de prueba

Tabla 22.

Caso de uso de emisión de comprobante no procesado.

Caso N.	Clases de equivalencia	Archivo Excel	Resultado
1	1	FACTURA_001.xls	Comprobante autorizado.
2	2	FACTURA_002.xlsx	CLAVE DE ACCESO YA REGISTRADA.

d. PRUEBA 004 – Autorizar comprobante pendiente

Clases de equivalencia

Tabla 23.

Clases de equivalencia de autorización de comprobante pendiente.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Clave de acceso	Valor	1. Clave de acceso de comprobante pendiente.	2. Clave de acceso de comprobante rechazado. 3. Clave de acceso de comprobante ya autorizado.

Casos de prueba

Tabla 24.

Caso de uso de autorización de comprobante pendiente.

Caso N.	Clases de equivalencia	Clave de acceso	Resultado
1	1	FACTURA_001.xls	Comprobante autorizado.
2	2	FACTURA_002.xlsx	Comprobante devuelto.

e. PRUEBA 005 – Buscar comprobantes

Clases de equivalencia

Tabla 25.

Clases de equivalencia de búsqueda de comprobantes.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Tipo de comprobante	Conjunto de datos	1. "FACTURA" 2. "NOTA CREDITO" 3. "NOTA DEBITO" 4. "COMPROBANTE RETENCION" 5. "GUIA REMISION"	6. Datos diferentes a los expuestos
Fecha inicial	Valor	7. Fecha en formato dd/mm/aaaa.	8. Fecha o dato con formato diferente.
Fecha final	Valor	9. Fecha en formato dd/mm/aaaa.	10. Fecha o dato con formato diferente.

Casos de prueba

Tabla 26.

Casos de uso de búsqueda de comprobantes,

Caso N.	Clases de equivalencia	Tipo de comprobante	Fecha inicial	Fecha final	Resultado
1	1, 7, 9	"FACTURA"	20/05/2016	20/05/2016	Lista de comprobantes.
2	1	"FACTURA"	"enero"	"febrero"	No records found.

f. PRUEBA 006 – Descargar RIDE comprobante

Clases de equivalencia

Tabla 27.

Clases de equivalencia de descarga de formato RIDE.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Clave de acceso	Valor	1. Clave de acceso de un comprobante autorizado.	2. Clave de acceso de un comprobante no autorizado. 3. Clave de acceso de un comprobante pendiente. 4. Clave de acceso de un comprobante no procesado. 5. Clave de acceso inválida.

Casos de prueba

Tabla 28.

Casos de prueba de descarga de formato RIDE.

Caso N.	Clases de equivalencia	Clave de acceso	Resultado
1	1	“0705201601171596036300110010040000000886578599717”	Descarga exitosa del formato RIDE.
2	2	“2810201501171596036300110010040000000718194678114”	El comprobante debe estar autorizado para descargar.

g. PRUEBA 007 – Descargar XML comprobante

Clases de equivalencia

Tabla 29.

Clases de equivalencia de descarga de formato XML.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Clave de acceso	Valor	1. Clave de acceso de un comprobante.	2. Clave de acceso inválida. 3. El archivo XML no existe.

Casos de prueba

Tabla 30.

Casos de prueba de descarga de formato XML.

Caso N.	Clases de equivalencia	Clave de acceso	Resultado
1	1	“0705201601171596036300110010040000000886578599717”	Descarga exitosa del archivo XML.
2	2, 3	“2510201501171596036300110010050000000718194678114”	No se genera la descarga.

h. PRUEBA 008 – Descargar ZIP de comprobantes

Clases de equivalencia

Tabla 31.

Clases de equivalencia de descarga de archivo ZIP.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Lista de comprobantes	Conjunto de datos	1. Conjunto de claves de acceso de comprobantes.	2. Claves de acceso de comprobantes inválidos.

Casos de prueba

Tabla 32.

Casos de uso de descarga de archivo ZIP.

Caso N.	Clases de equivalencia	Lista de comprobantes	Resultado
1	1	Varios	Descarga de ZIP exitosa.
2	2	Varios	Se genera el archivo de la descarga.

i. PRUEBA 009 – Personalizar correo

Clases de equivalencia

Tabla 33.

Clases de equivalencia de personalización de correos electrónicos.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Tipo de comprobante	Conjunto de datos	1. "FACTURA" 2. "NOTA CREDITO" 3. "NOTA DEBITO" 4. "COMPROBANTE RETENCION" 5. "GUIA REMISION"	6. Datos diferentes a los expuestos
Asunto	Valor	7. Cualquier valor alfanumérico	8. Texto mayor a 255 caracteres.
Cuerpo	Valor	9. Cualquier valor alfanumérico	10. Texto mayor a 10.000 caracteres.
Remitentes	Valor	11. Correos electrónicos separados por comas.	12. Texto mayor a 1000 caracteres.

Casos de prueba

Tabla 34.

Casos de uso de personalización de correos electrónicos.

Caso N.	Clases de equivalencia	Tipo de comprobante	Asunto	Cuerpo	Resultado
1	1, 7, 9, 11	"FACTURA"	"Factura electrónica"	"Su factura electrónica ha sido emitida"	Datos almacenados en la base de datos.
2	1, 8, 11	"FACTURA"	"Factura electrónica..."	"Su factura electrónica ha sido emitida"	Ocurrió un error al procesar el comprobante.

j. PRUEBA 009 – Personalizar RIDE

Clases de equivalencia

Tabla 35.

Clases de equivalencia de personalización de RIDE.

Condición de entrada	Tipo	Clase de equivalencia válida	Clase de equivalencia inválida
Modo de impresión	Conjunto de datos	1. "NORMAL" 2. "ECONÓMICO"	3. Datos diferentes a los expuestos 4. Dato en blanco.
Decimales	Rango de datos	5. Un entero entre 2 y 6.	6. Un número < 2 y > 6 7. Dato en blanco.

Casos de prueba

Tabla 36.

Casos de uso de personalización de RIDE.

Caso N.	Clases de equivalencia	Modo impresión	Decimales	Resultado
1	1, 3	"NORMAL"	7	Datos almacenados en la base de datos.
2	2,	"ECONÓMICO"	1	No se puede elegir 1 decimal.

4.2.2. MANUAL DE USUARIO

Como soporte para el uso del sistema SEBS se consideró la realización del manual de usuario en formato PDF que contiene lo siguiente:

- a. Ingreso al sistema.
- b. Autenticación del contribuyente.
- c. Subir comprobantes.
- d. Comprobantes emitidos y descargas.
- e. Descarga de ZIP.
- f. Configurar correos electrónicos.
- g. Configurar RIDE.

h. Salir del sistema.

Este manual de usuario se encuentra en el Anexo 5.

4.2.3. CARTA DE ACEPTACIÓN

SOFYA S.A., como empresa auspiciante del proyecto desarrollado, se hace cargo de proporcionar una carta de aceptación del sistema una vez puesto en producción, garantizando que SOFYA Electronic Billing System – SEBS satisface con las necesidades iniciales expuestas por la institución. La carta de aceptación se proporciona en el Anexo 6.

CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- El proceso de investigación de la normativa legal del SRI tuvo varias dificultades al momento de obtener información vital para la aplicación debido a que en algunos casos el SRI no dio dicha información de manera clara y lo hacía por medio de su página web en donde no se la actualiza de manera constante, adicionalmente no existe información de casos particulares que se han dado en el desarrollo de la aplicación.
- En el desarrollo de la aplicación, a diferencia de software desarrollado para terceros, SOFYA S.A. es el cliente y desarrollador de la misma aplicación, lo que involucra un trabajo de levantamiento de requerimientos un poco diferente. UWE proporciona una herramienta muy potente para definir los flujos de los procesos de manera gráfica, pero enflaquece en cuanto a detalle se refiere, por lo que se necesitó apoyar la especificación de requisitos con un modelo basado en la IEEE 830.
- La metodología UWE proporciona mecanismos que ayudan a diseñar el software de manera ágil, concentrándose en la parte funcional del mismo como, por ejemplo, el diseño de interfaces y navegabilidad de la aplicación, sin embargo, la metodología es muy dependiente de herramientas que incluyan su notación, como es el caso de Magic Draw. Las herramientas case tradicionales que no integran esta notación también son útiles para seguir los lineamientos de la metodología, a costa de sacrificar tiempo de desarrollo integrándola con herramientas de terceros.
- Debido a los problemas de conectividad con los servicios web del SRI, SOFYA S.A. tuvo solventar sobre la marcha del proceso de desarrollo el control de estos errores, como: errores de tiempo de respuesta, de disponibilidad de servicios, reinicio de conexión, etc.
- La metodología UWE, si bien es una metodología robusta y ágil, no proporciona mecanismos y entregables más allá de los necesarios para el diseño, por lo que carece, de manera formal, de un plan de pruebas de software. SOFYA S.A. se vio

en la necesidad de validar y verificar el software, por lo que se optó utilizar pruebas de caja negra para cumplir con este cometido.

- Para validar el marco de trabajo de software como servicio se necesitó desplegar la aplicación de manera pública en Internet. Se desplegó la aplicación exitosamente utilizando un servidor local expuesto en Internet, así como en la nube privada de Amazon Web Services, lo que garantiza la disponibilidad de la aplicación como software como servicio (SaaS) independientemente de la plataforma utilizada.

5.1 RECOMENDACIONES

- Revisar periódicamente las actualizaciones a las fichas técnicas en la página del SRI y estar al tanto mensualmente de los cambios a la normativa legal ya que esto incide en el comportamiento de la aplicación.
- Complementar la metodología UWE con entregables de otras metodologías basadas en RUP debido a la afinidad y similitud entre las mismas, sobre todo en las etapas de construcción y transición de la metodología.
- Utilizar herramientas case que integren la notación de la metodología UWE para acelerar el proceso de diseño como Magic Draw y reducir tiempos de desarrollo.
- Sugerir formalmente al SRI mejorar la comunicación y retroalimentación con las empresas desarrolladoras, para mejorar la captura de errores de conexión y disponibilidad de los servicios.
- Realizar planes y casos de prueba de acuerdo a la naturaleza de la aplicación y a las guías definidas por la empresa desarrolladora.
- Desplegar la aplicación en servidores de aplicaciones compatibles con la especificación JEE6 para reducir los problemas de compatibilidad.

BIBLIOGRAFÍA

- Congreso Nacional. (17 de abril de 2002). LEY DE COMERCIO ELECTRÓNICO, FIRMAS ELECTRÓNICAS Y MENSAJES DE DATOS . Quito, Pichincha, Ecuador.
- Congreso Nacional. (2002). *Ley de comercio electrónico, firmas electrónicas y mensajes de datos* .
- Director General del Servicio de Rentas Internas. (12 de marzo de 2014). Resolución NAC-DGERCGC14-00157. Ecuador.
- Directora General del Servicio de Rentas Internas. (27 de mayo de 2014). Resolución NAC-DGERCGC13-00236. Ecuador.
- Koch, N., Kraus, A., & Hennicker, R. (Junio de 2001). The Authoring Process of the UML-based Web Engineering Approach.
- Kraus, A., Knapp, A., & Koch, N. (2007). Model-Driven Generation of Web Applications in UWE. *Model-Driven Generation of Web Applications in UWE* . MDWE.
- Kroiß, C., & Koch, N. (Febrero de 2008). UWE Metamodel and Profile. Múnich, Baviera, Alemania.
- Leong, L., Toombs, D., & Gill, B. (18 de Mayo de 2015). *Magic Quadrant for Cloud Infrastructure as a Service, Worldwide* .
- Ludwig-Maximilians-Universität, R. U. (30 de Enero de 2006). *UWE – UML-based Web Engineering*. Retrieved 07 de Febrero de 2015 from <http://uwe.pst.ifi.lmu.de/index.html>
- Ministerio de Coordinación de Seguridad. (30 de Octubre de 2015). Registro Oficial N. 619. *Resolución NAC-DGERCGC15-00000745-B Registro* .
- Nieves Guerrero, C., Ucán Pech, J., & Menéndez Domínguez, V. (2014). UWE en Sistema de Recomendación de Objetos de Aprendizaje. Aplicando Ingeniería Web: Un Método en Caso de Estudio. *Revista Latinoamericana de Ingeniería de Software* , 137-143.
- Noboa Bejarano, G. (31 de diciembre de 2002). Reglamento General a la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos. Quito, Pichincha, Ecuador.

- Pressman, R. S. (2005). *Ingeniería del Software*. Mexico DF: Mc Graw Hill.
- Servicio de Rentas Internas. (09 de marzo de 2002). Resolución No. NAC-DGERCGC12-00105. Ecuador.
- Servicio de Rentas Internas. (2014 de septiembre de 2014). Resolución NAC-DGERCGC13-00236. Ecuador.
- Servicio de Rentas Internas. (2015). FICHA TÉCNICA. *Manual de usuario, catálogo y especificaciones técnicas, "Emisión de Comprobantes Electrónicos", Método de autorización online* .
- Servicio de Rentas Internas. (21 de marzo de 2012). Resolución NAC-DGERCGC15-00000004 . Ecuador.
- Servicio de Rentas Internas. (30 de mayo de 2014). Resolución NAC-DGRCGC14-00788. Ecuador.
- Servicio de Rentas Internas. (9 de octubre de 2014). Resolución NAC-DGRCGC14-00788. Ecuador.
- The Institute of Electrical and Electronics Engineers, I. (2 de Diciembre de 1998). IEEE Standard for Developing Software Life Cycle Processes. New York, Estados Unidos.