



Facultad 4

TÍTULO:

“Análisis y Diseño de una aplicación móvil para la interacción con el Repositorio de Objetos de Aprendizaje RHODA”

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas.

AUTOR:

Yanelis Batista Reyes.

TUTORES:

Ing. Leonardo Rodríguez González.

Ing. Yandris Mata Cabrera.

Ciudad de La Habana, junio de 2014

Declaración de autoría

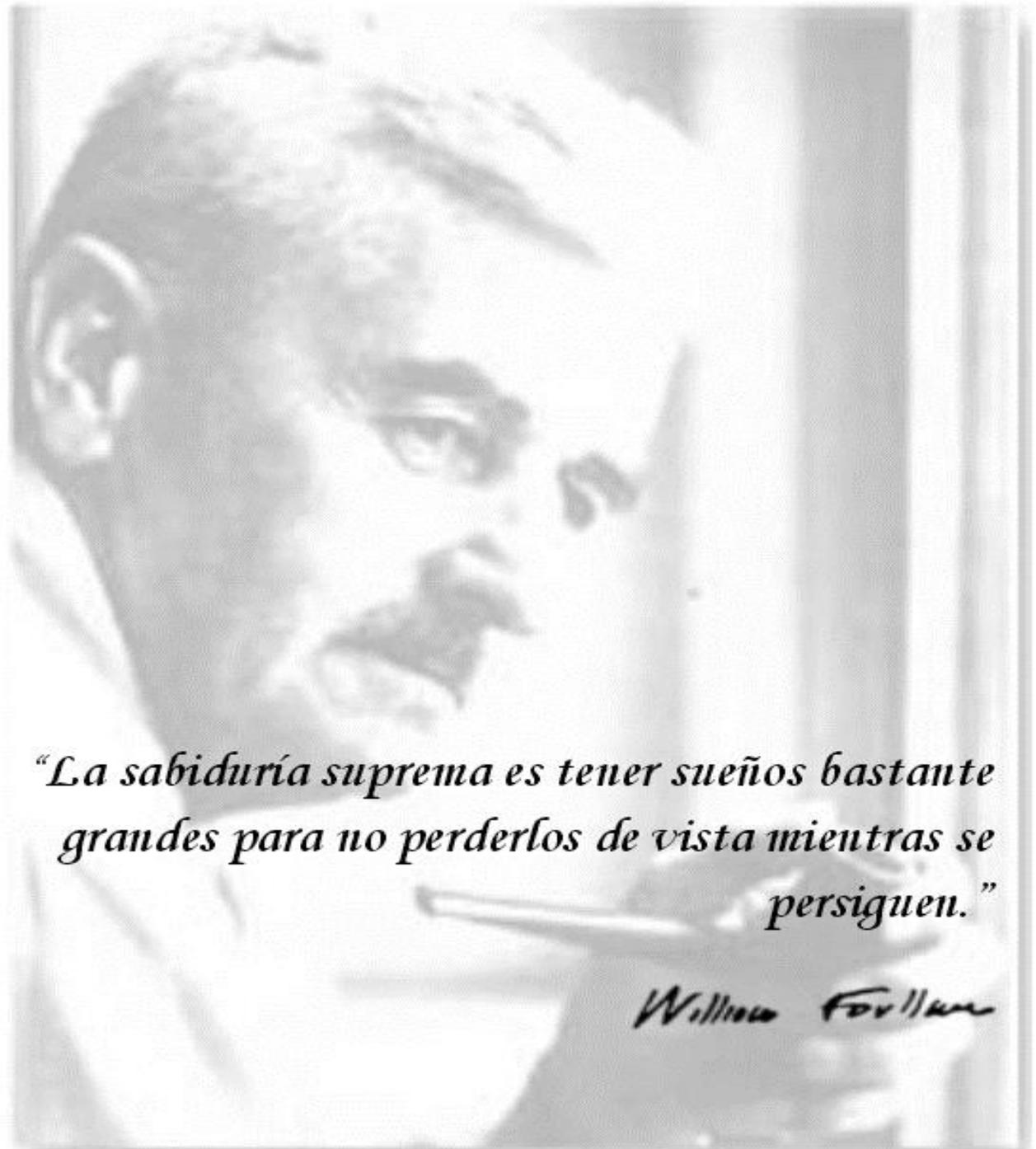
Por este medio declaro que soy único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas (UCI) para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo el presente a los _____ días del mes de ____ del año ____.

Autor: Yanelis Batista Reyes

Tutor: Ing. Leonardo Rodríguez González

Tutor: Ing. Yandris Mata Cabrera



“La sabiduría suprema es tener sueños bastante grandes para no perderlos de vista mientras se persiguen.”

Willis Follmer

Dedicatoria y agradecimientos

Primero a quienes les debo todo lo que soy, los que forjaron los cimientos de la persona en la que me he convertido, los que me darían su vida si se las pidiese; a mis padres Cecilia y Juan Carlos, a ustedes les dedico mi tesis.

Segundo a mi novio, por tanta entrega y amor, por estar siempre a mi lado en la buena y en la mala; a ti Sol, muchas gracias.

Tercero a mi tía por haberme apoyado en los momentos difíciles, siendo una segunda madre para mí, a ti Paula, muchas gracias.

Cuarto a los que me orientaron y enseñaron, a esos profesores que tantos consejos me dieron, a los que fueron mi guía y camino a seguir; a Basulto y Maritza, a ustedes, muchas gracias.

Quinto a mis tutores Yandris y Leonardo, por haberme aguantado, cuanta paciencia y comprensión, muchas gracias y a la vez mis disculpas por los momentos malos que han pasado por mis errores.

Sexto a Osvaldo y Pepe, por haberme guiado en el desarrollo de la tesis, sin su ayuda no hubiera sido posible, a ustedes, muchas gracias.

Por último, a mis amigos (Emilio, Celia, Yailín y Gleydis), mis hermanos de la Universidad (Jordamis (duro), Yuliet (la loka), Ever (empella), Lester (muñecón), Ernesto (mongui)), los que han estado a mi lado desde el primer día, también a los que no pudieron llegar pero que siempre lo intentaron, a los de las fiestas y locuras, a ustedes, muchas gracias.

Resumen

El avance de las tecnologías ha tenido gran impacto en la esfera de la educación. Como ejemplo se tienen los dispositivos móviles, los cuales han expandido su uso en esta esfera. La utilización de estos medios en el proceso de enseñanza-aprendizaje ha permitido que florezca el m-learning.

En la actualidad el Repositorio de Objetos de Aprendizaje RHODA no brinda soporte a la estrategia m-learning, puesto que no fue diseñado para que fuese accedido desde este tipo de tecnología, motivo fundamental de que se iniciase el proceso de análisis y diseño de una aplicación para dispositivos móviles, que permita en el futuro la interacción de los usuarios con el repositorio RHODA.

Para la ejecución de este proceso ingenieril se hizo necesario seleccionar y describir un conjunto de herramientas y tecnologías, que sirven de soporte desde el punto de vista arquitectónico y como resultado del mismo se obtuvo un conjunto de productos de trabajo, que permiten sentar las bases para una futura implementación de una aplicación para dispositivos móviles, que utilice el sistema operativo Android.

Palabras clave: educación, dispositivos móviles, m-learning, repositorio.

Tabla de contenido

Introducción	1
Capítulo 1: Fundamentación teórica	4
1.1 Aprendizaje móvil.....	4
1.2 Objeto de aprendizaje	5
1.3 Repositorio de Objetos de Aprendizaje	6
1.4 Tipos de variantes para lograr una presencia móvil	8
1.5 Dispositivos móviles.....	9
1.6 Soluciones similares	13
1.7 Herramientas y tecnologías para el sistema operativo Android	14
1.8 Metodología de desarrollo del software.....	17
1.9 Herramienta CASE.....	19
Capítulo 2 Características del sistema	21
2.1 Modelo del dominio	21
2.2. Requerimientos del sistema	23
2.3. Personas relacionadas con el sistema	25
2.4 Patrones de casos de uso.....	26
2.5 Diagrama de casos de uso identificados	27
2.6 Listado de casos de uso	28
2.7 Diagrama de estructura de navegación.....	34
Capítulo 3: Análisis y Diseño del sistema.....	36
3.1 Modelo del análisis	36
3.2 Diagrama de interacción	38
3.3 Patrones arquitectónicos.....	42
3.4 Patrones de diseño	44

Índice de contenidos

3.5 Modelo del diseño	45
3.6 Diagrama de componentes	46
3.7 Revisión de la solución propuesta.....	47
Conclusiones generales.....	55
Recomendaciones	56
Referencias bibliográficas	57
Glosario de términos.....	62
Anexo 1. Descripción de los casos de uso.....	65
Anexo 2 Diagramas de clases del análisis	93
Anexo 3 Diagramas de colaboración.....	98
Anexo 4: Diagrama de secuencia	104
Anexo 5: Resultado de la validación de los requerimientos.....	109
Anexo 6: Resultado de la validación de los casos de uso	114

Índice de figuras

Figura 1-Definición de OA.....	6
Figura 2-Fases e Iteraciones de la Metodología RUP.....	19
Figura 3-Modelo del dominio.....	22
Figura 4-Diagrama de actores del sistema.....	26
Figura 5-CRUD Parcial.....	27
Figura 6-Múltiples actores (roles comunes).....	27
Figura 7-Diagrama de casos de uso.....	28
Figura 8-Diagrama de estructura de navegación.....	35
Figura 9-Diagrama de colaboración "Listar OA (más solicitados)".....	37
Figura 10-Diagrama de clases del análisis "Servicio de mensajería (Bandeja de entrada)".....	38
Figura 11-Diagrama de clases del análisis "Gestionar perfil del usuario".....	38
Figura 12-Diagrama de colaboración "Listar OA (más solicitados)".....	39
Figura 13-Diagrama de secuencia "Listar OA (más solicitados)".....	40
Figura 14-Diagrama de colaboración "Servicio de mensajería (Bandeja de entrada)".....	41
Figura 15-Diagrama de secuencia "Servicio de mensajería (Bandeja de entrada)".....	41
Figura 16-Diagrama de colaboración "Gestionar perfil del usuario".....	41
Figura 17-Diagrama de secuencia "Gestionar perfil del usuario".....	42
Figura 18-Arquitectura de la aplicación.....	43
Figura 19-Diagrama de diseño.....	46
Figura 20-Diagrama de componentes.....	47
Figura 21-Interfaz externa de la aplicación.....	92
Figura 22-Diagrama de clases del análisis "Recuperar contraseña".....	93
Figura 23-Diagrama de clases del análisis "Mostrar portada".....	93

Figura 24-Diagrama de clases del análisis “Listar OA (en revisión)”	93
Figura 25-Diagrama de clases del análisis “Listar OA (por revisar)”	94
Figura 26-Diagrama de clases del análisis “Ver usuarios conectados”	94
Figura 27-Diagrama de clases del análisis “Actualizar contenido”	94
Figura 28-Diagrama de clases del análisis “Servicio de mensajería (Crear mensaje)”	95
Figura 29-Diagrama de clases del análisis “Servicio de mensajería (Mensajes enviados)”	95
Figura 30-Diagrama de clases del análisis (Registrar usuario)	95
Figura 31-Diagrama de clases del análisis “Listar OA (por categorías)”	96
Figura 32-Diagrama de clases del análisis “Autenticar usuario”	97
Figura 33-Diagrama de clases del análisis “Listar OA (recientes)”	97
Figura 34-Diagrama de clases del análisis “Gestionar OA”	98
Figura 35-Diagrama de colaboración “Recuperar contraseña”	98
Figura 36-Diagrama de colaboración “Mostrar portada”	98
Figura 37-Diagrama de colaboración “Listar OA (en revisión)”	99
Figura 38-Diagrama de colaboración “Listar OA (por revisión)”	99
Figura 39-Diagrama de colaboración “Ver usuarios conectados”	100
Figura 40-Diagrama de colaboración “Actualizar contenido”	100
Figura 41-Diagrama de colaboración “Servicio de mensajería (crear mensaje)”	100
Figura 42-Diagrama de colaboración “Servicios de mensajería (mensajes enviados)”	101
Figura 43-Diagrama de colaboración “Registrar usuario”	101
Figura 44-Diagrama de colaboración “Autenticar usuario”	101
Figura 45-Diagrama de colaboración “Listar OA (por categorías)”	102
Figura 46-Diagrama de colaboración “Listar OA (recientes)”	103
Figura 47-Diagrama de colaboración “Gestionar OA”	103
Figura 48-Diagrama de secuencia “Recuperar contraseña”	104

Índice de figuras

Figura 49-Diagrama de secuencia “Mostrar portada”	104
Figura 50-Diagrama de secuencia “Lista OA (en revisión)”	104
Figura 51-Diagrama de secuencia “Lista OA (por revisar)”	105
Figura 52-Diagrama de secuencia “Ver usuarios conectados”	105
Figura 53-Diagrama de secuencia “Actualizar contenido”	105
Figura 54-Diagrama de secuencia “Autenticar usuario”	106
Figura 55-Diagrama de secuencia “Servicio de mensajería (Crear mensaje)”	106
Figura 56-Diagrama de secuencia “Servicio de mensajería (Mensajes enviado)”	106
Figura 57-Diagrama de secuencia “Registrar usuario”	107
Figura 58-Diagrama de secuencia “Listar OA (recientes)”	107
Figura 59-Diagrama de secuencia “Listar OA (por categoría)”	108
Figura 60-Diagrama de secuencia “Gestionar OA”	109

Índice de tablas

Tabla 1-Comparación de las metodologías tradicionales y ágiles (34).....	17
Tabla 2-Personas relacionadas con el sistema.	25
Tabla 3-Caso de Uso “Autenticar Usuario”.....	28
Tabla 4-Caso de Uso “Recuperar contraseña”.....	29
Tabla 5-Caso de Uso “Servicio de Mensajería”.....	29
Tabla 6-Caso de Uso “Listar publicaciones de OA”.....	30
Tabla 7-Caso de Uso “Listar OA en revisión”.....	31
Tabla 8-Caso de Uso “Listar OA por revisar”.....	31
Tabla 9-Caso de Uso “Ver usuarios conectados”.....	31
Tabla 10-Caso de Uso "Mostrar la portada".....	32
Tabla 11-Caso de Uso "Actualizar contenido".....	32
Tabla 12-Caso de uso "Gestionar perfil".....	33
Tabla 13-Caso de uso "Gestionar OA".....	33
Tabla 14-Caso de uso “Registrar usuario”.....	34
Tabla 15-Estereotipos de las clases del análisis.....	36
Tabla 16-Clasificación de la estabilidad.....	52
Tabla 17: Resultado de la compleción.....	110
Tabla 18: Resultado de la corrección.....	110
Tabla 19: Resultado de la comprensión.....	111
Tabla 20: Resultado de la consistencia interna.....	112
Tabla 21: Resultado de la consistencia externa.....	113
Tabla 22: Resultado de la estabilidad.....	113
Tabla 23. Resultado de la completitud.....	114
Tabla 24. Resultado de la comprensibilidad.....	115

Tabla 25. Resultado de la concisión	115
Tabla 26. Resultado de no trivialidad	116

Introducción

Desde su surgimiento los dispositivos móviles se manifestaron como un innegable paso de avance en la esfera de las comunicaciones, acogido su uso por un gran número de usuarios y respaldado su desarrollo por varias empresas. Actualmente los dispositivos móviles continúan incrementando sus potencialidades, llegando incluso estas a diferir un tanto del objetivo principal de creación: la transmisión de voz, pero manteniendo el concepto de servir como medio de comunicación.

Con el paso del tiempo y de los avances de la ciencia los móviles se han convertido en pequeñas computadoras personales permitiendo: descargas de imágenes, vídeos, tonos, principales noticias, guiar a través de un mapa hacia una dirección determinada, etc. El desarrollo de esta tecnología ha tenido un gran impacto en la esfera educacional, dando origen al aprendizaje móvil (m-learning), el cual ha propiciado que el software educativo sea desarrollado de tal forma que se pueda acceder a sus contenidos desde estos dispositivos.

Una de las tendencias actuales es la creación de contenidos formativos en forma de Objetos de Aprendizaje (OA). La gran diversidad y cantidad de OA existentes han devenido en la necesidad de una herramienta que permita su almacenamiento y gestión, dando origen a lo que se conoce como Repositorio de Objetos de Aprendizaje (ROA) con el fin de localizar y reutilizar con eficacia el contenido almacenado.

En el Centro de Tecnologías para la Formación (FORTES) de la Universidad de las Ciencias Informáticas (UCI), se ha estado desarrollando desde el año 2009 un Repositorio de Objetos de Aprendizaje denominado RHODA, el cual brinda funcionalidades desde las más básicas hasta las más complejas para llevar a cabo la gestión de los OA que almacena.

A raíz del éxito que está teniendo el aprendizaje a través de sistemas móviles y considerando la evolución de las redes inalámbricas¹, es fácil entender cómo estas tecnologías han cobrado enorme importancia en la educación (1), permitiendo interactuar con los contenidos en cualquier lugar y cualquier momento.

La necesidad existente de facilitar el acceso a profesores y estudiantes, desde los terminales posibles al repositorio RHODA y de interactuar con él sin necesidad de una computadora, es un factor que afecta hoy el nivel de uso que tiene este sistema. En la actualidad se dificulta el acceso a los contenidos almacenados en RHODA a través de dispositivos móviles debido a que no está adaptado a algunas

¹Red inalámbrica: (*Wireless network* en inglés) es un término que se utiliza en informática para designar la conexión de nodos sin necesidad de una conexión física (cables), esta se da por medio de ondas electromagnéticas.

restricciones que presenta este tipo de tecnología, como el despliegue de información limitado (en algunos casos) y poseen entrada de información deficientes (teclados pequeños en algunos casos).

Para poder realizar la implementación de una aplicación móvil para RHODA por parte del equipo de desarrolladores, se hace necesario definir las características que poseerá la aplicación, modelar la estructura lógica y sus funciones enmarcadas en un contexto pedagógico. Por estas razones es necesario aportarle al equipo la documentación y modelación necesaria que le sirva de guía para realizar la implementación de la aplicación, situación que conlleva al desarrollo de la presente investigación, donde se documenta el proceso de análisis y diseño de una aplicación móvil para RHODA.

La situación problemática definida, induce al siguiente **problema a resolver** ¿Cómo lograr una adecuada entrada al flujo de trabajo de implementación de una aplicación para dispositivos móviles que permita la interacción con RHODA?

El problema descrito genera como **objeto de estudio** el proceso de desarrollo de aplicaciones para dispositivos móviles y como campo de acción el desarrollo de una aplicación para dispositivos móviles que permita interactuar con el Repositorio de Objetos de Aprendizaje RHODA.

Para solucionar el problema planteado, el presente trabajo tiene como **objetivo general** realizar el análisis y diseño de una aplicación que posibilite la interacción con RHODA mediante dispositivos móviles.

En vista de guiar el desarrollo de este trabajo se toma la siguiente **idea a defender**. El análisis y diseño de una aplicación móvil para RHODA permitirá obtener una adecuada entrada al flujo de trabajo de implementación, que facilitará la interacción con RHODA mediante dispositivos móviles.

El objetivo general se desglosa en los siguientes **objetivos específicos**:

- Describir los elementos teóricos que sustentan la investigación a través del estudio del estado del arte.
- Desarrollar los artefactos correspondientes al análisis y diseño de la aplicación móvil para RHODA.
- Realizar las revisiones correspondientes de la propuesta de solución mediante las métricas de especificación de requerimientos y casos de uso.

Los métodos empleados para el desarrollo de esta investigación se basan en la combinación dialéctica² de los métodos teóricos y empíricos. Entre los métodos teóricos empleados se encuentra el **Histórico – Lógico** que permitió el estudio del estado del arte, para investigar acerca de aplicaciones móviles; el **Analítico – Sintético** que permitió identificar los conceptos y definiciones relacionados con las aplicaciones para dispositivos móviles, y permitió generar una propuesta de análisis y diseño adecuada a la situación planteada. **Modelación**, el cual facilitó el esbozo de los diferentes diagramas y modelos del proceso de análisis y diseño de la aplicación. Dentro de los métodos empíricos está el método Observación, que permitió estudiar el objeto de la investigación, las acciones, causas y consecuencias.

El contenido de este documento está estructurado en tres capítulos, conclusiones, recomendaciones, referencias bibliográficas, glosario de términos y anexos.

Capítulo 1: "Fundamentación teórica", se realiza un estudio sobre cómo realizar aplicaciones para dispositivos móviles, se describen los conceptos fundamentales que se van a tratar en la presente investigación, así como el análisis de las tecnologías y herramientas a utilizar en el desarrollo de aplicaciones para dispositivos móviles.

Capítulo 2: "Características del Sistema", se describe la solución que se propone. Primero se plasma el modelo de dominio para comprender el ámbito donde se desarrolla la aplicación. Se crea el flujo de trabajo requerimientos de la metodología RUP, donde se exponen básicamente los requisitos funcionales y no funcionales del sistema, para llegar así a los casos de uso. Se realiza una descripción textual de los mismos y se efectúa un diseño de los prototipos de interfaz de usuario de cada caso de uso identificado.

Capítulo 3: "Análisis y Diseño", se hace referencia a los detalles descriptivos de la solución propuesta a través del análisis y diseño, así como las pruebas de los requerimientos y casos de uso generados.

² *Dialéctica: (del griego διαλεκτική (dialektiké), τέχνη (téchne), literalmente: técnica de la conversación; con igual significado, en latín (ars) dialéctica) es una rama de la filosofía cuyo ámbito y alcance ha variado significativamente a lo largo de la historia.*

Capítulo 1: Fundamentación teórica

Introducción

En el mundo existe una gran demanda de acceso a Internet, esto trae consigo un aumento de solicitudes a las grandes industrias de comunicaciones inalámbricas. Internet se ha hecho indispensable para el ser humano y la necesidad de acceder a la misma ha favorecido al surgimiento de las redes inalámbricas. Los dispositivos permiten estas comunicaciones, que con el uso de algunas de sus técnicas es la que más auge ha tomado.

1.1 Aprendizaje móvil

El dispositivo móvil ha alcanzado un gran auge a nivel mundial debido a las numerosas ventajas que puede ofrecer. Es una tecnología tan utilizada en la actualidad que se ha hecho imprescindible para el ser humano en diferentes sectores, posibilitando una nueva alternativa de enseñanza llamada m-learning.

El m-learning surgió en la década de los 80, cuando floreció el Dynabook³. En el año 1990 siguió desarrollándose en universidades de Europa y Asia, donde se evaluaron las posibilidades de la educación móvil para estudiantes (2). Muchos son los autores que han definido al m-learning, a continuación se citarán algunos de ellos:

Sharples dice que *“...el aprendizaje móvil es un paradigma emergente en un estado de intenso desarrollo impulsado por la confluencia de tres corrientes tecnológicas, poder de cómputo, ambiente de comunicación y el desarrollo de interfaces de usuario inteligente”* (3).

MoLeNet define al aprendizaje móvil como *“la explotación de tecnologías ubicuas de mano, junto con las redes de teléfonos inalámbricos y móviles, para facilitar, apoyar, mejorar y ampliar el alcance de la enseñanza y el aprendizaje”* (3).

Quin señala que *“...el aprendizaje móvil puede ser visto como un subconjunto de e-learning. El e-learning es el concepto que incluye los entornos de aprendizaje móvil y en línea. En este sentido, la simple definición siguiente: m-learning es el e-learning a través de dispositivos móviles de cómputo”* (3).

La definición a tomar como patrón en la presente investigación es: el m-learning significa aprendizaje electrónico móvil o educación móvil. Es como su propio nombre indica, una metodología o difusión de contenidos de enseñanza y aprendizaje a través de dispositivos móviles (4).

³Dynabook: es una computadora del tamaño de un libro, portátil, con red inalámbrica y pantalla plana.

Ventajas del Mobile Learning

Las tecnologías móviles se usaban principalmente para realizar llamadas y enviar mensajes. Hoy en día el teléfono móvil ha evolucionado en su casi tres décadas de existencia, el cual es una herramienta indispensable para obtener y gestionar información oportunamente. Y es que estar conectado a Internet se ha convertido en una necesidad, más aun si hay que referirse a la educación, la cual se beneficia de la tecnología.

Los dispositivos móviles permiten un aprendizaje a cualquier hora y en cualquier lugar. Es más fácil adquirir un móvil que una computadora, esto ayuda a reducir la brecha digital. Debido a que la tecnología móvil está al alcance de casi todas las personas y que el m-learning aumenta el interés del usuario por el uso de las nuevas tecnologías, surge mayor penetración a la educación formal. Facilita la interacción instantánea docente–alumno, aumentando la retroalimentación anónima de temas particulares o generales (5).

Desventajas del Mobile Learning

La tecnología móvil presenta problemas asociados a la usabilidad, ya que en el caso de los teléfonos inteligentes (smartphones) las pantallas son demasiado pequeñas (en algunos casos) y no pueden usarse durante un largo período de tiempo. En el caso de las tabletas, este inconveniente no es tan acusado. Las baterías suelen durar, en uso intensivo, de dos a cuatro horas. Reduce el trabajo colaborativo entre los estudiantes, requiere la disposición total por parte del estudiante para aprender en cualquier lugar sin importar los agentes distractores que se encuentren (5).

Se piensa que los avances de la tecnología pueden hacer que las desventajas asociadas a la usabilidad se maten a través del tiempo (6), por lo que se recomienda el uso del cómputo móvil con otro modelo educativo para formar una estrategia más integral, permitiendo interactuar con los contenidos educativos en cualquier parte y en cualquier momento.

1.2 Objeto de aprendizaje

En la actualidad gran parte de los contenidos formativos son elaborados en forma OA para brindar grandes beneficios, ya que un mismo recurso puede utilizarse en distintos contextos debido a la accesibilidad, interactividad, durabilidad, interoperabilidad, flexibilidad, granularidad y reutilización con que cuentan.

Los criterios relacionados con los OA son muy variados, algunos autores los definen como:

“Entidades digitales distribuidas a través de Internet, con posibilidades de acceso simultáneo, utilizables por los diseñadores para construir pequeñas piezas de componentes instruccionales, reutilizables en diferentes contextos. Estas piezas pueden ser autocontenidas e incluir en su estructura otros objetos o soportar objetivos instruccionales individuales” (7).

Otros se refieren a un OA como “una entidad digital o no digital que puede ser utilizada, reutilizada y referenciada durante el aprendizaje apoyado con tecnología” (8).

Wiley dice que un OA es “cualquier recurso digital que puede ser reutilizado para apoyar el aprendizaje” (9).

Basándose en las definiciones estudiadas y debido a que es de todos los conceptos el que más se ajusta a la línea de la presente investigación, se puede definir un OA como: “cualquier recurso con una intención formativa, compuesto de uno o varios elementos digitales, descrito con metadatos⁴, que pueda ser utilizado y reutilizado dentro de un entorno e-learning” (10).

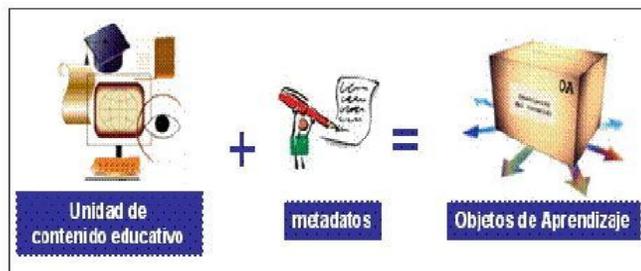


Figura 1-Definición de OA.

1.3 Repositorio de Objetos de Aprendizaje

Es casi intrínseco que se tenga un lugar destinado para el almacenamiento y clasificación de un OA. Este almacén es lo que se conoce como Repositorio de Objetos de Aprendizaje (ROA), que almacena colecciones de OA, que cuentan con una descripción detallada (metadatos) o solamente guarda las descripciones de OA almacenados en otros sitios, para poder acceder a su ubicación (11).

Enfocado al ROA existen una amplia gama de conceptos, algunos autores se refieren a estos como: “un catálogo electrónico/digital que facilita las búsquedas en internet de objetos digitales para el aprendizaje”

⁴Metadatos: (del griegoμετα, meta), literalmente «sobre datos», son datos que describen otros datos.

Capítulo 1 Fundamentación teórica

(12). Otros plantean que “*son bases de datos con búsqueda, que alojan recursos digitales y/o metadatos que pueden ser utilizados para el aprendizaje mediado*” (12).

Jorum Team dice que es “*una colección de OA que tienen información (metadatos) detallada que es accesible vía Internet. Además de alojar los OA, los ROA pueden almacenar las ubicaciones de aquellos objetos almacenados en otros sitios, tanto en línea como en ubicaciones locales*” (12).

Por los conceptos mencionados anteriormente, se puede decir que los ROA son un tipo de bibliotecas digitales especializadas en recursos educativos, que usan los estándares de metadatos que han desarrollado los organismos encargados de la estandarización del e-learning, preparadas tecnológicamente para interoperar con otros repositorios y con otras aplicaciones de los entornos e-learning (12).

En la actualidad existen diferentes tipos de ROA, los cuales pueden ser agrupados en dos ramas: repositorios digitales y los repositorios de acceso abierto. El primero es un sistema para almacenar, conservar y dar acceso a documentos digitales. Los repositorios de acceso abierto recogen, almacenan y dan accesos a resultados (publicaciones y/o documentos) de investigación científica (13). Algunos de los repositorios de acceso abierto son:

Institucionales: Contienen la producción de los miembros de una institución, ya sea una universidad o un centro de investigación. Tienen carácter multidisciplinar. A veces se centran exclusivamente en contenidos científicos (artículos de revista, tesis, congresos), pero existen también ejemplos de inclusión de material docente, documentación administrativa y colecciones patrimoniales (13).

Temáticos: Son contenidos que están especializados en un determinado ámbito científico. Los creadores pueden ser instituciones académicas, organismos públicos u organismos sin ánimo de lucro. Fueron los primeros en crearse (13).

Agregadores o portales: Recogen su contenido en repositorios institucionales y/o acción disciplinaria (13).

Un ROA provee un conjunto de funciones básicas y complejas con el fin de dar acceso a los OA en un ambiente seguro. Según Advanced Distributed Learning (ADL) las funcionalidades básicas son: buscar, que es la habilidad para localizar un OA apropiado; consultar la información; pedir un OA que ha sido localizado; recuperar un OA perdido; almacenar dentro de un registro de datos un objeto con un identificador único que le permita ser localizado y brindar la opción de publicar, es decir, proveer metadatos a otros repositorios (14).

Dentro de las funcionalidades complejas que tiene un ROA están: configuración de fuentes para la recolección de metadatos a través de OAI-PMH, configuración del sistema de autenticación y configuración del sistema de revisión.

En la actualidad la UCI tiene una matrícula superior a dos mil personas entre estudiantes y profesores, por lo que como parte del proceso de enseñanza-aprendizaje se crea un gran cúmulo de OA, de ahí surge el repositorio RHODA para todos los usuarios. Además, se ha desarrollado una personalización del repositorio para el Ministerio del Poder Popular para la Educación Universitaria de Venezuela (MPPEU).

La UCI no está lejana de la nueva forma de acceso a aplicaciones mediante dispositivos móviles, ya existentes en el mundo. Actualmente no se puede acceder a los contenidos de RHODA mediante dispositivos móviles, debido a que no fue desarrollado para ser accesible mediante estas tecnologías. Por todo lo antes mencionado y para dar solución al problema a resolver de la presente investigación, se va a proponer una solución de una aplicación para dispositivos móviles vinculada a RHODA.

1.4 Tipos de variantes para lograr una presencia móvil

En la actualidad existen dos tipos de variantes que se pueden desarrollar para lograr una presencia móvil, los sitios web móviles y las aplicaciones para dispositivos móviles (app). A continuación se caracterizan y se muestran las ventajas y desventajas de cada una de ellas:

Sitios web móviles

Un sitio web móvil es similar a uno normal, sólo que ha sido diseñado y adaptado para poder navegar cómodamente desde un móvil. Se accede a ellos desde las distintas tecnologías de conexión a Internet existentes en los dispositivos móviles (Wifi, 3G⁵, 4G⁶) a través de algún navegador (Opera, Safari, Firefox) instalado previamente (15).

Los sitios web móviles dependen del navegador para interactuar con ellos por lo que es accesible para todos los tipos de dispositivos con penetrabilidad a Internet. Las actualizaciones pueden realizarse en cualquier momento y estar disponibles de inmediato (15).

Un sitio web adaptado debe ajustarse a las medidas de un dispositivo móvil, por lo que será menos atractivo, nunca será tan complejo y dinámico como una aplicación, además, la mayoría de los

⁵ 3G: Primera generación de la telefonía móvil.

⁶ 4G: Cuarta generación de la telefonía móvil.

navegadores móviles son extremadamente simples. Los sitios web dependen directamente de una conexión a Internet para poder ejecutarse (15).

Aplicación para dispositivos móviles

Una aplicación puede utilizar toda la pantalla del dispositivo, debido a que es desarrollada con el software que ofrece cada sistema operativo llamado software SDK (Paquete de Desarrollo de Software, siglas en inglés), proporcionando una mejor interfaz y acceso a algunas características que no están presentes en un sitio web móvil, como el GPS (Sistema de Posicionamiento Global, siglas en inglés), la cámara o el micrófono (16). Aunque no se disponga de Wifi o de 3G, se puede utilizar la aplicación que se ha descargado en el dispositivo, por lo que puede ser ejecutada sin la necesidad de estar conectado a Internet. Siempre será visible en el escritorio del dispositivo móvil, por lo que las oportunidades de que la utilicen son mayores (15).

Una aplicación tiene un desarrollo más costoso que un sitio web móvil, ya que debe ser desarrollada por separado. Para que un usuario pueda utilizar la aplicación, primero tiene que descargarla con la dificultad de que depende directamente del sistema operativo para el que fue diseñada, siendo incompatible con otros. Por ese motivo es que antes de iniciar el proceso de desarrollo, se debe haber realizado un estudio preliminar relacionado con los dispositivos móviles y sistemas operativos para el cual se desea que funcione la aplicación (15).

Debido a que una aplicación móvil tiene más velocidad de funcionamiento que un sitio web móvil, ya que para ser usada no hay necesidad de conexión a Internet. Además, de brindar mejor interfaz de usuario por ser desarrollada por el software del sistema operativo (SO) seleccionado. Se considera que es la adecuada para el desarrollo de la presente investigación.

1.5 Dispositivos móviles

Tener un dispositivo hoy en día puede ser muy útil. El teléfono inteligente, lo que antes se conocía como un teléfono celular, se usaba principalmente para realizar llamadas y enviar mensaje. Hoy puede ser una herramienta indispensable para obtener información oportunamente y gestionar las tareas. Y es que estar conectado a Internet hoy en día se ha convertido en una necesidad en muchos sectores.

Tipos de dispositivos móviles

Dado el variado número de niveles de funcionalidad asociados con los dispositivos móviles, en el 2005, T38 y DuPont Global Mobility Innovation Team propusieron los siguientes estándares para la definición de

dispositivos móviles: Dispositivo Móvil de Datos Limitados, Dispositivo Móvil de Datos Básicos y el Dispositivo Móvil de Datos Mejorados (17).

Los dispositivos móviles de datos limitados: son dispositivos inalámbricos electrónicos para acceder y utilizar los servicios de la red de telefonía móvil, que tienen la misma funcionalidad de un teléfono de línea fija. Poseen teclado simplificado y pantallas pequeñas en algunos casos. Relativamente tienen baja capacidad de memoria y procesamiento en evolución, sistemas operativos propietarios en su mayoría, soporte a Internet móvil, y los nuevos modelos adquieren soporte a formatos multimedia (video, sonido). Los servicios de datos generalmente son mediante SMS (Servicio de Mensajes Cortos, siglas en inglés) y acceso WAP (Protocolo de Aplicaciones Inalámbricas, siglas en inglés). Estos tipos de dispositivos móviles son los celulares (18).

Los dispositivos móviles de datos básicos: son dispositivos móviles de mayor capacidad de almacenamiento de información (hasta 32 GB) y mejor conectividad que un teléfono móvil. Poseen pantalla de mediano tamaño en algunos casos (por lo general no exceden las 5 pulgadas) (19). Ofrecen e-mails, cámara, GPS, bluetooth, lista de direcciones, SMS, navegador Web básico y acceso a Internet mediante redes Wifi, 3G, 4G (20). Estos tipos de dispositivos son los teléfonos inteligentes.

Los dispositivos móviles de datos mejorados: tienen pantalla de mediana a grandes (por encima de los 240 x 120 píxels), ofrecen las mismas características que el dispositivo móvil de datos básicos. Poseen aplicaciones nativas, aplicaciones de Microsoft Office Mobile (Word, Excel, PowerPoint) y aplicaciones corporativas usuales en versión móvil. Entre estos dispositivos portátiles se encuentran los Pocket PC y las tabletas.

Pocket PC es un ordenador de bolsillo, también llamado PDA (Asistente Personal Digital, siglas en inglés). Se trata de un pequeño ordenador, que proporciona capacidades similares a las computadoras de escritorios. Este dispositivo permite grabar, enviar y recibir e-mails, contactos, citas, mostrar archivos multimedia, juegos, intercambiar mensaje de texto con MSN Messenger y navegar por la Web.

Una tableta es un tipo de computadora portátil, de mayor tamaño que un teléfono inteligente o un PDA, su pantalla mide desde 7 pulgadas hasta más de 9 generalmente. Además, tienen conectividad y velocidad de procesamiento mayor que la de los teléfonos inteligentes (21).

Teniendo en cuenta que se estima que Latinoamérica en el año 2015 tendrá 750 millones de conexiones móviles a través de teléfonos inteligentes y tabletas y por las características mencionadas anteriormente se considera que son los indicados a la hora de dar solución a la presente investigación.

Sistemas operativos para teléfonos inteligentes y tabletas

Hoy en día la compra de un teléfono va más allá del diseño. El SO se ha convertido en una de las cosas que suelen resultar buenos factores. Un SO en su sentido más amplio es una *“capa compleja entre el hardware y el usuario, concebible también como una máquina virtual, que facilita al usuario o al programador herramientas e interfaces adecuadas para realizar sus tareas informáticas, abstrayéndolas de los complicados procesos necesarios para llevarlos a cabo”* (22).

Básicamente un sistema operativo proporciona las funciones principales para el dispositivo móvil, tales como el reconocimiento de entrada y visualización de la interfaz de usuario dentro de la pantalla. Pero además dependiendo del sistema operativo que utilice el teléfono, también se verá afectada la rapidez y la fiabilidad de todos los procesos. Actualmente existe una enorme variedad de tipos de SO para teléfonos inteligentes y tabletas, y entre lo más conocidos y populares se encuentran Android, iOS, Blackberry OS, Windows Phone (22).

El sistema operativo Android es de distribución libre y código abierto. Soporta tecnología de conectividad 2G, 3G, 4G. Dispone de una plataforma adaptable a pantallas grandes, gráficos 2D y 3D. En cuanto a mensajes, utiliza SMS y MMS (Servicios de Mensajes Multimedia, siglas en inglés). Permite que los usuarios personalicen sus teléfonos inteligentes y la integración de los servicios de Google. Las aplicaciones móviles se desarrollan habitualmente con una optimización del lenguaje Java, el cual posibilita ahorro de recursos, además del Android SDK desde cualquier sistema operativo (22). Una de las partes más poderosas de Android desde el punto de vista del desarrollo es que se puede reutilizar el código existente para crear aplicaciones funcionales rápidamente. Por ejemplo, si la aplicación necesita de datos GPS, no es necesario escribir el código, sólo hay que llamar al marco GPS existente inherente a Android para su uso en la aplicación. Uso de la cámara, bases de datos SQLite, SMS, son todos iguales de fáciles.

Una de las características que siempre ha establecido Android aparte de ser un feroz competidor de Apple, es el uso de widgets⁷ interactivos en la pantalla de inicio. El actor puede instalar widgets para sus aplicaciones favoritas que le permite interactuar directamente desde la pantalla de inicio del dispositivo. Esta es una característica que aún no está disponible, incluso en la reciente versión de iOS.

⁷ Widgets: pequeñas aplicaciones o programas utilizadas para dar fácil acceso a funciones frecuentemente usadas y proveer de información visual.

En las versiones más recientes de Android, los widgets están incluso disponibles en la pantalla de bloqueo del dispositivo, una característica que aún no ha sido igualada por ninguna otra plataforma móvil, incluyendo Windows Phone y iOS.

El **sistema operativo iOS** es usado por iPhone, iPod e iPad. Uno de los aspectos fundamentales de iOS está dado por la seguridad que posee. Es una plataforma de código cerrado, por lo que no ofrece las posibilidades de personalización que brinda Android (22).

El **Blackberry OS (RIM)** es uno de los sistemas operativos más conocidos dentro del mundo de los teléfonos inteligentes. Los dispositivos de esta marca contienen teclados físicos, pensados y desarrollados para ofrecer facilidad y rapidez para el ingreso de texto. Posee una arquitectura que trabaja con un verdadero ahorro de energía, por eso los dispositivos que pertenecen a esta marca son más ahorrativos en cuanto a la carga de batería. Para el desarrollo de aplicaciones en este SO, se necesita el Blackberry JDE (Entorno de Desarrollo de Java, siglas en inglés), el cual puede ser instalado en casi cualquier Entorno de Desarrollo Integrado (IDE) (22).

El sistema operativo **Windows Phone** es compatible con casi todos los programas que funcionan en Windows. Es un sistema de código cerrado, por lo que no ofrece las posibilidades de personalización que brinda Android (22).

Firefox OS es un sistema operativo nuevo por lo que no brinda la estabilidad que ofrece Android. Las aplicaciones web desarrolladas tienen como base de programación el HTML 5, JavaScript y CSS. Aunque el HTML 5 es un estándar, todavía tiene que evolucionar para que se visualice todo su potencial en cuanto al desarrollo de aplicaciones complejas (23).

Luego de analizar cada una de las características de los sistemas operativos móviles mencionados anteriormente, se decide tomar Android como SO para el desarrollo de la presente investigación, ya que las principales compañías de móviles usan Android y actualmente más del 50 % de los teléfonos inteligentes y tabletas del mundo lo utilizan demostrando la aceptación de este en los usuarios (22).

Todas las aplicaciones que se utilicen deben ser desarrolladas para la mayor cantidad de dispositivos móviles posibles. Actualmente el Android SDK tiene librerías, que permiten desarrollar aplicaciones móviles para todas las versiones de Android a partir de la 2.2.

1.6 Soluciones similares

Al igual que el Repositorio de Objetos de Aprendizaje RHODA, hay varias aplicaciones que buscan incorporar la nueva estrategia de aprendizaje llamada m-learning. A continuación se hace un análisis de algunas aplicaciones, indagando en sus principales funcionalidades:

AppCentral: Los estudiantes de la Facultad de Ciencias Sociales Humanidades y Arte de la Universidad Central desde enero cuentan con App Central para teléfonos inteligentes y tabletas, donde tienen acceso a las notificaciones de la agenda cultural de los departamentos de Música, Arte Dramático, Estudios Musicales y Comunicación Social y Periodismo. Otras de sus funcionalidades es hacer solicitud de certificados y procesos de preinscripción sin tener que ir a la universidad, y tener acceso a un directorio de profesores, en el que pueden consultar sus temáticas de interés y su disponibilidad para tutor. La App de la Universidad Central comenzó a crearse desde septiembre de 2013. En principio, se hicieron pruebas con el departamento de Publicidad, que lideró esta iniciativa y hoy ya se encuentra disponible para Android.

ZeraApp: es una aplicación educativa para teléfonos inteligentes y tabletas con Android, que utiliza tecnologías webs comunes, entre ellas REST (Transferencia de Estado Representacional, siglas en inglés) como protocolo para obtener y enviar información a la plataforma ZERA. Algunas de las funcionalidades que brinda son: autenticación en el sistema; actualizar la aplicación; realizar búsqueda de usuarios; agregar a los contactos del móvil un usuario de la app; realizar llamadas y enviar SMS a un usuario; crear un nuevo comentario en un foro; mostrar y descargar archivos asociados con las evidencias; listar calificaciones, comentarios, foros y notificaciones.

AUMovil: es una aplicación de software que opera con el sistema operativo Android y busca dar soporte a talleres y cursos en la Universidad Metropolitana unidad Cuajimalpa. Entre las funcionalidades que tiene son: soporte multiusuarios con distintas funcionalidades, vistas, privilegios y subsistemas. Llevar la gestión de: los cursos y talleres que se imparten en ciclo escolar; el horario escolar de los alumnos en dos modalidades: presencial con ayuda de personal de Sistemas Escolares y remota realizada completamente por el alumno; la difusión de avisos relacionados con el proceso de inscripción de Sistemas Escolares hacia los alumnos y la distribución de turnos para realizar el proceso de inscripción y modificación de horario (24).

Las aplicaciones mencionadas no compensan la necesidad de la investigación, debido a que son adaptadas al sistema para la que fueron realizadas. Sin embargo, se estudió el impacto que estas tuvieron

y la aceptación de las funcionalidades desarrolladas, sirviendo de guía para el desarrollo de la actual investigación.

1.7 Herramientas y tecnologías para el sistema operativo Android

Para la realización del análisis y diseño de una aplicación móvil, se hace necesaria la descripción de un conjunto de tecnologías y herramientas para el sistema operativo Android.

ANDROID SDK

El Android SDK 22.6 brinda las librerías y herramientas necesarias para construir, probar y depurar aplicaciones para Android (22). A través de este, Google ofrece un paquete completo que incluye depurador, emulador, documentación, demos y el código de muchas aplicaciones de ejemplo; armando al desarrollador con todo lo necesario para la creación de aplicaciones. Este paquete de desarrollo para su funcionamiento debe de ser importado en un IDE.

Lenguajes de desarrollo

El lenguaje **Java** es el más usado en el desarrollo de aplicaciones para Android. El lenguaje tiene un modelo más simple que C y C++ y elimina herramientas de bajo nivel. Este potente lenguaje de programación ofrece múltiples ventajas, algunas de ellas son: independencia de la plataforma es un lenguaje natural y sencillo. La seguridad inherente a la no existencia de punteros hace el código seguro; y las capacidades inherentes multihilo del lenguaje y su alto rendimiento lo hacen ideal para todo tipo de entorno.

Por lo antes mencionado se propone el lenguaje Java en su versión 8, para el desarrollo de la presente investigación.

Entorno de Desarrollo Integrado

NetBeans es un IDE libre creado principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. Este se vincula con Android a través del ADT Plugin (Plugin de Herramientas de Desarrollo para Android, siglas en inglés) disponible para las versiones 7.2 y 7.3 del NetBeans. Entre las funcionalidades que brinda se encuentran: soporte al núcleo del Android SDK;

permite el uso de emuladores y dispositivos reales para ejecutar los proyectos; integra el visor LogCat⁸, imprescindible para saber que está pasando con el proyecto que se encuentra ejecutándose.

Eclipse es un entorno de desarrollo de código abierto (22). Utiliza el ADT Plugin, con el objetivo de integrar el Android SDK para utilizar sus herramientas, crear nuevos proyectos de Android y agregar a las clases paquetes propios de la API (Interfaz de Programación de Aplicación, siglas en inglés) del Android Framework.

Android Studio es un nuevo entorno de desarrollo para Android, basado en IntelliJ IDEA⁹. Incorpora nuevas características que no han sido incorporadas en el tradicional IDE basada en Eclipse (27).

Motodev Studio for Android: es un entorno de desarrollo creado por Motorola para el desarrollo de aplicaciones Android. Se basa en Eclipse y añade características adicionales. Es la opción más sencilla para usuarios sin experiencias en Eclipse. Algunas de sus características son: dispone de Eclipse IDE, herramientas Java y Plugin de Android perfectamente integrados. Agrega códigos para resolver tareas frecuentes, ejemplo enviar SMS, hacer vibrar el teléfono. Permite ver y editar la base de datos SQLite para las aplicaciones Android. Prueba las aplicaciones sin cambiar del entorno de desarrollo al emulador del terminal. Dispone de muchos tutoriales que se pueden ver desde el interior del entorno de desarrollo. Se puede instalar la versión en castellano. Genera parte del código repetitivo de Java. Se puede desarrollar la aplicación en múltiples idiomas. Utiliza las herramientas avanzadas de diagnóstico para detectar problemas de memoria y probar las aplicaciones antes de su distribución.

Como la aplicación “RHODA” se desarrollará para el sistema operativo móvil Android y por las características mencionadas anteriormente, el entorno de desarrollo para la propuesta de solución es Motodev Studio for Android en su versión 4.0.

Servicios web

Los servicios web se definen como “...un conjunto de tecnologías con capacidad para operar en la Web intercambiando datos entre sí con el objetivo de ofrecer servicios. Estos servicios proporcionan mecanismo de comunicación estándares entre diferentes aplicaciones interactuando entre sí para presentar información dinámica al usuario” (28).

⁸LogCat: herramienta insertada en la mayoría de los IDE que permite la visualización de mensajes o los emitidos por la aplicación que se está ejecutando.

⁹IntelliJ IDEA: es un entorno para desarrollar aplicaciones Java, cliente y servidor. Se trata de un completísimo IDE que también permite desarrollar aplicaciones para móviles.

La W3C (World Wide Web) define un servicio web como un sistema software diseñado para soportar una interacción interoperable entre diferentes equipos en red (29). Dentro de las implementaciones comunes de servicios web se encuentran SOAP Y REST. El primero hace referencia al protocolo usado para la comunicación entre cliente y servidor intercambiando mensajes basados en XML (Lenguaje de Marcas Extensibles, siglas en inglés), mientras que en el segundo caso los mensajes son por lo general en formato JSON (Notación de objetos de JavaScript, siglas en inglés) (30).

Para dar solución a la presente investigación se define como estándar de comunicación a REST y de transmisión de datos a JSON por ser de formato ligero para el intercambio de datos, de fácil comprensión para los seres humanos. Además, en este trabajo sustituyen a XML por JSON, debido a su mayor velocidad de procesamiento para los equipos de cómputo (30).

En (30) se demuestra como usando XML, se requiere un proceso de interpretación o análisis, como es característico en el procesamiento del lenguaje, mientras que usando JSON, se presenta un nuevo enfoque, pasando por alto la necesidad de procesamiento del lenguaje, que lo convierte en una solución más ágil.

Estándar para la comunicación REST

El protocolo REST, plantea un estilo de arquitectura cliente-servidor en la cual un servicio es visto como un recurso identificado a través de una dirección URL, mediante el cual puede ser consumido. Para consumir estos servicios web, se hace uso de mensajes simples, los cuales se intercambian entre cliente y servidor (30). REST define a partir de HTTP (Protocolo de Transferencia de HyperText, siglas en inglés), cuatro métodos: GET, PUT, POST, DELETE, de los cuales los más usados son GET y PUT. El primero de los métodos es usado para enviar la representación de un recurso o servicio al cliente, mientras que el otro es usado para transferir el estado de un cliente al recurso (30). Para la comunicación e intercambio de información entre cliente y servidor a través de REST, se puede hacer uso de diferentes lenguajes: XML, HTML, JSON.

Estándar para el intercambio de datos JSON

JSON es un formato ligero basado en de texto, con definición de lenguaje completamente independiente, cuya sintaxis está basada en un subconjunto de funcionalidades y palabras reservadas del lenguaje JavaScript. JSON usa las convenciones para el manejo de datos, que son comunes a la familia de lenguajes: C, C++, C# y Java; esta característica hace de JSON un lenguaje ideal para el intercambio de datos entre aplicaciones cliente-servidor o aplicaciones que se ejecuten en diferentes escenarios (30).

1.8 Metodología de desarrollo del software

Se entiende por metodología de desarrollo una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software (31). Una metodología es necesaria en el proceso de desarrollo de software para obtener un sistema con calidad, en el tiempo esperado y con el coste planificado.

Metodologías ágiles

En una reunión celebrada en febrero de 2001 en Utah-EEUU, nace el término "ágil" aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto (32).

Metodologías tradicionales

Existen una serie de metodologías que suelen llamar tradicionales, propuestas casi todas ellas con anterioridad a los años 1990, que pretendían ayudar a los profesionales indicando pautas para realizar y documentar cada una de las tareas del desarrollo del software. Las metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo el software, con el fin de conseguir un software más eficiente. Para ello, se hace énfasis en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del software (33).

Tabla 1-Comparación de las metodologías tradicionales y ágiles (34).

Metodologías Tradicionales	Metodologías Ágiles
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo. Cierta resistencia a los cambios.	Basadas en heurísticas provenientes de prácticas de producción de código. Especialmente preparados para cambios durante el proyecto.
Impuestas externamente. Proceso mucho más controlado, con numerosas políticas/normas.	Impuestas internamente (por el equipo). Proceso menos controlado, con pocos principios.
El cliente interactúa con el equipo de desarrollo mediante reuniones. Más	El cliente es parte del equipo de desarrollo. Pocos artefactos.

artefactos.	
Más roles. Grupos grandes y posiblemente distribuidos.	Pocos roles. Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.
La arquitectura del software es esencial y se expresa mediante modelos.	Menos énfasis en la arquitectura del software.

Proceso Unificado de Rational

Basado en la comparación de las metodologías, teniendo en cuenta que es robusta y considerando el tamaño del equipo del proyecto, se decide utilizar para el desarrollo de la presente investigación la metodología tradicional, con el fin de conseguir un software eficiente. Como metodología tradicional está el RUP (Proceso Unificado de Rational, siglas en inglés).

RUP es un conjunto de actividades que permiten transformar los requerimientos de un usuario en un sistema de software. Es un marco de trabajo genérico que puede especializarse para una gran variedad de: sistemas de software, áreas de aplicación, tipos de organizaciones de niveles de aptitud y tamaños de software (35). Esta metodología define “quién” (trabajadores) debe hacer “qué” (artefactos), “cuándo” (flujo de trabajo y fases) y “cómo” (actividades) debe hacerlo.

En RUP se divide el desarrollo de software en cuatro fases: Inicio, Elaboración, Construcción y Transición; estas fases son desarrolladas mediante un ciclo de iteraciones que son llevadas bajo nueve disciplinas conocidas también como flujos de trabajo, las cuales las seis primeros son de ingeniería y las tres restantes de apoyo. Estos flujos son: Modelado de Negocio, Requerimientos, Análisis y Diseño, Implementación, Pruebas, Despliegue, Configuración y Administración de Cambio, Administración de Proyecto y Entorno.

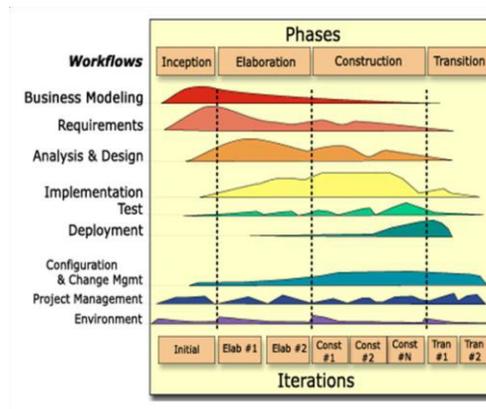


Figura 2-Fases e Iteraciones de la Metodología RUP.

RUP cuenta con numerosas características, tres de estas lo hacen único: dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental. Existen otras no menos importantes entre las que se encuentran que utiliza UML como lenguaje de modelado, el tipo de programación que utiliza es orientado a objetos (OO). Es una metodología adaptable que permite que se le realicen los cambios que se necesite.

Lenguaje de modelado

UML (Lenguaje Unificado de Modelado, siglas en inglés), notación con que se construyen sistemas por medio de conceptos orientados a objetos. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software orientado a objetos. Además de ser el lenguaje más conocido y usado en la actualidad.

Ofrece un estándar para describir un plano del sistema incluyendo aspectos conceptuales, tales como: procesos de negocio, funciones del sistema, aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. Ha sido adoptado por muchos proveedores de herramientas CASE y es el lenguaje de modelado soportado por RUP, metodología escogida para el desarrollo del sistema.

1.9 Herramienta CASE

Las herramientas CASE (Computer Aided Software Engineering) son un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores durante todos los pasos del ciclo de vida de desarrollo de una aplicación (36).

Visual Paradigm para UML

La herramienta CASE Visual Paradigm (VP), brinda soporte para UML 2.0, soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, implementación y pruebas. Ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación. VP es multiplataforma por su estabilidad de ejecución en diferentes sistemas operativos, también proporciona abundante tutoriales y demostraciones interactivas (37).

La herramienta CASE Rational Rose es la que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML. Esta herramienta propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del diseño, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software (38). Rational Rose es una herramienta basada en Eclipse. Sirve para plataformas Unix, Linux y Windows. Brinda soporte para UML 2.0 y modelado de datos.

Las dos herramientas mencionadas anteriormente son muy potentes, soportan todos los diagramas de UML y además ayudan para la gestión de requisitos de software. Por sus características y los beneficios que brinda para la construcción del software, especialmente por tener una licencia libre y ser multiplataforma se seleccionó Visual Paradigm para UML 8.0 para el modelado de la presente investigación. Además, se tuvo en cuenta que es la herramienta que se utilizó para generar los artefactos del repositorio RHODA.

Conclusiones parciales

Con el resultado de los estudios e investigaciones realizados en este capítulo, dio como resultado la importancia que tiene llevar a cabo la implementación de aplicaciones móviles para la educación, debido al auge que están tomando los dispositivos móviles a nivel mundial. Se define la metodología de desarrollo de software RUP, la herramienta asociada para el modelamiento de los diagramas correspondientes al flujo de trabajo análisis y diseño de esta investigación Visual Paradigm y el lenguaje de modelado UML. Además, se realiza la propuesta de herramientas y tecnologías necesarias para la futura implementación de la aplicación móvil vinculada al Repositorio de Objetos de Aprendizaje RHODA.

Capítulo 2 Características del sistema

Introducción

En este capítulo, se realiza una descripción de la propuesta de solución en aras de una fácil comprensión. Se identifican los actores que interactúan con la aplicación, los casos de uso que inicializan y los requerimientos funcionales y no funcionales que conllevaron a dichos casos de uso. Además, se realizó la relación de actores con casos de uso mediante un diagrama de clases, así como el diagrama de navegabilidad con el objetivo de una mejor vista sobre el funcionamiento de la aplicación.

2.1 Modelo del dominio

El flujo de trabajo Modelamiento del negocio determina el funcionamiento del negocio. Es un modelo real de las personas, grupos o entidades que se benefician de la actividad desarrollada en el negocio (actores del negocio) y la interacción con las personas, que realizan estas actividades (trabajadores del negocio).

En este flujo se describen las entidades, actividades y procesos (casos de uso del negocio) que se realizan, para entender las implicaciones y ámbito del negocio, que sirva de guía en la realización de los restantes flujos de trabajo.

Un modelo de dominio constituye una especificación de los conceptos más importantes de la vida real que se utilizarán en el sistema, *“una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés”* (39). Es de gran ayuda para desarrolladores y usuarios, ya que de esta forma utilizan un vocabulario común y pueden entender el contexto en que se enmarca el sistema.

RUP define la creación de un modelo del dominio como uno de los artefactos a generar, durante el flujo de trabajo: Modelamiento del negocio. El modelo del dominio se construye utilizando la notación UML y se representa a través de un conjunto de diagramas de clases, en los que no hay operaciones definidas (40) y puede mostrar:

- Objetos de dominio o clases conceptuales.
- Asociaciones entre las clases conceptuales.
- Atributos de las clases conceptuales.

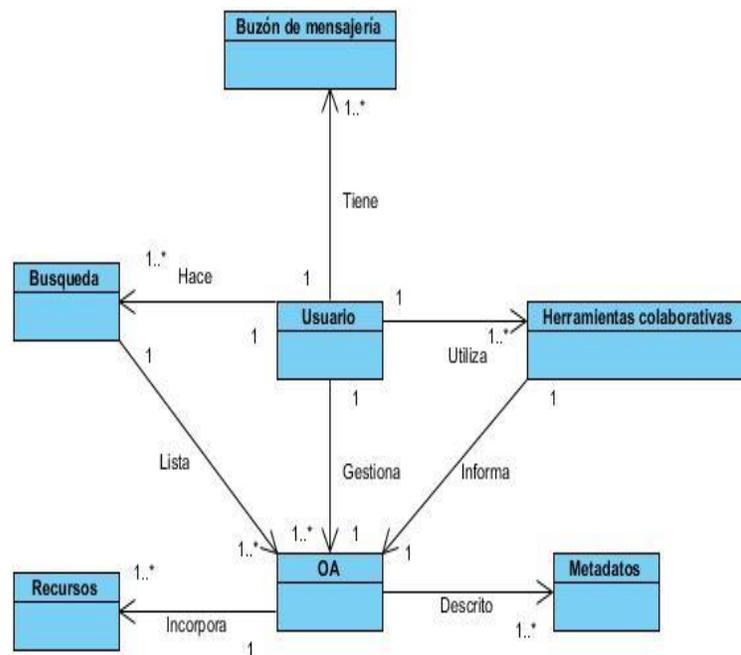


Figura 3-Modelo del dominio.

Definición de conceptos fundamentales

Usuario: Representa los usuarios del sistema.

OA: Recurso digital dirigido a la formación, fácilmente identificado y encontrado a través de sus metadatos, con capacidades de integración en diferentes sistemas e-learning, para ser usado con distintos propósitos educativos y para combinarse con otros objetos formando una sola entidad.

Metadatos: Conjunto de información que describe los OA, para facilitar su búsqueda y reutilización dentro del mismo.

Recursos: Un OA puede incorporar cualquier tipo de archivo de tipo imagen, audio, sonido o texto, etc.

Buzón de mensajería: Servicio que posee la aplicación, al cual son enviadas algunas de las notificaciones que genera el sistema. Cada usuario posee un buzón de mensajería interna, y la aplicación tiene definidas qué notificaciones envía a este buzón.

Búsqueda: Servicio que posee la aplicación, utilizado por los usuarios con el fin de localizar un OA.

Herramienta colaborativa: Como herramienta colaborativa están los foros, que permiten la interacción de los usuarios y la colaboración de unos con otros a fin de lograr un objetivo específico.

2.2. Requerimientos del sistema

“Los requerimientos son una especificación de lo que debe ser implementado. Estos son descripciones de cómo el sistema se debe comportar, de las propiedades y atributos del mismo. Deben ser una restricción del proceso de desarrollo del sistema” (41).

Los requisitos de un sistema o software son la condición o capacidad que debe exhibir o poseer un sistema para satisfacer un contrato, estándar, especificación, u otra documentación formalmente impuesta (IEEE). A continuación se exponen los siguientes requisitos con los que debe cumplir la aplicación.

Requisitos funcionales

Un requisito funcional es una descripción de lo que un sistema debe hacer. Este tipo de requisito especifica algo que el sistema entregado debe ser capaz de realizar. Después de un análisis de las funcionalidades básicas de un Repositorio de Objetos de Aprendizaje y de las soluciones similares mencionadas en el Capítulo 1, fueron definidos los siguientes:

RF1. Listar publicaciones de OA: Muestra las opciones: **RF 1.1, RF 1.2 y el RF 1.3.**

RF1.1. Listar OA más recientes: Lista los últimos OA publicados en el sistema, mostrando por cada uno el título.

RF1.2. Listar OA más solicitados: Lista los OA que han sido más solicitados en el sistema, mostrando por cada uno el título.

RF 1.3. Listar OA por categorías: Lista los OA por categorías. De cada uno se mostrará el título.

RF 2. Listar OA por revisar: Lista los OA que deben ser revisados, mostrando por cada uno el título.

RF 3. Listar OA en revisión: Lista los OA que se encuentran en revisión, mostrando por cada uno el título.

RF 4. Autenticar usuario: El sistema debe permitir que el usuario acceda a la portada del sistema. Para hacer la operación debe llenar los siguientes campos: usuario (dominio UCI y ser registrado) y contraseña.

RF 5. Realizar servicio de mensajería: El sistema brindará la posibilidad de utilizar la mensajería interna del sistema, los cuales son: **RF 5.1, RF 5.2 y el RF 5.3.**

RF 5.1. Crear mensaje: El sistema debe permitir que el usuario pueda crear un mensaje. Para realizar la operación debe llenar los campos: destinatarios y la descripción del correo.

RF 5.2. Mensajes enviados: Muestra al usuario un listado de los mensajes que ha enviado el usuario.

RF 5.3. Ver bandeja de entrada: Muestra al usuario el listado de los mensajes recibidos.

RF 6. Recuperar contraseña: El sistema debe permitir que el usuario pueda recuperar su contraseña en caso de que la haya olvidado. Para realizar la operación debe llenar el campo: dirección de correo.

RF 7. Ver usuarios conectados: El sistema debe permitir que el usuario pueda ver todos los usuarios conectados en la plataforma.

RF. 7.1 Agregar a los contactos del móvil un usuario de la aplicación: El sistema debe permitir agregar el número telefónico de un usuario a la lista de contactos del móvil.

RF. 7.2 Enviar SMS al número de teléfono de un usuario seleccionado: El sistema debe permitir enviar un mensaje al número de teléfono de un usuario seleccionado.

RF. 7.3. Llamar al número de teléfono de un usuario de la aplicación: El sistema debe permitir llamar al número telefónico de un usuario.

RF 8. Gestionar perfil: El sistema debe permitir que el administrador pueda modificar o eliminar un usuario de la aplicación.

RF 9. Actualizar contenido: El sistema debe permitir que el usuario actualice los servicios que contiene la aplicación.

RF 10. Gestionar OA: El sistema debe permitir que el administrador pueda eliminar o modificar los metadatos de un OA.

RF. 11. Mostrar portada: Muestra al usuario todas las funcionalidades de la aplicación dependiendo del rol.

RF. 12 Registrar usuario: El sistema debe permitir que el administrador registre un usuario. Para realizar la operación debe llenar los siguientes campos: usuario (dominio UCI y ser registrado), contraseña, nombre, rol, dirección del correo y número de teléfono.

Requisitos no funcionales.

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener (42) para desarrollar un producto atractivo y con la calidad requerida.

Requerimiento de apariencia

- Usar la menor cantidad de imágenes posibles.

Requerimiento de Confiabilidad

- Solo con una combinación de usuario (dominio UCI y registrado) y contraseña, el usuario podrá acceder a la portada de la aplicación.

Requerimiento de interfaz de comunicación

- Las conexiones mediante internet, se va a realizar por vía Wifi.

2.3. Personas relacionadas con el sistema

Se define como usuarios relacionados con el sistema a todas aquellas personas que de una forma u otra van a interactuar con la aplicación.

Tabla 2-Personas relacionadas con el sistema.

Personas relacionadas con el sistema.	Justificación
Usuario anónimo	Es la persona que navega por el sistema, sin estar autenticado. No va a tener los privilegios que la aplicación define para usuarios autenticados.
Usuario autenticado	Es la persona que se encuentra registrada al sistema. Además de tener sus privilegios, va a obtener los definidos para usuarios anónimos.
Revisor	Es la persona que cuenta con los privilegios de usuario autenticado, además de ser encargada de realizar los procesos de revisión de los aspectos relacionados con los metadatos de los OA.
Administrador	Es la persona que cuenta con los privilegios del usuario autenticado. Se encarga de la gestión del usuario y de los OA. Tiene libertades ilimitadas para ejecutar todas las funciones administrativas.

Diagrama de actores del sistema

Un actor es un usuario del sistema. Incluye usuarios humanos y otros sistemas computarizados. Un actor usa un caso de uso para desempeñar alguna porción de trabajo que es de valor para el negocio. El

conjunto de casos de uso al que un actor tiene acceso define su rol global en el sistema y el alcance de su acción (43).

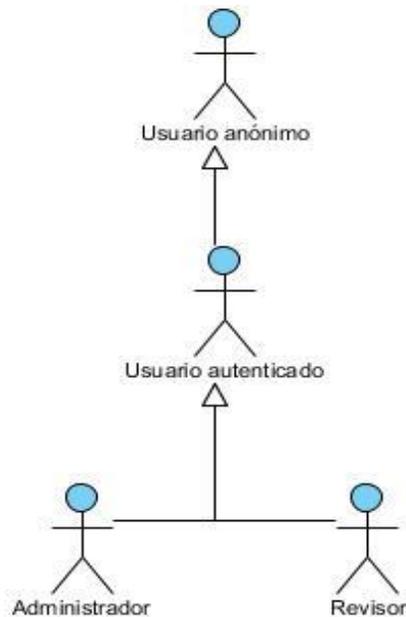


Figura 4-Diagrama de actores del sistema.

2.4 Patrones de casos de uso

La experiencia en la utilización de casos de uso (CU) ha evolucionado en un conjunto de patrones que permiten con precisión reflejar los requisitos reales, haciendo más fácil el trabajo con los sistemas y mucho más simple su mantenimiento.

A continuación, se explican los patrones presentes en el diagrama de CU del sistema:

Navegabilidad: Indica quién inicia la comunicación en la interacción y se muestra con una flecha. El sentido de la flecha indica: Si apunta al CU, la comunicación la inicia el actor del sistema.

CRUD (Crear, Leer, Modificar, Eliminar): Este patrón se basa en la fusión de casos de uso simples para formar una unidad conceptual.

- ✓ CRUD Parcial: Es utilizado cuando se decide separar una de las acciones debido a que esta es más importante, larga o mucho más compleja que el resto de las acciones.

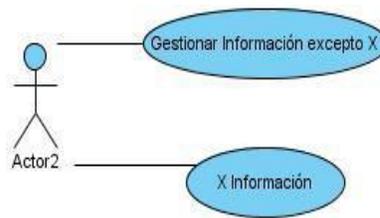


Figura 5-CRUD Parcial.

Múltiples actores (roles comunes): Este patrón es aplicable cuando los actores juegan roles comunes durante el desempeño de determinadas tareas, en este caso, los actores heredan el comportamiento del rol común cuando interactúa con un determinado CU.

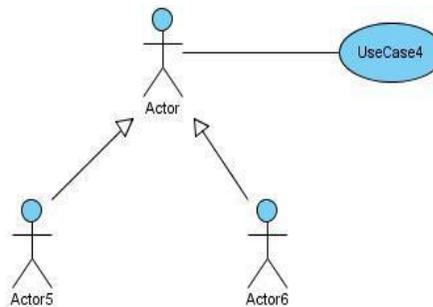


Figura 6-Múltiples actores (roles comunes).

2.5 Diagrama de casos de uso identificados

El modelo de casos de uso describe las funcionalidades propuestas. Un caso de uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema. Es una unidad simple de trabajo significativo; por ejemplo, "Validarse en el sistema", "Registrarse en el sistema" y "Crear un pedido" son todos casos de uso (44).

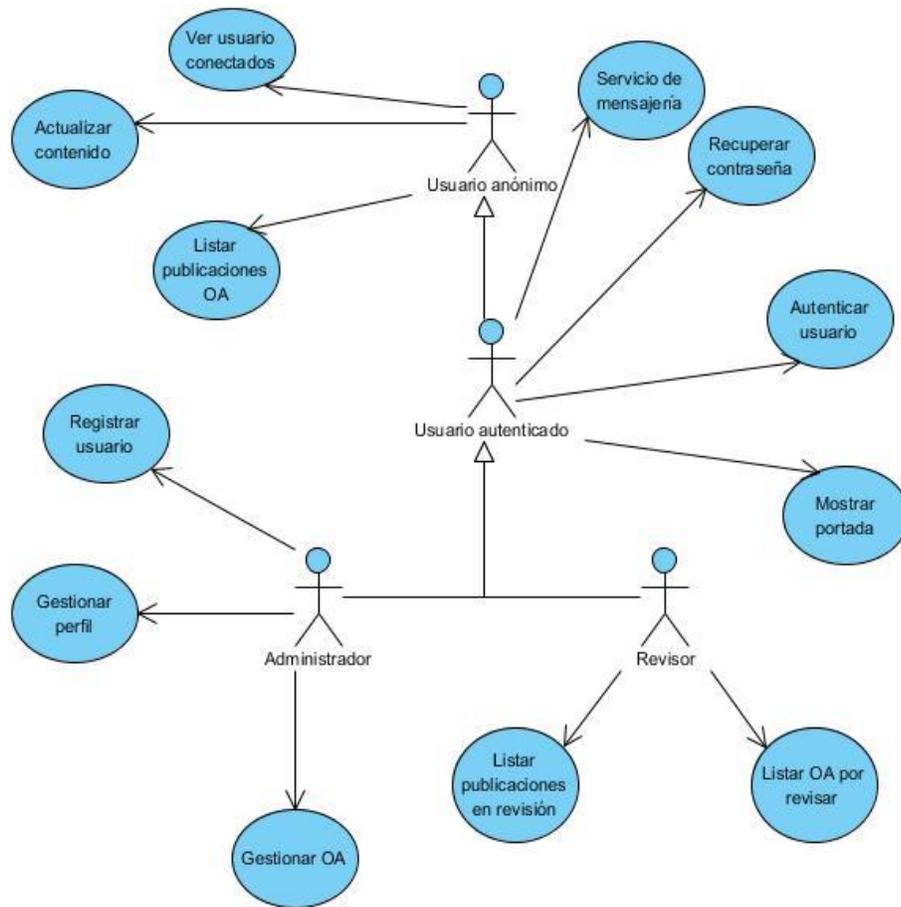


Figura 7-Diagrama de casos de uso.

2.6 Listado de casos de uso

Cada caso de uso tiene una descripción que describe la funcionalidad que se construirá en el sistema propuesto. Un caso de uso puede "incluir" la funcionalidad de otro caso de uso o "extender" a otro caso de uso con su propio comportamiento.

La descripción detallada de los casos de uso se encuentra en el **Anexo I: Descripción textual de casos de uso**.

Tabla 3-Caso de Uso "Autenticar Usuario".

Caso de Uso:	Autenticar Usuario
Actores:	Administrador, revisor, usuario autenticado

Resumen:	El caso de uso inicia cuando el actor decide autenticarse en la aplicación. El sistema muestra una pantalla donde se deberá introducir el usuario y contraseña. El sistema permite la autenticación en él. Termina el caso de uso.
Precondiciones:	El actor debe estar registrado en el sistema.
Referencias	RF. 12, requerimiento de apariencia, requerimiento de confiabilidad, requerimiento de interfaz de comunicación.
Prioridad	media

Tabla 4-Caso de Uso “Recuperar contraseña”.

Caso de Uso:	Recuperar contraseña olvidada.
Actores:	Administrador, revisor, usuario autenticado
Resumen:	El caso de uso inicia cuando se selecciona la opción de recuperar contraseña. El sistema muestra una pantalla donde se deberá introducir la dirección de correo. El sistema permite la recuperación de los datos perdidos, enviándole un mensaje al correo especificado. Termina el caso de uso.
Precondiciones:	El actor debe estar registrado en el sistema.
Referencias	-RF. 12, requerimiento de apariencia, requerimiento de interfaz de comunicación.
Prioridad	Media

Tabla 5-Caso de Uso “Servicio de Mensajería”.

Caso de Uso:	Servicio de Mensajería
Actores:	Administrador, usuario autenticado, revisor
Resumen:	El caso de uso inicia cuando se selecciona la opción de servicio de mensajería. El sistema muestra una pantalla con las opciones (ver bandeja de entrada, ver mensajes enviados y crear mensaje) las que el actor puede seleccionar. Si selecciona ver bandeja de entrada o mensajes enviados, el sistema muestra un listado de mensajes. El actor selecciona el mensaje a ver y el sistema brinda la

	opción de eliminar o reenviar. En el caso de que el actor seleccione la opción de crear mensaje, el sistema muestra una pantalla donde el usuario deberá entrar la dirección del remitente, y deberá introducir el mensaje a enviar. Termina el caso de uso.
Precondiciones:	El actor debe estar autenticado en el sistema.
Referencias	RF 4. RF. 12, requerimiento de apariencia, requerimiento de interfaz de comunicación.
Prioridad	Media

Tabla 6-Caso de Uso “Listar publicaciones de OA”.

Caso de Uso:	Listar publicaciones de OA
Actores:	Administrador, revisor, usuario autenticado, usuario anónimo
Resumen:	El caso de uso inicia cuando se selecciona la opción de listar publicaciones de OA. El sistema muestra una pantalla con las opciones (por categoría, recientes, más solicitados) a seleccionar. Si el actor selecciona la opción de recientes o más solicitados, el sistema muestra el listado de los OA. El actor selecciona el OA a ver, el sistema muestra la información que contiene y permite las opciones de descargar, agregar comentario y calificar. En caso de que seleccione la opción de por categoría, el sistema mostrará un listado de todas las categorías (Matemática, Sistema Operativo, Base de datos...), el actor debe seleccionar para qué categoría, luego el sistema muestra el listado de OA. El actor selecciona el OA, el sistema muestra la información que contiene y permite la opción de calificar, comentar, descargar. Termina el caso de uso.
Precondiciones:	-
Referencias	Requerimiento de apariencia.
Prioridad	Media

Tabla 7-Caso de Uso “Listar OA en revisión”.

Caso de Uso:	Listar OA en revisión
Actores:	Revisor
Resumen:	El caso de uso inicia cuando el actor decide listar los OA que estén en revisión. El sistema muestra una pantalla con el listado de todos los OA que se están revisando. El revisor selecciona el OA a visualizar, el sistema muestra la información que contiene. Termina el caso de uso.
Precondiciones:	El actor tiene que estar autenticado.
Referencias	RF. 4, RF. 12, requerimiento de apariencia.
Prioridad	Media

Tabla 8-Caso de Uso “Listar OA por revisar”.

Caso de Uso:	Listar OA por revisar
Actores:	Revisor
Resumen:	El caso de uso inicia cuando el actor decide ver todos los OA que faltan por revisar. El sistema muestra una pantalla con el listado de todos los OA que no han sido revisados. El revisor selecciona el OA a visualizar, el sistema muestra la información que contiene. Termina el caso de uso.
Precondiciones:	El actor tiene que estar autenticado en el sistema.
Referencias	RF. 4, RF. 12, requerimiento de apariencia.
Prioridad	Media

Tabla 9-Caso de Uso “Ver usuarios conectados”.

Caso de Uso:	Ver usuarios conectados
Actores:	Administrador, usuario autenticado, usuario anónimo, revisor

Resumen:	El caso de uso inicia cuando el actor decide ver los usuarios conectados. El actor selecciona el usuario a visualizar y el sistema muestra sus datos. Termina el caso de uso.
Precondiciones:	-
Referencias	Requerimiento de apariencia, requerimiento de interfaz de comunicación.
Prioridad	Media

Tabla 10-Caso de Uso "Mostrar la portada".

Caso de Uso:	Mostrar la portada del sistema
Actores:	Usuario autenticado, administrador, revisor
Resumen:	El caso de uso inicia cuando el actor se autentica correctamente. La aplicación muestra una pantalla con todas las funcionalidades que brinda el sistema dependiendo del rol del actor.
Precondiciones:	Debe estar autenticado en el sistema.
Referencias	RF. 4, RF. 12, requerimiento de apariencia, requerimiento de confiabilidad, requerimiento de interfaz de comunicación.
Prioridad	Media

Tabla 11-Caso de Uso "Actualizar contenido".

Caso de Uso:	Actualizar contenido
Actores:	Administrador, revisor, usuario autenticado, usuario anónimo
Resumen:	El caso de uso inicia cuando el actor decide actualizar la información que contiene la aplicación. Se descarga el nuevo contenido y se guarda en la base de datos del dispositivo, sustituyendo el ya existente.
Precondiciones:	Debe estar autenticado en el sistema.
Referencias	RF. 4, RF. 12, requerimiento de apariencia, requerimiento de interfaz de

	comunicación.
Prioridad	Media

Tabla 12-Caso de uso "Gestionar perfil".

Caso de Uso:	Gestionar perfil
Actores:	Administrador
Resumen:	El caso de uso inicia cuando el actor decide eliminar o modificar el perfil de un usuario. Se realiza la operación y se actualiza la base de datos.
Precondiciones:	Debe estar autenticado en el sistema.
Referencias	RF. 4, RF. 12, requerimiento de apariencia, requerimiento de interfaz de comunicación.
Prioridad	Media

Tabla 13-Caso de uso "Gestionar OA".

Caso de Uso:	Gestionar OA
Actores:	Administrador
Resumen:	El caso de uso inicia cuando el administrador decide eliminar o modificar un OA. Se realiza la operación y se actualiza la base de datos.
Precondiciones:	Debe estar autenticado en el sistema.
Referencias	RF. 4, RF. 12, requerimiento de apariencia, requerimiento de interfaz de comunicación.
Prioridad	Media

Tabla 14-Caso de uso "Registrar usuario".

Caso de Uso:	Registrar usuario
Actores:	Administrador
Resumen:	El caso de uso inicia cuando el actor decide registrar un usuario. El sistema muestra los datos a llenar.
Precondiciones:	Debe estar autenticado en el sistema.
Referencias	RF. 4, RF. 12, requerimiento de apariencia, requerimiento de interfaz de comunicación.
Prioridad	Media

2.7 Diagrama de estructura de navegación

Los diagramas de estructura de navegación definen las rutas de navegación que permite al usuario acceder a los contenidos y a los servicios (45). Surgen porque el lenguaje UML no cuenta con herramientas suficientes para la modelación de aplicaciones especializadas o avanzadas, por ejemplo en UML hace falta valerse de otros recursos a la hora de describir el ensamblaje temporal entre diferentes objetos medias.

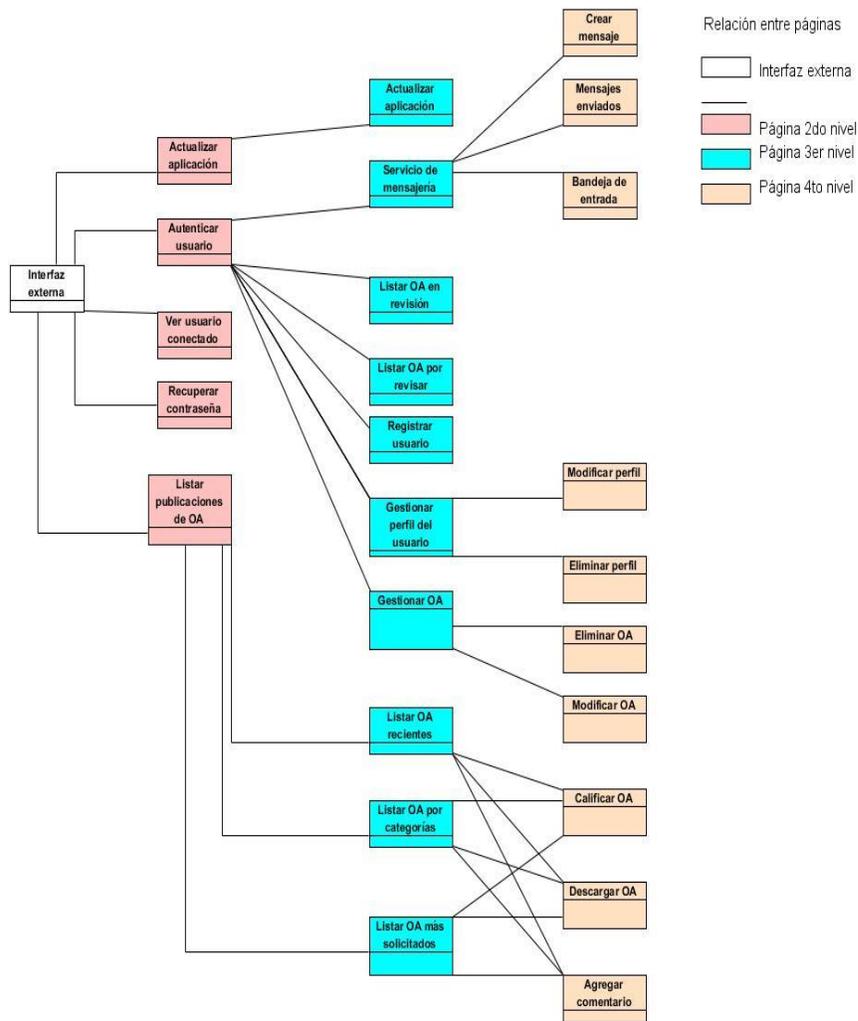


Figura 8-Diagrama de estructura de navegación.

Conclusiones parciales

A partir de la realización del presente capítulo, se identificaron los principales procesos que el sistema debe realizar, así como los actores que interactúan con la aplicación y cada una de las acciones que estos realizan. Partiendo de los resultados de esta investigación, se comenzará con la realización de la propuesta de solución.

Capítulo 3: Análisis y Diseño del sistema

Introducción

El presente capítulo se centra en modelar el flujo de trabajo Análisis y Diseño que propone la metodología RUP, donde se representan los artefactos correspondientes. De forma general este capítulo abarca los diagramas de clases del análisis, los diagramas de interacción específicamente, el diagrama de colaboración y el diagrama de clases del diseño orientado a objetos; basados en las descripciones textuales propuestas en el capítulo anterior. Además, se realiza la validación de algunos de estos artefactos utilizando métricas.

3.1 Modelo del análisis

Análisis es uno de los flujos de trabajo que se encuentran dentro de la fase de elaboración. Esta disciplina tiene como propósito:

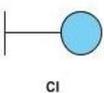
- Conseguir una comprensión más precisa de los requisitos, refinarlos y estructurarlos.
- Utilizar el lenguaje de los desarrolladores para analizar con profundidad los requisitos funcionales.
- Proporcionar una visión general del sistema.

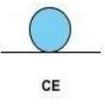
Durante el análisis, es importante realizar un estudio de los requisitos obtenidos, con el objetivo de tener una mejor comprensión antes de entrar al diseño de dicho software, garantizando así una arquitectura robusta, eficaz, eficiente y capaz de sobrevivir a cambios.

Diagrama de clases del análisis

En la construcción del modelo de análisis se deben identificar las clases que describen la realización de los casos de uso, los atributos y las relaciones entre ellas. Con esta información se construye el diagrama de clases del análisis, siendo el principal diagrama de análisis y diseño para un sistema. Este diagrama se representa mediante tres estereotipos:

Tabla 15-Estereotipos de las clases del análisis.

Estereotipo	Descripción
 CI	Clase Interfaz: Se encargan de la modelación de toda la interacción que puede existir entre los actores y el sistema.
 CC	Clase Controladora: Representan la coordinación, secuenciación, transacciones y a veces la lógica del negocio; se emplean a menudo para encapsular el control referido

	a un caso de uso.
	Clase Entidad: Se utilizan para modelar información que posee una vida larga y que es a menudo persistente. En la mayoría de los casos se derivan directamente de una clase entidad del negocio.

Diagramas de clases del análisis

A continuación se muestran los diagramas de clases de análisis correspondientes a los casos de uso descritos anteriormente:

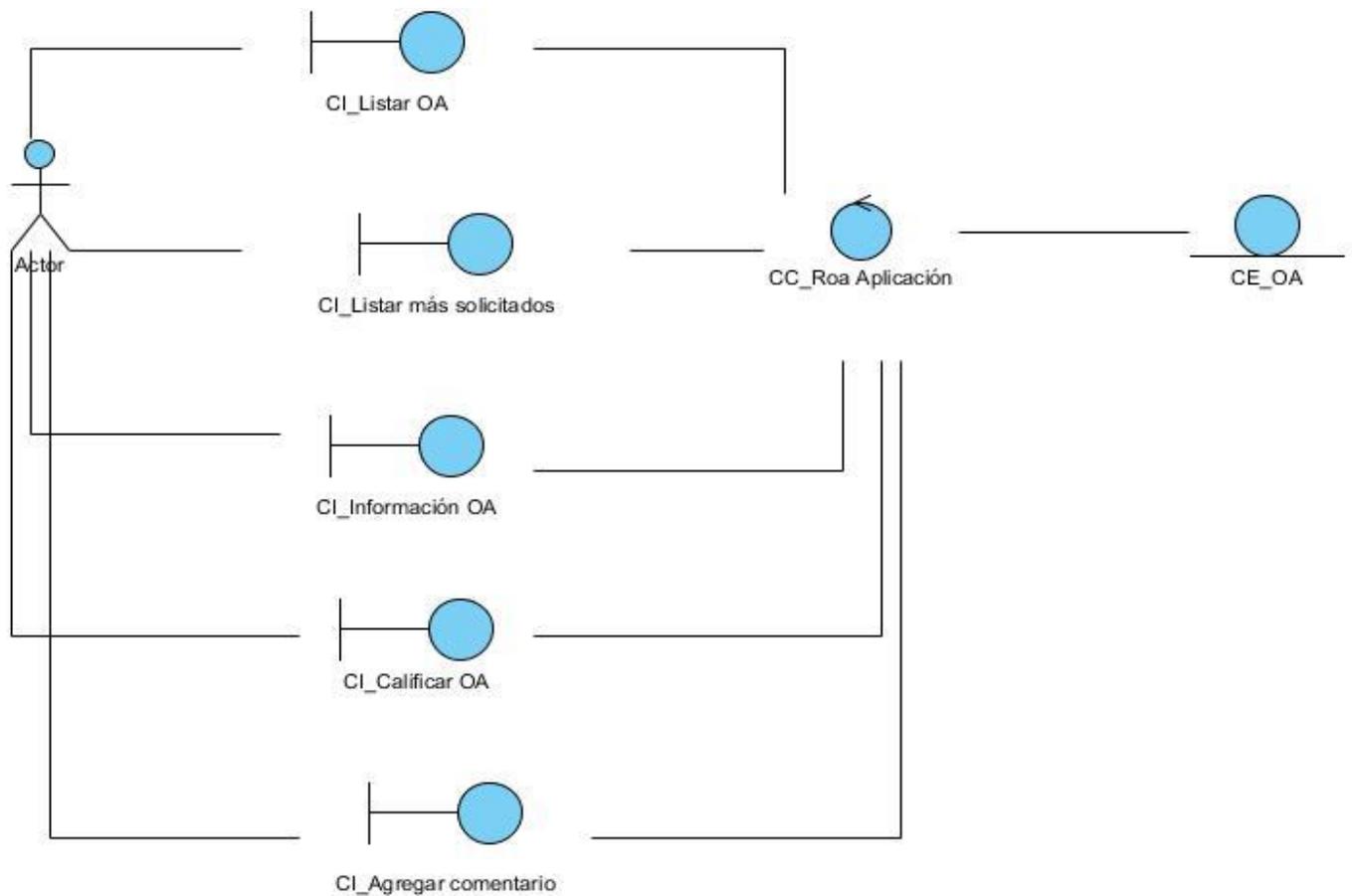


Figura 9-Diagrama de colaboración "Listar OA (más solicitados)".

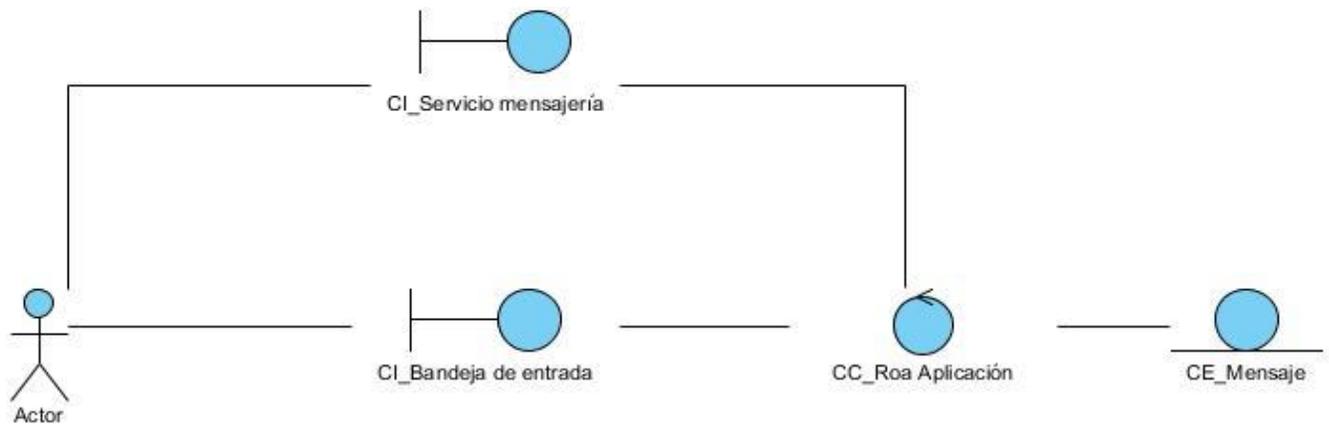


Figura 10-Diagrama de clases del análisis "Servicio de mensajería (Bandeja de entrada)".

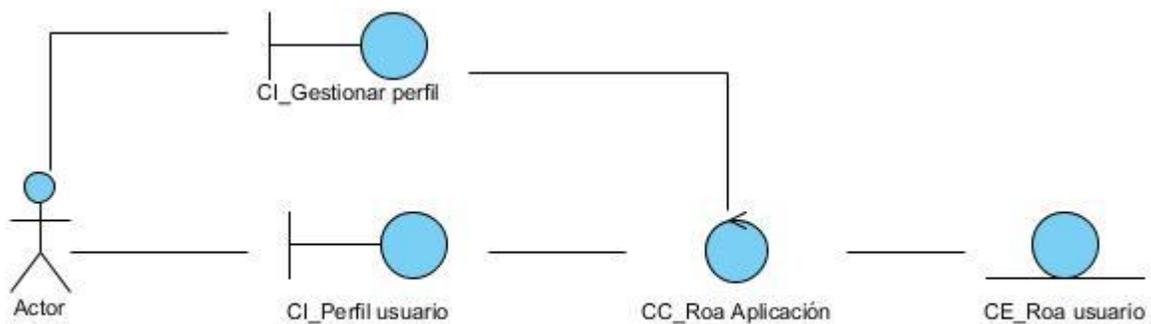


Figura 11-Diagrama de clases del análisis "Gestionar perfil del usuario".

Ver en el Anexo (Anexo 2) las figuras de los diagramas de clases del análisis.

3.2 Diagrama de interacción

Los diagramas de interacción muestran cómo se comunican los objetos. Estos objetos interactúan para realizar colectivamente los servicios ofrecidos por las aplicaciones. Existen dos tipos de diagramas de interacción: los diagramas de secuencia y los diagramas de colaboración que son los que se ven en este punto.

Los Diagramas de Colaboración se prestan al descubrimiento de abstracciones pues permite representar los objetos en una disposición próxima a la realidad. En esencia, su misión es localizar el comportamiento de los objetos.

En el diseño en ocasiones es preferible representar la interacción entre los objetos o subsistemas con los diagramas de secuencia ya que el centro de atención principal es el encontrar secuencias de iteraciones detalladas y ordenadas en el tiempo (46).

A continuación se muestran los diagramas de colaboración y secuencia correspondientes a los casos de usos descritos:

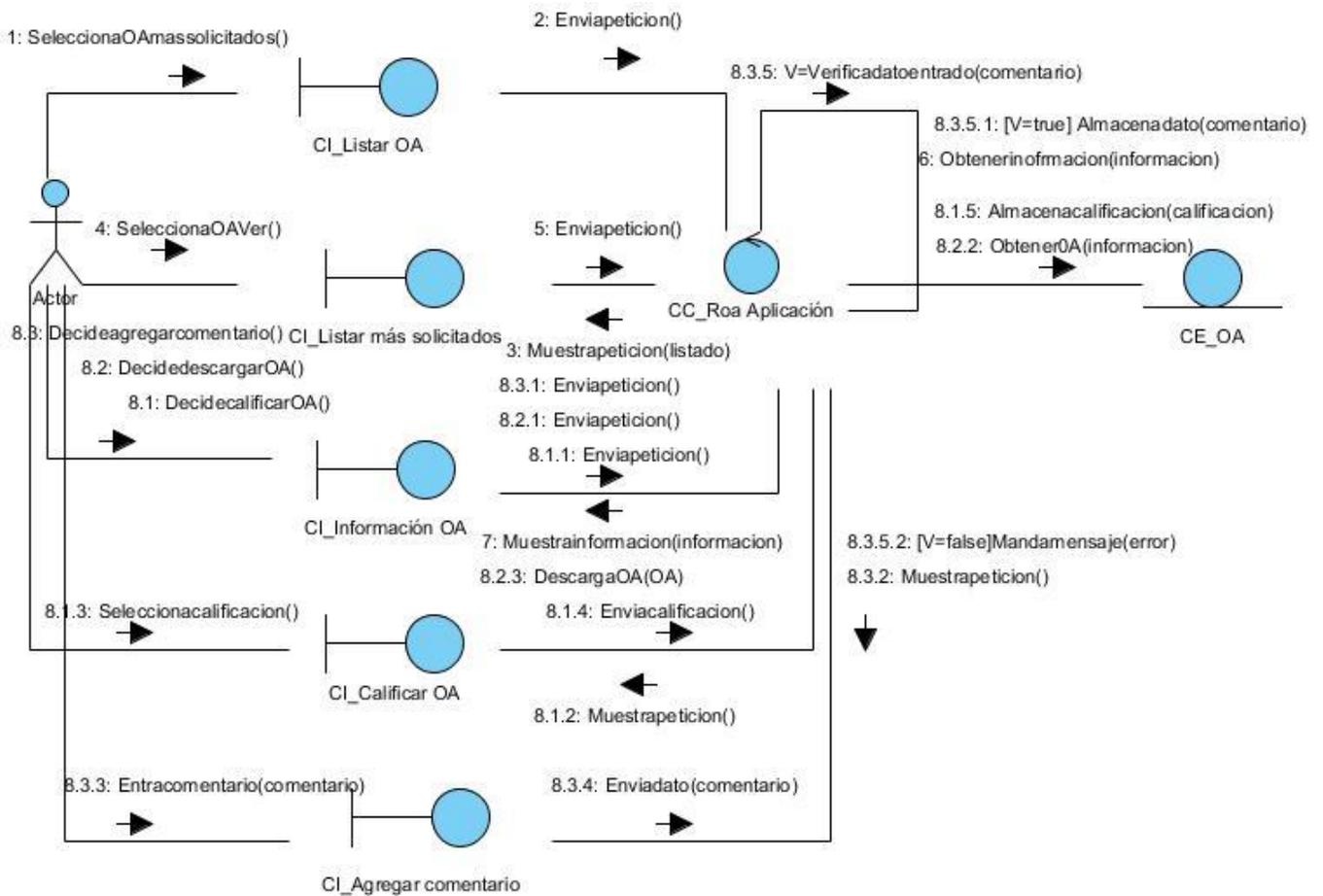


Figura 12-Diagrama de colaboración "Listar OA (más solicitados)".

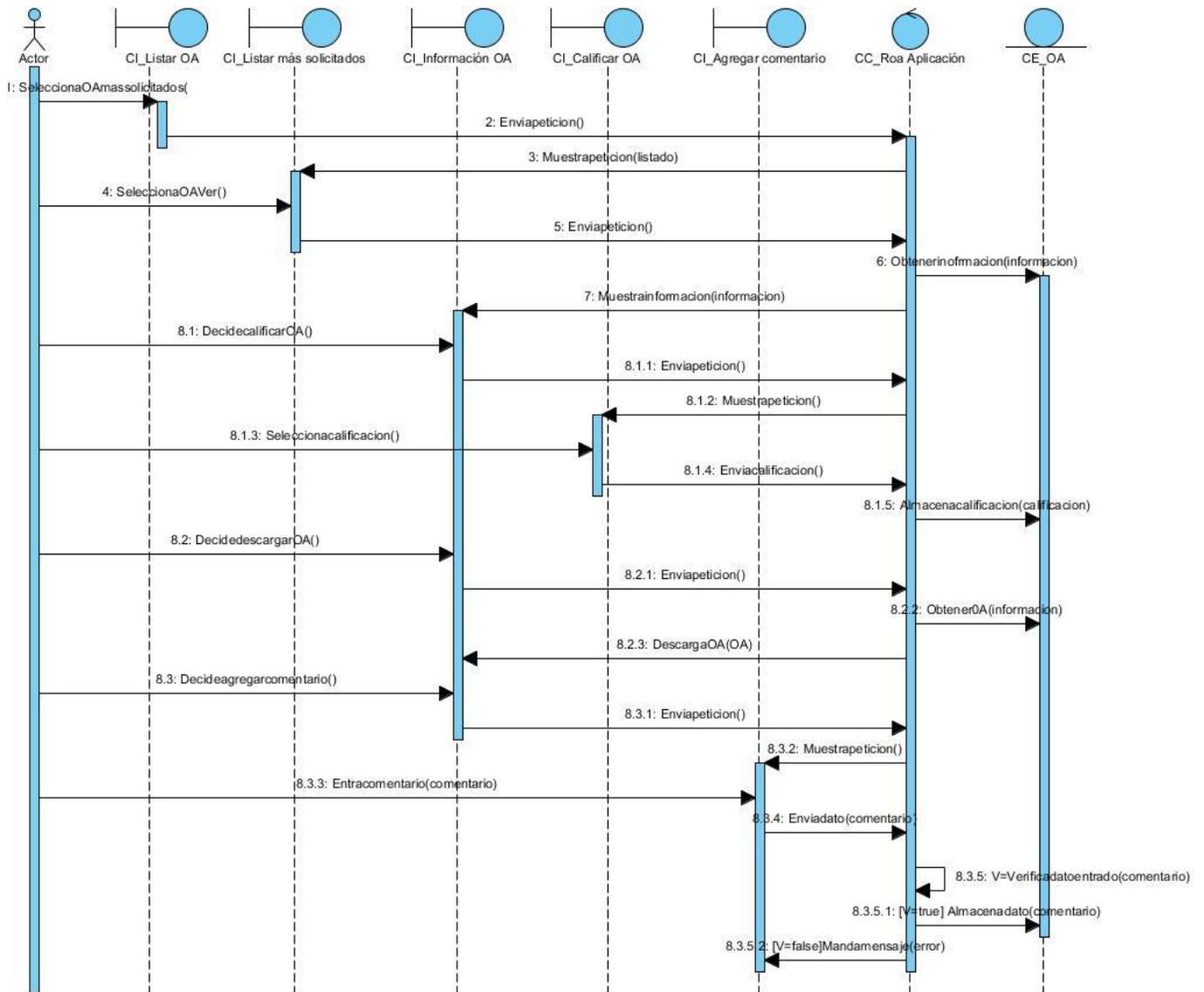


Figura 13-Diagrama de secuencia "Listar OA (más solicitados)".

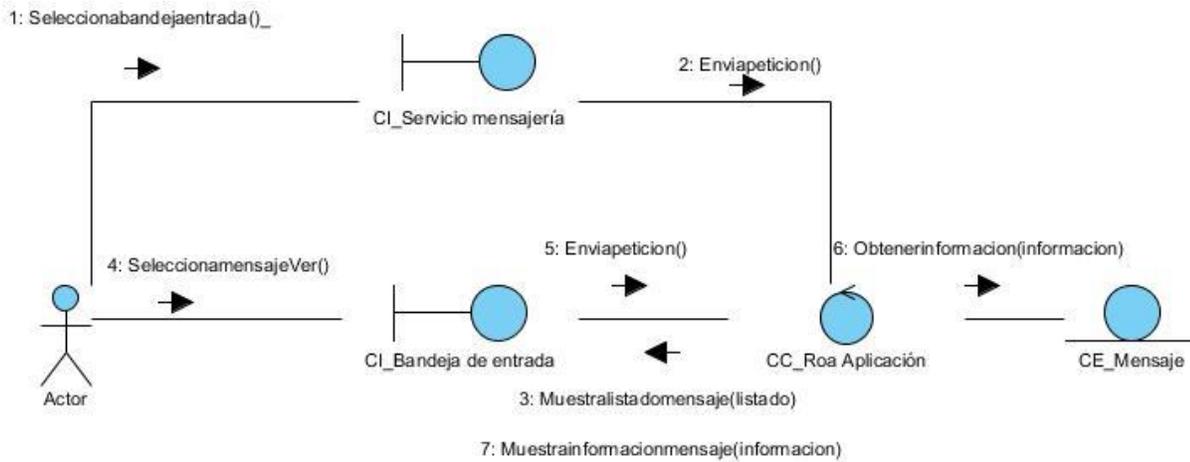


Figura 14-Diagrama de colaboración "Servicio de mensajería (Bandeja de entrada)".

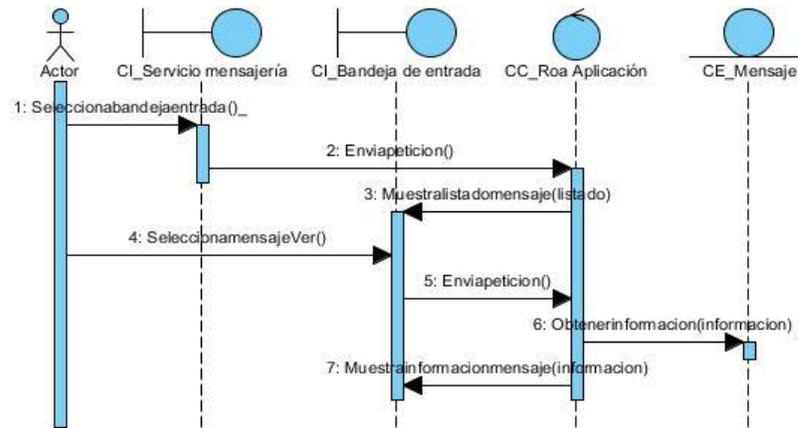


Figura 15-Diagrama de secuencia "Servicio de mensajería (Bandeja de entrada)".

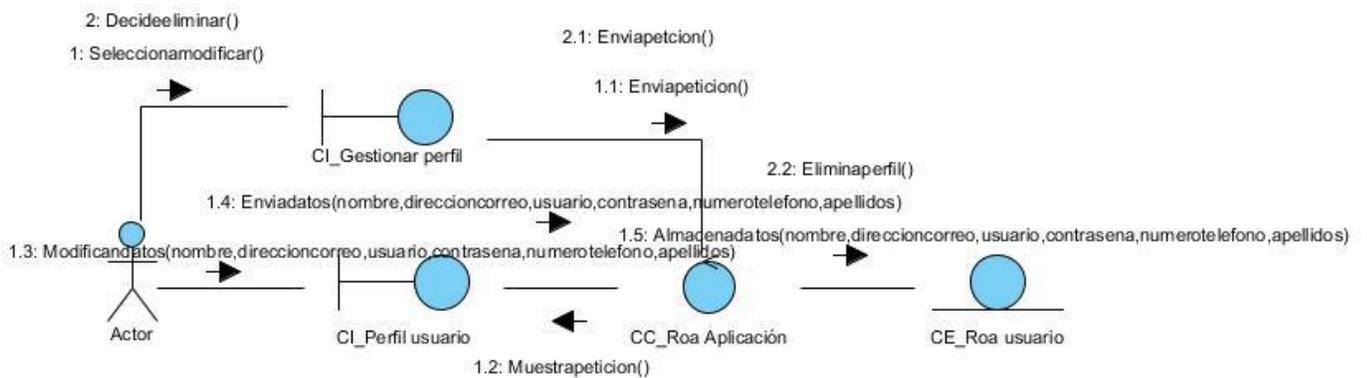


Figura 16-Diagrama de colaboración "Gestionar perfil del usuario".

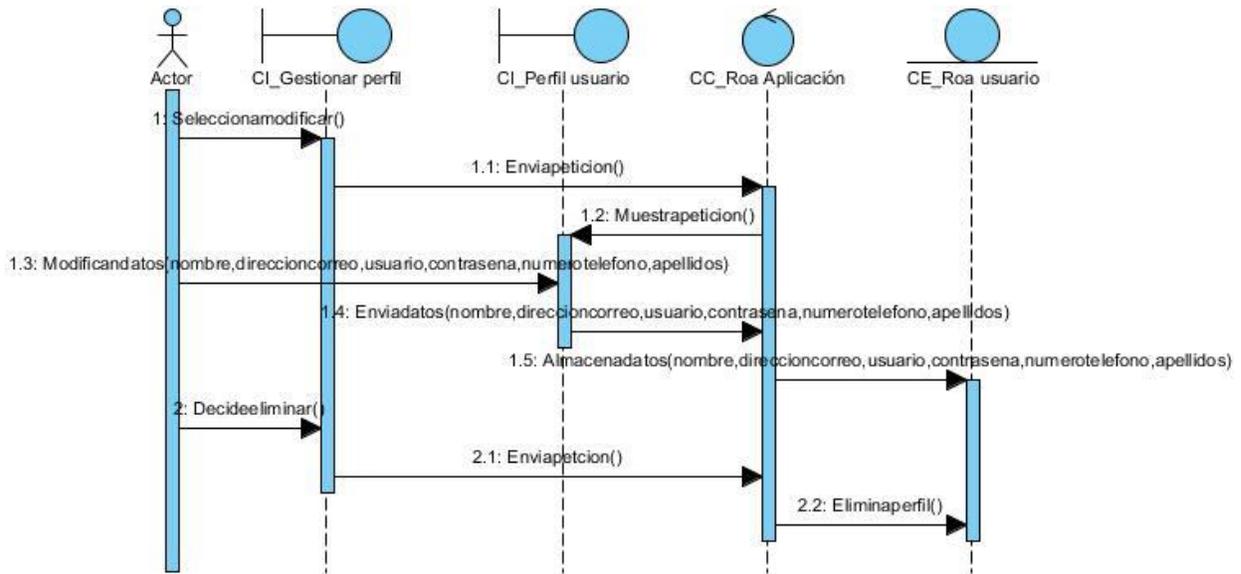


Figura 17-Diagrama de secuencia "Gestionar perfil del usuario".

Ver el Anexo (Anexo 3) las figuras de los diagramas de colaboración.

Ver el Anexo (Anexo 4) las figuras de los diagramas de secuencias.

3.3 Patrones arquitectónicos

Una arquitectura de software diseñada en N-Capas consiste en la definición de niveles de abstracción, los cuales tienen una función específica permitiendo un diseño modular. Ello permite que sean creados sistemas con un bajo acoplamiento entre sus módulos o componentes. Una variante de este patrón muy utilizada es la de Tres Capas, mediante la cual se fracciona el sistema informático en la capa de presentación, la capa de lógica del negocio y la capa de acceso a datos. Una restricción de este patrón consiste, en que las capas inferiores no deben conocer ni hacer llamadas a procedimientos implementados en capas superiores, sino que las funcionalidades que ellas ofrecen, son accedidas desde niveles mayores.

La capa de presentación es la que se encarga de interactuar con el usuario mediante la interfaz de usuario. La capa lógica del negocio, llamada también como lógica de la aplicación, se encarga de realizar las tareas para las cuales está concebido el sistema. Es implementada utilizando un modelo orientado a objetos del dominio de la aplicación. Se encarga de controlar las operaciones de acuerdo con las reglas del negocio. La capa de acceso a datos es la que gestiona el almacenamiento de los datos, ya sea en una base de datos o en un fichero, así como la consulta a los mismos.

Ventajas de la arquitectura de tres capas:

- Fácil escalabilidad.
- Independencia de plataforma.
- Fácil comunicación.
- Reutilización del código.

Para el desarrollo de la solución que se presenta en este trabajo, se adoptará una arquitectura de tres capas. Debido a sus potencialidades para la reutilización y el aprovechamiento de los beneficios que brinda, además se tuvo en cuenta que muestra una clara separación entre las funciones desplegadas en capas. También permite la estandarización y proporciona una distribución clara del trabajo entre los miembros de un equipo de desarrollo.

En cambio, el término "nivel" corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física, ejemplo: una solución de tres capas que residen en un solo ordenador (presentación + negocio + datos) se dice que la arquitectura de la presentación es de tres capas y un nivel. Una solución de tres capas que residen en dos ordenadores (presentación + lógica por un lado; lógica + datos por el otro lado) se dice que la arquitectura de la solución es de tres capas y dos niveles.

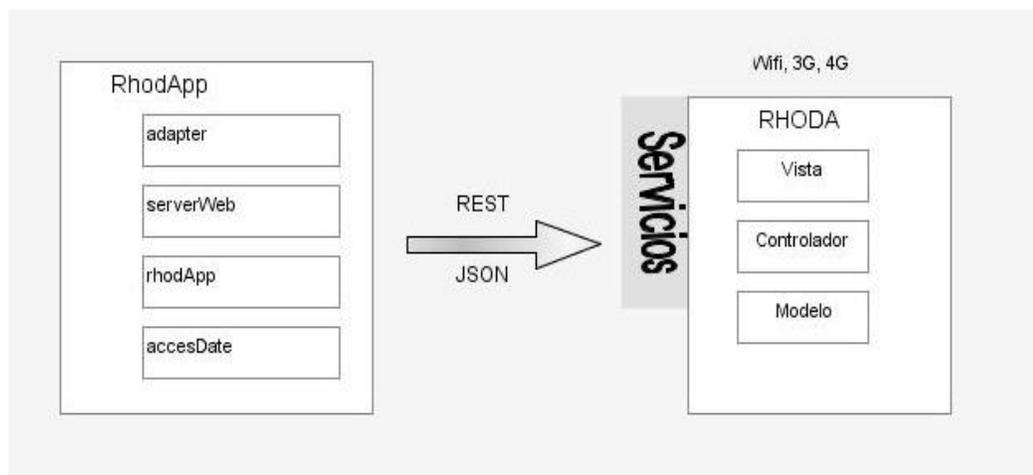


Figura 18-Arquitectura de la aplicación.

RhodApp: Contiene la clase controladora que es la encargada de vincular todos los paquetes. Este consume los servicios habilitados por RHODA mediante REST, utilizando JSON para la transmisión de datos.

Adaptador: Este paquete agrupa las clases que crean las clases en lo que el usuario espera ver.

AccesDate: Contiene las clases que modelan la base de datos de la aplicación.

ServerWebController: Contiene las clases que se encargan de la conexión con la aplicación RHODA y la ejecución de los servicios.

3.4 Patrones de diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referente al diseño de interacción o interfaces. Se utiliza para mostrar la existencia de clases y sus relaciones en la visión lógica de un sistema. Durante el análisis se utiliza para indicar las misiones y responsabilidades comunes de las entidades, que caracterizan el comportamiento de un sistema. Durante el diseño se utilizan para plasmar la estructura de las clases que forman la arquitectura del sistema (47). Los patrones utilizados para el diseño de la investigación son:

Alta cohesión: Aconseja mantener las clases lo más cohesionadas posible para controlar la complejidad de las mismas.

Experto: Soluciona el problema de asignar una responsabilidad de forma general, al recomendar para esta función a la clase que maneje la información necesaria.

Bajo acoplamiento: Recomienda asignar las responsabilidades de tal forma que se le brinde soporte a una mayor reutilización y poca dependencia. Cuando hay relación de dos o más clases, si se modificada una de ellas, no se afectan las demás.

Creador: Trata el problema de qué clase debe ser la encargada de instanciar a otra.

Controlador: Se encarga de toda la lógica del negocio.

Proxy: Permite la conexión con el repositorio RHODA para que capte los servicios.

Fuera de estas clasificaciones se estudiaron otros patrones de diseño que se pueden utilizar en cualquier aplicación móvil:

Adaptador: Este patrón se encarga de adaptar clases de interfaces incompatibles para que trabajen de manera conjunta. Un ejemplo claro es cuando se tiene que hacer uso de alguna librería, pero surge la necesidad de adaptarse a ella. Las opciones inmediatas son: o el actor se adapta a ella, o ella al actor. Lo mejor en este caso es crear un objeto que adapte tanto lo del actor a lo de ella, como lo de ella al actor.

Recordador: Representa y externaliza el estado interno de un objeto sin violar la encapsulación, de forma que éste puede volver a dicho estado más tarde. Se emplea para almacenar el contexto de las vistas junto con el estado de las variables en determinado momento, y que el usuario pueda volver a ese estado exacto cuando lo desee.

Navegación: La navegación es el proceso de ir de un lugar a otro. Tiene dos metas; en primer lugar, la navegación le permite al usuario saber qué información está disponible en la aplicación; y en segundo lugar, ayuda a los usuarios a encontrar la información deseada lo más rápido posible.

Social: Los patrones de diseño contenidos en esta categoría, le permiten al actor comunicarse e interactuar con otros actores. Por ejemplo, responder a cierto contenido a través de un comentario, hacer alguna valoración de algún contenido.

Notificación: Las notificaciones le dicen al usuario acerca de un evento que ocurre en la aplicación. Puede que en alguna de ellas se requiera de la respuesta del usuario, y en otras ocasiones la notificación simplemente suministra alguna información.

Interacción de pantalla: Las interacciones de pantalla incluyen todas las interacciones que puede realizar el usuario con su dedo sobre la pantalla.

3.5 Modelo del diseño

De acuerdo con lo que plantea la arquitectura en tres capas, la cual organiza las clases en tres paquetes, se organizaron las clases en la solución que plantea este trabajo de diploma. Cada uno de los paquetes se ha dividido a la vez en subpaquetes de acuerdo con las funcionalidades de las clases contenidas. Cada paquete representa una capa de la arquitectura y está construido sobre su predecesor. Las clases pertenecientes a una capa están relacionadas con las clases de la capa inmediata inferior. (Ver figura 19)

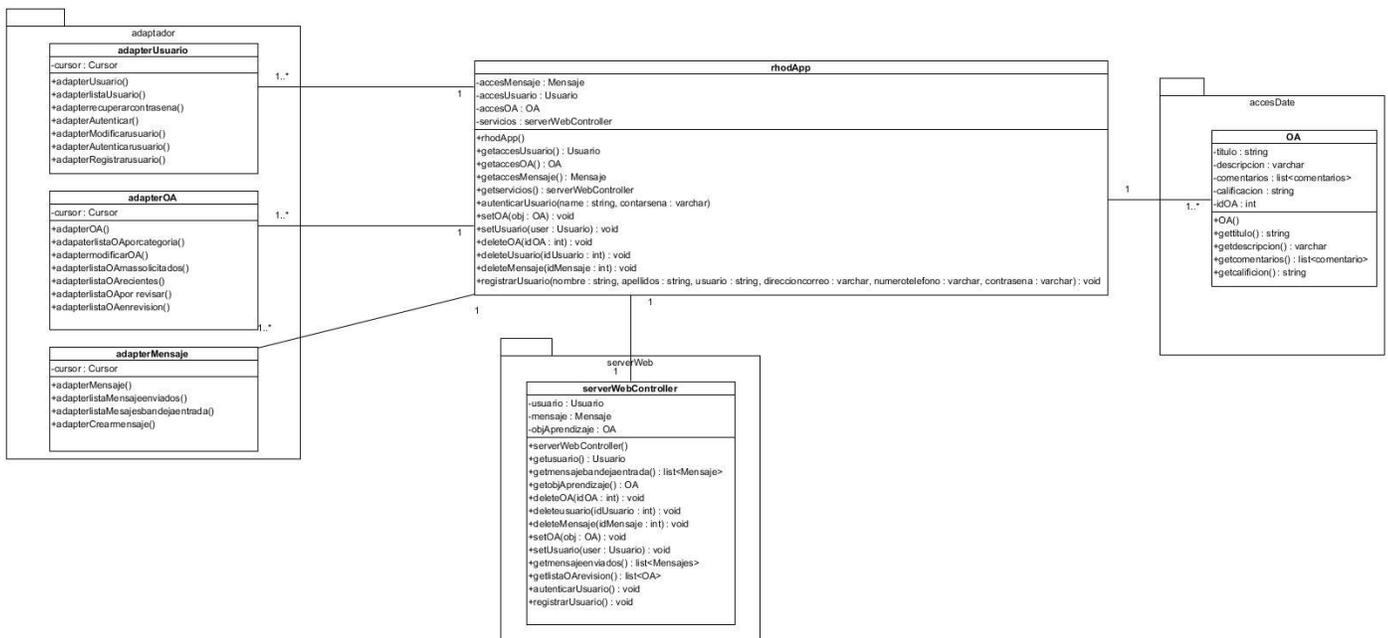


Figura 19-Diagrama de diseño.

3.6 Diagrama de componentes

Un diagrama de componente muestra cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes. Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables o paquetes. Los diagramas de Componentes prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura del sistema.

Debido a que los diagramas de componentes son más parecidos a los diagramas de casos de uso, éstos son utilizados para modelar la vista estática y dinámica de un sistema. Muestra la organización y las dependencias entre un conjunto de componentes. No es necesario que un diagrama incluya todos los componentes del sistema, normalmente se realizan por parte. Cada diagrama describe un apartado del sistema. (Ver figura 20)

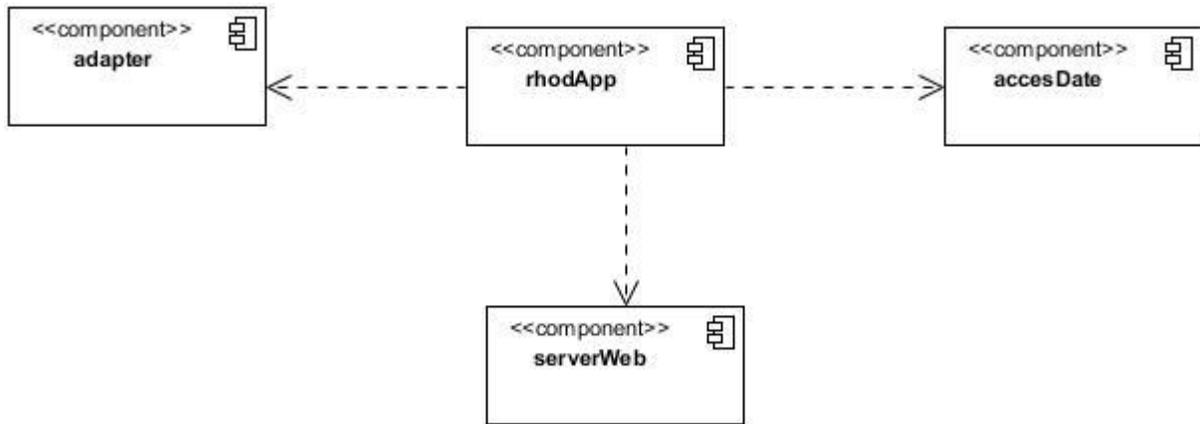


Figura 20-Diagrama de componentes.

3.7 Revisión de la solución propuesta

Uno de los procesos más importantes en el ciclo de vida de un software es el proceso de Verificación y Validación (V&V). Este es el nombre que se le da a los procesos de comprobación y análisis que asegura que el software que se desarrolla está acorde a su especificación y cumple las necesidades de los clientes. Inicia con las revisiones de los requerimientos y continúa con la de los casos de uso y termina con las revisiones del diseño y las inspecciones del código hasta la prueba del producto.

La utilización de determinados métodos como métricas en algunos de los artefactos generados permitirá identificar y erradicar errores en los requisitos y casos de uso.

Métricas

Las métricas son medidas generalmente absolutas pero si objetivas, que proporcionan una forma sistemática de valorar la calidad de los elementos basándose en un conjunto de reglas bien definidas. Tienen un papel decisivo en la obtención de un producto de alta calidad, porque determinan mediante estadísticas basadas en la experiencia, el avance del software y el cumplimiento de parámetros requeridos (48).

Se realizó un estudio sobre las siguientes métricas para sistemas orientados a objetos: Lorenz y Kid (1994) (49), Métricas R. Martín (1994) (50); y las métricas propuestas por Alan Davis (1993) (51) que corresponden a la calidad de la especificación de requisitos, la cual fue seleccionada para la validación de estos.

Métricas de la calidad de Especificación de los Requisitos

La métrica de la Calidad de Especificación de los Requisitos mide la especificidad de los requisitos, haciendo que la parte interesada pueda entenderlos de manera fácil y se puedan probar (52). Estas métricas incluyen las siguientes características (52):

Especificidad: Ausencia de ambigüedades en la especificación, facilitando los procesos de captura y procesamiento de requerimiento. Un requisito debe tener una única interpretación.

Corrección: El requerimiento debe representar algo requerido por el sistema a construirse, el grado de validación de los requerimientos mide la corrección en sus definiciones.

Compleción: Completitud de las definiciones.

Comprensión: Todo tipo de lector (clientes, usuarios, equipos de calidad, desarrolladores, gestores, entre otros) debe entender el requerimiento.

Consistencia interna: No existen subconjuntos de requisitos contradictorios.

Consistencia externa: Ninguno de los requerimientos está en contradicción con lo expresado en documentos de nivel superior.

Estabilidad: Ninguno de los requerimientos ha sido modificado en cuanto al indicador estabilidad.

En el presente trabajo se seleccionaron tres revisores teniendo en cuenta los criterios siguientes:

- Graduado de nivel superior.
- Tres años de experiencia como mínimo en el tema de la ingeniería de software.
- Conocimientos acerca de los indicadores de evaluación.
- Especialista en el rol de revisor.

Se tuvo en cuenta que si dos de los tres revisores clasifican en correcto (C) el requisito atendiendo a los indicadores, el resultado final será correcto, de igual manera se valoran los incorrectos (I). Para consultar el resultado emitido por los revisores ver el Anexo (**Anexo 5 Resultado de la validación de los requerimientos**).

Para aplicar la métrica, inicialmente fue necesario calcular el número total de requisitos.

$$R_t = R_f + R_{nf}$$

R_f : Número de requisitos funcionales.

R_{nf} : Número de requisitos no funcionales.

R_t : Total de requisitos.

$$R_t = 12 + 3 = 15$$

Especificidad

Para determinar la especificidad o falta de ambigüedad de los requisitos, se sugiere una métrica basada en la consistencia de la interpretación de los revisores de cada requisito (48).

En la evaluación de la especificidad de los requisitos cada revisor planteó lo que había interpretado por cada requisito y se pudo apreciar que las interpretaciones coincidían, demostrando que las especificaciones de los requisitos presentan un alto grado de claridad, ya que responden a una única interpretación, no siendo necesario modificarlos.

El objetivo de esta métrica es cuantificar la especificidad o falta de ambigüedad en la definición de los requerimientos. La especificidad se calcula como:

$$Q_1 = R_{ij} / R_t$$

Donde R_{ij} es el número de requisitos que todos los revisores interpretaron igual. Cuanto más cercano esté el valor de Q_1 (grado de especificación de los requisitos) a 1, menor será la ambigüedad de la especificación.

Primera revisión

$$Q_1 = 13 / 15$$

$$Q_1 = 0.87$$

Resultado: Se detectó que el 12.5% de los requerimientos presentaban ambigüedades, por lo que se realizó una reelaboración de los requerimientos, con términos más entendibles por todos.

Segunda revisión

$$Q_1 = 15 / 15 = 1$$

Resultado: No se detectaron ambigüedades.

Compleción

El resultado de esta métrica representa el nivel de completitud de las definiciones de los requisitos y está siempre entre 0 y 1. Grado en que los requerimientos cumplen las necesidades de los usuarios. Cuanto más próximo este el valor de 1 mayor será el nivel de completitud.

$$Q_2 = R_c / (R_c + R_{pe})$$

Donde Q_2 es el grado de completitud de los requisitos, R_c es el número de requisitos completos y R_{pe} el número de requisitos pobremente especificados.

Primera revisión

$$Q_2 = R_c / (R_c + R_{pe})$$

$$Q_2 = 15 / (15 + 2)$$

$$Q_2 = 0.9$$

Resultado: Se detectó que el 12.5% de los requisitos definidos no estaban correctamente especificados, se realizó una reelaboración de estos requisitos y para la segunda revisión realizada no se registró ningún inconveniente.

Corrección

El grado de validación de los requerimientos siempre entre 0 y 1. El valor óptimo de esta métrica es el más cercano a 1 e indica un alto nivel de corrección en la definición de los requisitos. Este valor se calcula como se muestra a continuación:

$$Q_3 = (R_c - R_i) / R_c$$

Donde Q_3 es el grado de validación de los requisitos, R_c número de requisitos validados como correctos y R_i número de requisitos validados como incorrectos.

Primera revisión

$$Q_3 = (R_c - R_i) / R_c$$

$$Q_3 = (12 - 3) / 12$$

$$Q_3 = 0,8$$

Resultado: Se concluyó que el 18.7% de los requisitos no estaban correctamente definidos, se realizó una nueva descripción de estos requisitos arrojando un resultado satisfactorio.

Comprensión

La comprensión de los requisitos marca el índice de legibilidad, no existe dificultad en su entendimiento. El resultado de esta métrica está siempre entre 0 y 1. El valor óptimo es el más cercano a 1. Se determinó a partir de la relación que se muestra a continuación:

$$Q_4 = R_f / R_t$$

Siendo:

Q_4 : Nivel de comprensión.

R_f : Número de requisitos que todos los revisores entienden.

Sustituyendo en:

$$Q_4 = 11 / 15 = 0.75$$

Resultado: Se detectó que el 25% de los requerimientos presentaban problema de entendimiento, por lo que se realizó una reelaboración de los requerimientos, con términos más entendibles por todos.

Consistencia interna

El resultado de esta métrica está siempre entre 0 y 1. El valor óptimo de esta métrica es el más cercano a 1 y expresa que no existen subconjuntos de requisitos contradictorios (no hay conflictos). El valor óptimo de esta métrica es el más cercano a 1. Este valor se calcula como se muestra a continuación:

$$Q_4 = (R_e - R_c) / R_e$$

Donde Q_4 es el grado de consistencia interna de los requisitos, R_e es el número de requisitos especificados y R_c es el número de requisitos en conflictos con otros requisitos en la especificación.

$$Q_4 = (15 - 0) / 15 = 1$$

Resultado: No se hallaron requisitos en conflictos con otros.

Consistencia externa

El resultado de esta métrica está siempre entre 0 y 1. Expresa que ninguno de los requisitos está en contradicción con lo expresado en documentos de nivel superior. El valor óptimo es el más cercano a 1. Este valor se calcula:

$$Q_5 = R_{ec} / R_t$$

Donde Q_5 es el grado de consistencia externa de los requisitos, R_{ec} es el número de requisitos que son consistentes con otros documentos.

$$Q_5 = 15 / 15 = 1$$

Resultado: No existen en contradicción con documentos de nivel superior.

Estabilidad

Para medir la estabilidad de los requisitos de software, se aplicó la métrica propia para esto, la cual ofrece valores entre 0 y 1. El mejor valor es el más cercano a 1. La estabilidad de los requisitos se calcula de la siguiente forma:

$$E = (R_t - R_m) / R_t$$

Donde **E** es el valor de la estabilidad de los requisitos y **R_m** es el número de requisitos modificados.

Tabla 16-Clasificación de la estabilidad.

Clasificación Intervalo	
Alta	$0.90 \leq E \leq 1$
Media	$0.80 \leq E \leq 0.90$
Baja	$E \leq 0.80$

$$E = (R_t - R_m) / R_t$$

$$E = (15 - 1) / 15 = 1$$

Resultado: Se detectó que los requisitos no han sido modificados en cuanto al indicador estabilidad, por lo que la estabilidad se clasifica como alta.

3.6.2 Métricas de casos de uso

Para validar los casos de uso se realizó la técnica de las métricas, en la que se emplearon 4 atributos (53):

Completitud:

Un caso de uso es completo si especifica todo lo que deben hacer el actor y el sistema para alcanzar el objetivo del caso de uso y si se consideran todas las respuestas del sistema a situaciones anormales.

Para verificarlos se plantean las siguientes interrogantes:

P1. ¿Hay respuestas a todas las peticiones que el actor del caso de uso hace al sistema y viceversa?

P2. ¿Se contemplan todos los posibles escenarios para poder alcanzar el objetivo del caso de uso?

P3. ¿Se especifican todas las secuencias alternativas a las secuencias anormales?

P4. ¿Se contemplan todas las posibles excepciones a la secuencia normal?

Comprensibilidad

Un caso de uso es comprensible si todos los tipos de lectores pueden entenderlo fácilmente con una mínima explicación del autor. Para verificarlo se plantean las siguientes interrogantes:

P1. ¿Es posible leer el caso de uso sin volver atrás en repetidas ocasiones?

P2. ¿Es difícil seguir la secuencia normal del caso de uso por la presencia de las relaciones *include* o *extend*?

P3. ¿Es difícil seguir la secuencia de pasos por la existencia de demasiados pasos alternativos?

P4. ¿Se han desglosados demasiado los pasos de algún actor o del sistema provocando que el caso de uso avance a un ritmo muy lento?

P5. ¿Aparecen casos condicionales para expresar que el sistema comprueba una situación que permite el caso de uso continuar su realización?

Concisión

Un caso de uso es conciso si no incluye información innecesaria. Para verificarlo se plantean las siguientes interrogantes:

P1. ¿Podría el caso de uso ser expresado con menos palabras?

P2. ¿Existen elementos que se pueden obviar o aparecen anotaciones innecesarias y que dificultan la lectura del caso de uso?

P3. ¿Aparecen demasiadas interacciones entre el actor principal del caso de uso y otros elementos del entorno?

No trivialidad

Un caso de uso es no trivial si su secuencia de pasos conduce al actor a conseguir el objetivo que persigue la realización del caso de uso. Para verificarlo se plantean las siguientes interrogantes:

P1. ¿Expresa el nombre del caso de uso un objetivo de un usuario que el sistema debe implementar?

P2. ¿Conduce el caso de uso al actor a conseguir alguno de sus objetivos sin representar un conjunto de interacciones triviales?

Se tuvo en cuenta que si dos de los tres revisores clasifican en correcto (Si) a una de las interrogantes mencionadas anteriormente, el resultado final será correcto, de igual manera se valoran los incorrectos (No).

Compleitud

Resultado: Se concluyó que el 75% de los casos de uso presentaban completitud. Para consultar el resultado emitido por los revisores ver el Anexo (**Anexo 6 Resultados de la validación de casos de uso**).

Comprensibilidad

Resultado: Se concluyó que el 100% de los casos de uso son entendibles. Para consultar el resultado emitido por los revisores ver el Anexo (**Anexo 6 Resultados de la validación de casos de uso**).

Concisión

Resultado: Se concluyó que el 100% de los casos de uso no incluyen información innecesaria. Para consultar el resultado emitido por los revisores ver el Anexo (**Anexo 6 Resultados de la validación de casos de uso**).

No trivialidad

Resultado: Se concluyó que el 50% de los casos de uso no presentaban trivialidad. Para consultar el resultado emitido por los revisores ver el Anexo (**Anexo 6 Resultados de la validación de casos de uso**).

Al aplicar la Técnica de Métrica de Casos de Uso, se hizo necesario realizar dos revisiones ya que en la primera de esta se detectaron errores, en cuanto a no trivialidad y completitud.

En la segunda iteración, luego de haber corregido los errores encontrados en la primera revisión, no se detectaron casos de uso que arrojaran algún índice de error sobre algunos de los indicadores, por lo que los casos de uso estaban 100% correctos en cuanto a completitud, comprensibilidad, concisión y no trivialidad.

Conclusiones parciales

Como resultado del estudio realizado en este capítulo, el cual corresponde al flujo de trabajo de análisis y diseño, se identificaron todas las clases, se desarrollaron los diagramas de clases del análisis de los casos de uso del sistema, se confeccionó los diagramas de colaboración y los diagramas de secuencia para tener una mejor visión de la interacción de los objetos. Además, se elaboró el diagrama de clases del diseño definiendo los objetivos que debe cumplir el sistema. Se realizaron las pruebas a los artefactos de especificación de los requerimientos y casos de uso.

Conclusiones generales

Con el propósito de darle cumplimiento al objetivo general en la realización de esta investigación se profundizó en el conocimiento de las herramientas y tecnologías dirigidas al desarrollo de aplicaciones para móviles, utilizándose las mismas en la realización de la propuesta de solución.

Con los artefactos generados en los flujos de trabajo Requerimientos, Análisis y Diseño de la metodología RUP, se crearon las bases necesarias para la implementación de las funcionalidades para el desarrollo de la aplicación móvil para el repositorio RHODA.

Se validó la solución propuesta a través de las métricas de especificación de requerimientos y casos de uso, encontrándose varias ambigüedades las cuales fueron resueltas como parte del proceso.

Recomendaciones

Poner en práctica la propuesta planteada en el presente trabajo y realizar la implementación de la aplicación móvil para el repositorio RHODA para Android que permita interactuar con el repositorio desde cualquier lugar y cualquier momento.

Realizar nuevamente el diseño para los usuarios que tengan en su dispositivo móvil otro sistema operativo que no sea Android.

Referencias bibliográficas

1. **Alberto García, Alfredo Sánchez.** *Bibliotecas Digitales U-DL-A; Alta Tecnología al Servicio de la Educación.* <http://ict.udlap.mx/pubs/Biblioteca.pdf>.
2. **ING. Carlos Zapata Periche, Chiroque Juarez Florella.** *Modalidad de e-Learning y m-learning.* <http://www.calameo.com/books/0018026694f2907410871>.
3. **APOSTOL DARISSET, MELENDEZ GEHIDI, TIMAURE MARYELIS.** GUIA DE ACTIVIDADES EDUCATIVAS PARA APLICAR EL M – LEARNING A LOS ESTUDIANTES DE LA SECCIÓN “A” DEL SÉPTIMO GRADO DE EDUCACIÓN BÁSICA DE LA UNIDAD EDUCATIVA NACIONAL “EL PLAYÓN”. [En línea] JUNIO 2011. http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=10&ved=0CGQQFjAJ&url=http%3A%2F%2Ffdgen020informatica.wikispaces.com%2Ffile%2Fview%2FModificacion%2BProyecto%2BM%2B-%2BLearning.docx&ei=YacUU_iDYXRkQfO1YDACQ&usg=AFQjCNGLGfz1VShEFnReD9cvRv-Pt.
4. **Alejandra marota.** B- learning, E- learning M- learning . [En línea] <https://sites.google.com/site/trabajonumero2/miss-mirce-marroquin/b--learning-e--learning-m--learning/b--learning-e--learning-m--learning-1>.
5. **S.Coop, ISEA.** *Mobile learning, Análisis prospectivo de las potencialidades asociadas al Mobile Learning.* http://www.iseamcc.net/eISEA/Vigilancia_tecnologica/informe_4.pdf.
6. **COMPUTACIONALES, INGENIERO EN SISTEMAS Lizandro Alexander Fierro.** *APLICACIÓN MULTIPLATAFORMA PARA EL APOYO EN EL APRENDIZAJE DEL IDIOMA EXTRANJERO.* 2013. <http://repositorio.ug.edu.ec/bitstream/redug/2540/1/tesis%20V%205.0.pdf>.
7. **Marcela Chiarani, Irma Pianucci, Berta Garcia, Hugo Viano, Guillermo Leguizamón.** *Integración de herramientas para la administración de OA.* http://www.dirinfo.unsl.edu.ar/profesorado/PagProy/articulos/Wicc07_148.pdf.
8. **Committee:, IEEE Learning Technology Standards.** *Draft Standard for Learning Object Metadata.* 15 julio 2002. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf.
9. **Objects, The Instructional Use of Learning.** *Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy.* In D. A. Wiley (2000), *The Instructional Use of Learning Objects: Online Version.* <http://reusability.org/read/chapters/wiley.doc>.
10. **Roxana Canizares Gonzáles, Dunia Maria Colome Cedeno, Javier Soler Martín, Orlando Felipe Salvador Broche, Daymi Tamayo Avila, Davis Leyva, Sergio Bello González, David Velazque Escalona, Yailen San Juan, Yailin Romero Rodríguez.** *Repositorio de Objetos de Aprendizaje (RHODA 1.0 Y 1.2).* http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=10&ved=0CGsQFjAJ&url=http%3A%2F%2Fpublicaciones.uci.cu%2Findex.php%2FSC%2Farticle%2Fdownload%2F445%2F264&ei=zEP5Uo2EOqXp0wHJsYHoAQ&usg=AFQjCNHwi1KucjORhMTD_RM3ZD5N48geLg&bvm=bv.60983673,d.dm.

11. *Repositorio de Objetos de Aprendizaje.*

<http://www.roa.unp.edu.ar:8080/graduate/bitstream/123456789/76/1/informe%2023%20feb%20final.pdf>.

12. **Penalvo, Clara López Guzmán y Francisco J. García.** *Repositorios de Objetos de Aprendizaje: bibliotecas para compartir y reutilizar recursos en los entornos e-learning.*

<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CDkQFjAC&url=http%3A%2F%2Frevistas.unam.mx%2Findex.php%2Frbu%2Farticle%2Fdownload%2F24994%2F23432&ei=Q9MUU8jQNsqfKaf3ioHwAw&usg=AFQjCNHGRFm7CjYOJwbDjOprowWvemRANQ&bvm=bv.61965928,d.eW0>.

13. **C, Gloria Elena Vásquez.** *Primer Curso sobre Escritura Científica y Repositorios de Acceso Abierto.* 2014.

http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CDQqFjAC&url=http%3A%2F%2Fwww.necobelac.eu%2Fdocuments%2FPresentacion%2520No.%252010%2520Repositorios%2520Necobelac.ppt&ei=X6ArU6PUKMPkQfK14HwCg&usg=AFQjCNGX6CGom_v_l31lSa0JovPQCIZViQ.

14. **ESCOBAR, CARLOS ALBERTO LÓPEZ.** *REPOSITORIOS DE OBJETOS DE APRENDIZAJE.* Universidad Interamericana para desarrollo : s.n. presentacionesmultimedia.wikispaces.com/file/.../REPOSITORIOS.docx.

15. **Slick, Sheila Arguello.** *Aplicaciones vs páginas web móviles .* 10/5/2013.

<http://www.confidencial.com.ni/articulo/11718/aplicaciones-vs-paginas-web-moviles>.

16. **Talen, Lance.** *tipo de aplicaciones moviles.* 2014.

17. **Barranco, Patricia Cabrera Muñoz, Ysauro González Neri, Carmen Castillo.** *Dispositivos móviles en la educación: Percepción de los usuarios sobre los dispositivos móviles como herramienta de aprendizaje.* Junio del 2012.

http://educoas.org/portal/la_educacion_digital/147/pdf/EAP_ILCE_EN.pdf.

18. *La Nuevas Tecnologías de la Información y la Comunicación aplicadas a la Formación Continua en el mundo de habla hispana.* . http://www.oei.es/pdfs/entornos_virtuales_aprendizaje_formacion_continua.pdf.

19. **Sánchez, María.** Celulares modernos y smartphone. [En línea] 27 de enero de 2014. [Citado el: 20 de marzo de 2014.] <http://www.informaticamoderna.com/Celular.htm>.

20. **Petrazzini, Grabiél Osmar Pedroso.** *Sistemas operativos en dispositivos móviles.* 2012.

http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Sistemas_Operativos_en_Dispositivos_Moviles.pdf.

21. **Corredera, José R. Casar.** *Tecnologías y Servicios para la Sociedad de la Información.*

http://www.upm.es/sfs/Rectorado/Organos%20de%20Gobierno/Consejo%20Social/Actividades/tecnologias_servicios_para_sociedad_informacion.pdf.

22. **Arcia, Sergio.** *Sistema operativo.*

<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&ved=0CDwQFjAD&url=http%3A%2F%2Fudo->

informatica.wikispaces.com%2Ffile%2Fview%2FCopia%2Bde%2Bwikispace_sin%2Bimagenes.docx&ei=IFAXU8qMEsSkkQfXtoHwAg&usg=AFQjCNE41sZ5vE8aOo0Wam-aFzvZKTYAcg&bvm.

23. **Oleaga, Jon.** *Android, IOS, Windows Phone, Firefox OS ¿Cuál es mejor?*

http://www.abc.es/tecnologia/informatica-software/20130227/abci-firefox-android-windows-201302270909_1.html.

24. **Oscar Sánchez Cortés, Jose Romero Durán.** *Aplicación móvil auxiliar en el proceso de inscripción de la UAM-C.* 2011.

<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CCoQFjAA&url=http%3A%2F%2Fbio.izt.uam.mx%2F~cavanos%2FOsKaR%2Finscripciones%2520movil.doc&ei=MXBqU4rnE-WxsATD34DACA&usg=AFQjCNFiSCZzCrnhghzLX5XSUHZ8B8i8CQ&cad=rja>.

25. **Juan Manuel Fernández Luna .** Descripción de J2ME - Origen y evolución . *Versión: 2.0, Septiembre, 2006.* [En línea] http://leo.ugr.es/J2ME/INTRO/intro_2.htm.

26. Versiones y distribuciones Java: J2SE, J2EE, J2ME. ¿Java 7, Java 8, Java9..., cuál es mejor? (CU00606B). [En línea] http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=377:versiones-y-distribuciones-java-j2se-j2ee-j2me-ijava-7-java-8-java9-cual-es-mejor-cu00606b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188.

27. **Valencia, Universidad Politecnica de.** *Android: Programación de aplicaciones para móviles.*

<http://www.androidcurso.com/index.php/curso-android-basico/tutoriales-android-basico/31-unidad-1-vision-general-y-entorno-de-desarrollo/100-instalacion-del-entorno-de-desarrollo>.

28. *Guía Breve de Servicios Web.* <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>.

29. **Machuca, CAM.** *Estado del arte de servicios web.* www.methesis.fcs.ucr.ac.cr/blogs/rumana/wp-content/.../articulo2.pdf.

30. **Chanchí.** *Proccedings Template word.* 2011.

<http://seer.dev.ufrgs.br/index.php/cadernosdeinformatica/article/download/v6n1p233-240/11807>.

31. **Díaz, Merlín Milián.** *Estación de monitización especializada para video vigilancia Suria.*

http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CDAQFjAB&url=http%3A%2F%2Fpublicaciones.uci.cu%2Findex.php%2FSC%2Farticle%2Fdownload%2F1028%2F596&ei=BJwYU--1LJHqkQf23YHIBg&usg=AFQjCNHel9m1pQFWQepJgsVXwnAuN6dM_Q&bvm=bv.62577051,d.eW0.

32. *Metodologías Ágiles.* diescense.files.wordpress.com/2013/04/metodologc3adas-c3a1giles.pptx.

33. *Metodologías de desarrollo de software.*

<http://latecladeescape.com/t/Metodolog%C3%ADas+de+desarrollo+del+software>.

34. **Letelier, Patricio Torres, Emilio A. Sánchez López.** *Metodologías ágiles en el desarrollo del software.* <http://issi.dsic.upv.es/archives/f-1069167248521/actas.pdf>.
35. **ADSI // FDO.**
<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=6&ved=0CEoQFjAF&url=http%3A%2F%2Fpulgaf.files.wordpress.com%2F2011%2F07%2Fadsii-fdo.docx&ei=4aMYU6exIZCqkQft0YGIBA&usg=AFQjCNE0dUyakvcxqit690DfSaeyQDdA8w&bvm=bv.62577051,d.eW0>.
36. **Tuesta, Ing. Martín.** *Sistema de información docente.* <https://www.google.com/cu/search?sclient=psy-ab&biw=734&bih=431&noj=1&q=Las+herramientas+CASE+%28Computer+Aided+Software+Engineering%29+son+un+conjunto+de+programas+y+ayudas+que+dan+asistencia+a+los+analistas%2C+ingenieros+de+software+y+desarrolladores+du>.
37. *Visual Paradigm.* <http://www.visual-paradigm.com/>.
38. **Aguero Jorge, Cornejo Anabella, Heredia Ana Gabriela, Pascal Alejandro.** *Base de datos y UML.* 2013.
<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&ved=0CDwQFjAD&url=http%3A%2F%2Fbibdigital.epn.edu.ec%2Fbitstream%2F15000%2F6623%2F1%2FManual%2520de%2520UML.doc&ei=DhkaU7nBOlaSkQfckYCoCQ&usg=AFQjCNHNqfzXEU0KC1N0fFT7XgXgWdhrJQ&bvm=bv.62578>.
39. *Modelo conceptual o modelo de dominio.* http://ldc.usb.ve/~martinez/cursos/ci3715/clase6_AJ2010.pdf.
40. **Hall, Prentice.** *Modelo del dominio.* <http://is.ls.fi.upm.es/docencia/is2/documentacion/ModeloDominio.pdf>.
41. **Silvia Mónica Aranguren, Marcelos Estayno.** *Técnicas para una mejor ingeniería de Requerimientos. Nuevas estrategias.* http://www.actiweb.es/upa_padilla2050/archivo3.pdf.
42. **Hernández, A.** *Requisitos a partir del modelo de negocio.*
<http://rii.cujae.edu.cu/index.php/revistaind/article/viewFile/132/117>.
43. *El modelo de caso de uso.* http://www.sparxsystems.com.ar/resources/tutorial/use_case_model.html.
44. *Estimación por casos de uso.*
<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&ved=0CD0QFjAD&url=http%3A%2F%2Fxfsoft-proyecto-cenfotec.googlecode.com%2Fsvn%2Ftrunk%2FDocumentaci%25C3%25B3n%2FDocumentos%2520de%2520proyecto%2Fmiembros%2FCarlos%2520Retana%2FPresentacion%2>.
45. *Desarrollo del software en el área de aplicaciones web.*
<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CDQQFjAC&url=http%3A%2F%2Fwww.itchihuahuaii.edu.mx%2FServTec%2FDocsAlumnos%2FResidencias%2FResidenciaEnfoqueWeb.doc&ei=OysaU-adJlrWkQeM54CoDA&usg=AFQjCNFSYAJhm4YhfPbZ-39Ad49q8USSUg&bvm>.
46. **G.Booch, I. Jacobson and.** *El proceso Unificado de desarrollo de software.* 2000.

47. *Programación Orientada a Objetos, conceptos y características.*

<http://www.google.com.cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=5&ved=0CEQQFjAE&url=http%3A%2F%2Fwww.itescam.edu.mx%2Fprincipal%2Fsylabus%2Fpdb%2Frecursos%2Fr232.DOC&ei=1zlaU9KLJMy0kQet0oCICw&usg=AFQjCNGUHQE7XjWl56bj506lbH9u13kEzg&bvm=bv.62578216,d.eW0>.

48. **Pressman.** *Métricas del software.*

http://elmasterdelaweb.wikispaces.com/file/view/T033_036_Metricas.pdf/160297501/T033_036_Metricas.pdf.

49. *Taxonomía de métricas orientadas a objetos.*

http://www.google.com.cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CDAQFjAB&url=http%3A%2F%2Fwww.researchgate.net%2Fpublication%2F39698368_Mtricas_orientadas_a_objetos%2Ffile%2F9fcd50b62c1618d2f.pdf&ei=lzcaU_SSBMaqkAeyjICIDw&usg=AFQjCNGA4xFaV4mWDJkAW.

50. **Sánchez, A. Gargías.** *Evaluación de métricas de calidad del software.*

http://eprints.ucm.es/11487/1/Proyecto_Fin_de_M%C3%A1ster.pdf.

51. *Métricas técnicas del software.* <http://www.uv.mx/personal/asumano/files/2012/08/MetricasTecnicas.pdf>.

52. **S, Pressman. R.** *Ingeniería del software: Un enfoque práctico.*

53. **COVELLA, Guillermo Juan.** *Medición y Evaluación de Calidad en Uso de Aplicaciones Web .*

http://sedici.unlp.edu.ar/bitstream/handle/10915/4082/Documento_completo.pdf%3Fsequence%3D1.

54. **System, Popkin Software and.** *Modelo del sistema como UML.* <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/doc-modelado-sistemas-uml.pdf>.

55. **Carmen Janeth Padron.** *ESTRATEGIAS DIDÁCTICAS BASADA EN APLICACIONES DE M-learning.* [En línea]

<http://docentesinnovadores.net/Contenidos/Ver/6647>.

56. *Las nuevas Tecnologías de la Información y la Comunicación aplicadas a la Formación Continua en el mundo de habla hispana.* http://www.oei.es/pdfs/entornos_virtuales_aprendizaje_formacion_continua.pdf.

Glosario de términos

Aplicación: En informática, una aplicación es un tipo de software diseñado para facilitar al usuario la realización de una determinada tarea o trabajo.

Accesibilidad: capacidad que hace identificable y ubicable cuando se necesite al contenido de un recurso, para los requerimientos formativos necesarios, que se conozca su adecuación a los objetivos sin necesidad de obtener el propio contenido o pagar derechos por él, mediante la provisión de información suficiente.

ActionBar: Es un nuevo elemento en la interfaz de usuario de Android llamado también como barra de acciones. Situada en la parte superior de la pantalla, creada para que el usuario tuviera una experiencia unificada a través de las distintas aplicaciones.

Confiabilidad: Capacidad de un producto de realizar su función de la manera prevista. De otra forma, la confiabilidad se puede definir también como la probabilidad en que un producto realizará su función prevista sin incidentes por un período de tiempo especificado y bajo condiciones indicadas.

CSS: Las hojas de estilo en cascada (Cascading Style Sheets) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML (Lenguaje extensible de Marcado de Hipertexto)). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

E-learning: Enseñanza a distancia caracterizada por una separación física entre profesores y alumnos, donde se usa preferiblemente Internet como medio de comunicación y de distribución del conocimiento.

Estándares: Los estándares son acuerdos (normas) documentados que contienen especificaciones técnicas u otros criterios precisos para ser usados consistentemente como reglas, guías, o definiciones de características. Para asegurar que los materiales productos, procesos y servicios se ajusten a su propósito.

Framework: Es una estructura de soporte definida, representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

HTML: Lenguaje de Marcas de Hipertexto (HyperText Markup Language). Se usa para describir la estructura y el contenido en forma de texto, así como completar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede

describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores Web y otros procesadores de HTML.

Herramienta CASE: Ingeniería de software asistida por ordenador. Varias aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo, reduciendo el costo de las mismas en términos de tiempo y de dinero.

Internet: Es un conjunto descentralizado de redes de comunicación interconectadas, que utilizan la familia de protocolos TCP/IP, posee alcance mundial, constituye una red mundial de comunicación.

Interfaz de usuario: Es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo.

Interoperabilidad: capacidad de las plataformas para comunicarse e integrarse de una forma efectiva aun siendo diferentes.

Lenguaje de marca: Es un conjunto de reglas que utilizan una sintaxis predefinida que sirven para estructurar un documento electrónico, utilizando etiquetas al principio y al final de un elemento.

MSN Messenger: Messenger, popularmente conocido como MSN, es un servicio de mensajería instantánea que actualmente es uno de los más populares.

Navegador: Un navegador web o cliente HTTP, es un programa que permite interpretar la información y el código que contiene una página web (esté alojada en un servidor dentro de la World Wide Web o en uno local) y presentarla de manera legible a los usuarios.

Protocolo: En informática, un protocolo es un conjunto de reglas usadas para computadoras para comunicarse unas con otras a través de una red por medio de intercambio de mensajes.

Plugin: se trata de un pequeño programa que proporciona alguna funcionalidad específica a otra aplicación mayor o más compleja.

Reutilización: capacidad de los OA para integrarse con otros formando una sola entidad. En términos de reutilización de metadatos se refiere a la capacidad de reutilizar los metadatos de un recurso para describir otro.

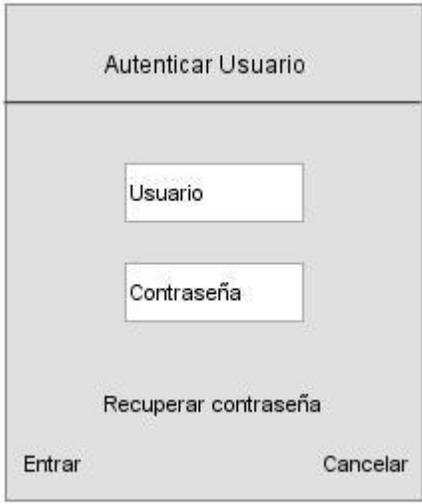
Script: Es un guion o conjunto de instrucciones. Permiten la automatización de tareas creando pequeñas utilidades. Es un trozo de código que puede recibir argumentos y devolver un valor. Los scripts se utilizan para generar código repetido y así ahorrar espacio y ganar velocidad.

World Wide Web: En informática World Wide Web (o la “Web”) o Red Global Mundial es un sistema de documentos de hipertexto accesibles a través de Internet. Con un navegador Web, el usuario puede visualizar las páginas web y navegar a través de ellas mediante hiperenlaces.

Anexo 1. Descripción de los casos de uso.

Caso de uso “Autenticar usuario”

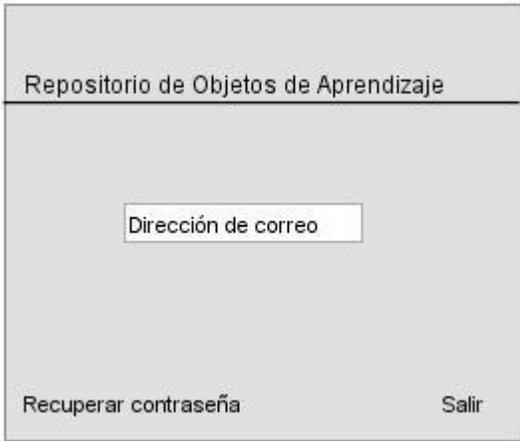
Flujo Normal de Eventos	
Sección: “Autenticar Usuario”	
Acción del Actor	Respuesta del Sistema
1. El usuario autenticado, administrador o revisor, decide acceder a la aplicación.	
	<p>1.1 Muestra una pantalla que permite introducir los siguientes datos:</p> <ul style="list-style-type: none"> a. Usuario b. Contraseña <p>Permite opciones:</p> <p>Recuperar contraseña o usuario olvidado (ver caso de uso: Recuperar contraseña olvidada).</p> <p>Cancelar.</p>
2. El usuario autenticado, administrador o revisor, entra sus datos y seleccionan entrar.	
	<p>2.1 Verifica que los datos entrados sean correctos.</p> <p>2.2 Guarda los datos del usuario automáticamente para un futuro acceso a la plataforma.</p> <p>2.3 Muestra la página principal del sistema (ver caso de uso: Mostrar la portada del sistema).</p> <p>2.4 Termina el caso de uso.</p>
Flujo alterno “ Datos entrados incorrectos o usuario no pertenece a la UCI”	
Acción del Actor	Respuesta del Sistema

	Muestra mensaje de error. Termina el caso de uso.
Flujo alternativo " Cancelar"	
Acción del Actor	Respuesta del Sistema
	Salte de la vista actual y regresa a la interfaz externa de la aplicación (ver figura Mostrar interfaz externa de la aplicación que se encuentra al final de las descripciones de casos de uso). Termina el caso de uso.
Prototipo de interfaz	
	
Poscondiciones	Se autenticó el actor correctamente.

Caso de uso "Recuperar contraseña"

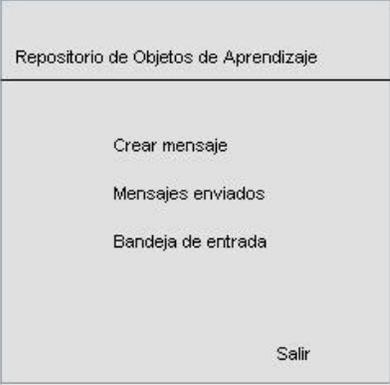
Flujo Normal de Eventos	
Sección " Recuperar contraseña olvidada "	
Acción del Actor	Respuesta del Sistema
1. El usuario autenticado, administrador o revisor selecciona recuperar contraseña.	

	Muestra una pantalla con campo de texto: Dirección del correo. Además permite: Salir.
2. El usuario autenticado, administrador o revisor entra la dirección de correo y selecciona recuperar.	
	2.2 Verifica que la dirección de correo es correcta. 2.5 Envía un mensaje con el dato perdido a la dirección de correo entrado por el actor y sale de la vista actual.
Flujo Alternativo: "salir"	
Acción del Actor	Respuesta del Sistema
	Sale de la vista actual y regresa a la interfaz externa de la aplicación (ver figura Mostrar interfaz externa de la aplicación que se encuentra al final de las descripciones de casos de uso).
Flujo Alternativo "El dato entrado no existe en la base de datos del sistema"	
Acción del Actor	Respuesta del Sistema
	Muestra mensaje de error. Termina el caso de uso.
Prototipo de interfaz	

	
Poscondiciones	Es recuperada la contraseña o usuario correctamente.

Caso de uso “Realizar Servicio de Mensajería”

Flujo Normal de Eventos	
Sección “Servicio de Mensajería”	
Acción del Actor	Respuesta del Sistema
1. El usuario autenticado, administrador o revisor selecciona la opción servicio de mensajería.	
	1.1. El sistema muestra una pantalla con las siguientes opciones: a. Crear mensaje (ver sección 1: crear mensaje). b. Bandeja de entrada (ver sección 2: bandeja de entrada). c. Mensajes enviados (ver sección 3: mensajes enviados). Además permite: Salir.
2. El usuario autenticado, administrador o revisor selecciona la opción a visualizar.	

	<p>El sistema muestra la opción especificada por el actor.</p> <p>Termina el caso de uso.</p>
Flujo Alternativo “Salir”	
Acción del Actor	Respuesta del Sistema
	<p>Regresa a la portada del sistema (ver caso de uso: Mostar portada del sistema).</p>
Prototipo de interfaz	
 <p>Repositorio de Objetos de Aprendizaje</p> <p>Crear mensaje</p> <p>Mensajes enviados</p> <p>Bandeja de entrada</p> <p>Salir</p>	
Flujo Normal de Eventos	
Sección 1 “Crear mensaje”	
Acción del Actor	Respuesta del Sistema
<p>1. El usuario autenticado, administrador o revisor escoge la opción crear mensaje.</p>	
	<p>1.1 Muestra una pantalla con los siguientes datos a entrar.</p> <p>Destinatario</p> <p>Descripción</p> <p>Además permite la opción de:</p> <p>Salir.</p>

<p>2. El usuario autenticado, administrador o revisor entra los datos y decide enviar mensaje.</p>	
	<p>2.1 Verifica si los campos están vacíos.</p> <p>2.2 Verifica si los datos entrados son correctos.</p> <p>Envía el mensaje y regresa al paso 1.1 del flujo normal de eventos de la sección servicio de mensajería.</p> <p>Termina el caso de uso.</p>
<p>Flujo Alternativo “Salir”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>Regresa al paso 1.1 del flujo normal de eventos de la sección servicio de mensajería.</p>
<p>Flujo Alternativo “Campo vacío”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>Muestra mensaje de error.</p> <p>El sistema regresa al paso 1.1 del flujo normal de eventos de la sección servicio de mensajería.</p> <p>Termina el caso de uso.</p>
<p>Flujo Alternativo “Datos incorrectos”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>Muestra mensaje de error.</p> <p>El sistema regresa al paso 1.1 del flujo normal de eventos de la sección servicio de mensajería.</p> <p>Termina el caso de uso.</p>
<p>Prototipo de interfaz</p>	

Repositorio de Objetos de Aprendizaje

Destinatario

Descripción del mensaje

Enviar Salir

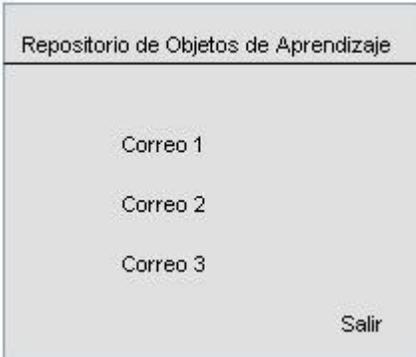
Flujo Normal de Eventos
Sección 2 “Bandeja de entrada”

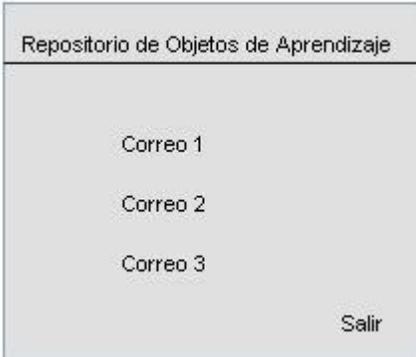
Acción del Actor	Respuesta del Sistema
1. El usuario autenticado, administrador o revisor decide ver bandeja de entrada.	
	1.1 Muestra una interfaz con un listado de los mensajes recibidos: Permite la opción: Salir.
2. El usuario autenticado, administrador o revisor selecciona el mensaje a leer.	
	2.1 Muestra la descripción del mensaje seleccionado y permite las opciones: eliminar y regresar.

Flujos Alternos

Flujos Alternos “Salir”

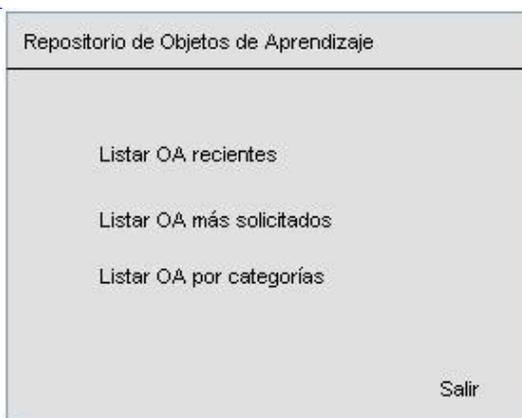
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de eventos de la sección servicio de mensajería.

	Termina el caso de uso.
Flujo Alternos “Eliminar”	
Acción del Actor	Respuesta del Sistema
	Elimina el mensaje seleccionado. Regresa al paso 1.1 del flujo normal de evento de la sección bandeja de entrada.
Flujos Alternos “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección bandeja de entrada.
Prototipo de interfaz	
	
Flujo Normal de Eventos	
Sección 3 “Mensajes enviados”	
Acción del Actor	Respuesta del Sistema
El usuario autenticado, administrador o revisor selecciona mensajes enviados.	
	Muestra una interfaz con un listado de los mensajes que han sido enviados. Permite la opción:

	Salir.
El usuario autenticado, administrador o revisor selecciona un mensaje a ver.	
	Muestra una interfaz con la descripción del mensaje y permite las opciones: eliminar y regresar.
Flujo Alternativo “Salir”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección servicio de mensajería.
Flujo Alternativo “Eliminar”	
Acción del Actor	Respuesta del Sistema
	3.1 Elimina el mensaje y regresa al paso 1.1 del flujo normal de evento de la sección mensajes enviados.
Flujo Alternativo “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección mensajes enviados.
Prototipo de interfaz	
 <p>Repositorio de Objetos de Aprendizaje</p> <p>Correo 1</p> <p>Correo 2</p> <p>Correo 3</p> <p>Salir</p>	
Poscondiciones	Se realizan los servicios que brinda la mensajería correctamente.

Caso de uso “Listar publicaciones OA”

Flujo Normal de Eventos	
Sección “Listar publicaciones OA”	
Acción del Actor	Respuesta del Sistema
El usuario anónimo, usuario autenticado, administrador o revisor selecciona la opción de listar OA.	
	<p>1.1. Muestra una interfaz con las siguientes opciones:</p> <p>Listar OA recientes (ver sección 1: listar OA más recientes).</p> <p>Listar OA más solicitados (ver sección 2 listar OA más solicitados).</p> <p>Listar OA por categorías (ver sección 3 listar OA por categorías).</p> <p>Permite la opción:</p> <p>Salir.</p>
Flujo Alternativo “Salir”	
Acción del Actor	Respuesta del Sistema
	<p>Regresa a la portada del sistema (Ver caso de uso: Mostrar la portada del sistema) en el caso de que este autenticado, sino regresa a la interfaz externa de la aplicación (ver figura Mostrar interfaz externa de la aplicación que se encuentra al final de las descripciones de casos de uso).</p> <p>Termina el caso de uso.</p>
Prototipo de interfaz	



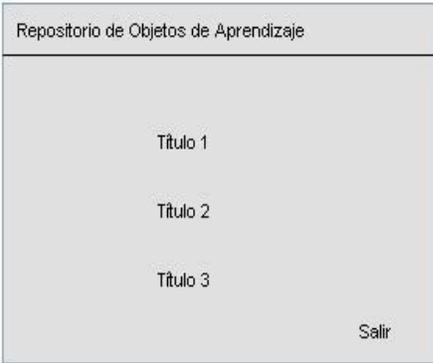
Flujo Normal de Eventos

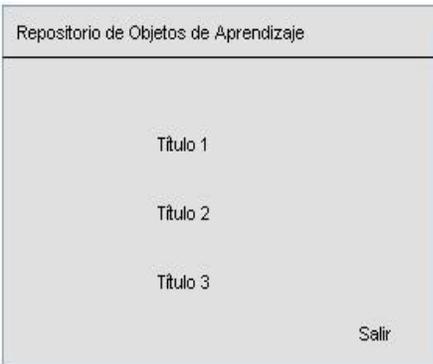
Sección 1 “Listar OA recientes”

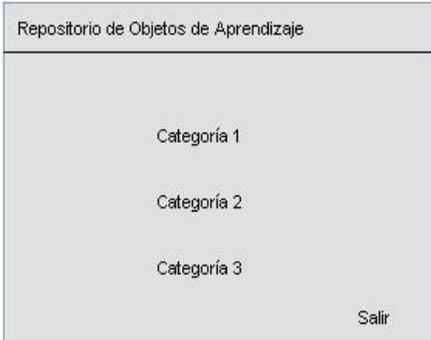
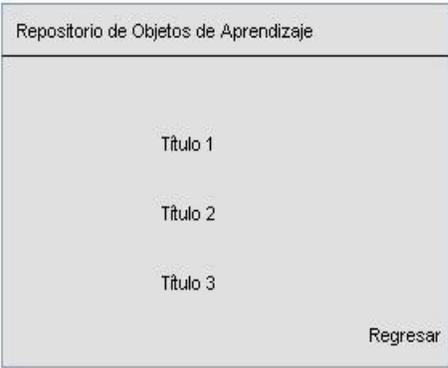
Acción del Actor	Respuesta del Sistema
El usuario anónimo, usuario autenticado, administrador o revisor selecciona la opción listar OA recientes.	
	El sistema muestra el listado de OA recientes por título y permite salir.
El usuario anónimo, usuario autenticado, administrador o revisor selecciona el OA a ver.	
	El sistema muestra una pantalla con la información que contiene el OA y permite: Calificar OA (ver sección 3: calificar OA). Descargar (ver sección 4: descargar OA). Agregar comentario (ver sección 5: agregar comentario). Además permite: Regresar.

Flujos Alternos “Salir”

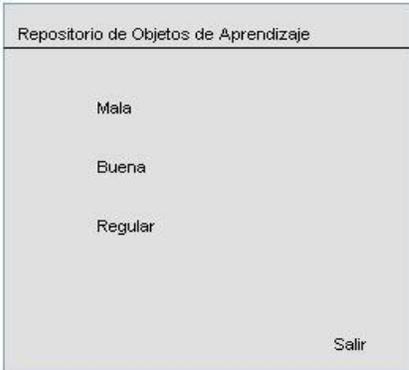
Acción del Actor	Respuesta del Sistema
------------------	-----------------------

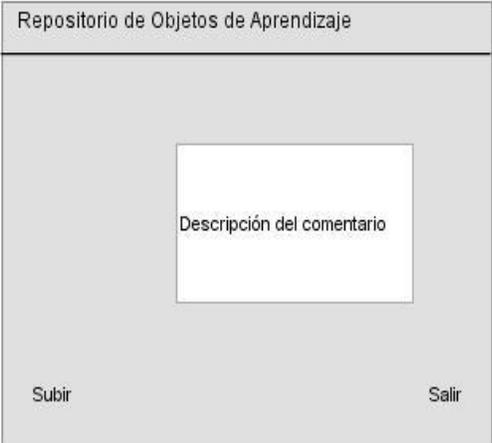
	Regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.
Flujos Alternos “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar OA recientes.
Prototipo de interfaz	
 <p>Repositorio de Objetos de Aprendizaje</p> <p>Título 1</p> <p>Título 2</p> <p>Título 3</p> <p>Salir</p>	
Flujo Normal de Eventos	
Sección 2 “Listar OA más solicitados”	
El usuario anónimo, usuario autenticado, administrador o revisor selecciona la opción de los OA más solicitados.	
	El sistema muestra el listado de OA recientes por título y permite salir.
El usuario anónimo, usuario autenticado, administrador o revisor selecciona la opción a ver.	
	El sistema muestra una pantalla con la información que contiene el OA y permite: Calificar OA (ver sección 4: calificar OA).

	<p>Descargar (ver sección 5: descargar OA).</p> <p>Agregar Comentario (ver sección 6: agregar comentario).</p> <p>Además permite: Regresar.</p>
Flujos Alternos “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar OA más solicitados.
Flujos Alternos “Salir”	
Acción del Actor	Respuesta del Sistema
.	Regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.
Prototipo de interfaz	
	
Flujo Normal de Eventos	
Sección 3 “Listar OA por categoría”	
Acción del Actor	Respuesta del Sistema
1. El usuario anónimo, usuario autenticado, administrador o revisor selecciona la opción OA por categoría.	

	<p>1.1. El sistema muestra el listado de las categorías a seleccionar. Permite la opción: Salir.</p>
<p>Prototipo de interfaz</p> 	
<p>El usuario anónimo, usuario autenticado, administrador o revisor selecciona la categoría de OA que quiere ver.</p>	
	<p>El sistema muestra el listado de OA por título. Permite: Regresar.</p>
<p>Prototipo de interfaz</p> 	
<p>El usuario anónimo, usuario autenticado, administrador o revisor seleccionan el OA a ver.</p>	
	<p>El sistema muestra una pantalla con la información que contiene el OA y permite:</p>

	<p>Calificar OA (ver sección 4: calificar OA).</p> <p>Descargar (ver sección 5: descargar OA).</p> <p>Agregar Comentario (ver sección 6: agregar comentario).</p> <p>Además permite: Regresar.</p>
Flujo Alterno “Salir”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.
Flujo Alterno “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar OA por categoría.
Flujo Normal de Eventos	
Sección 4 “Calificar OA”	
Acción del Actor	Respuesta del Sistema
El usuario autenticado, administrador o revisor deciden calificar un OA.	
	<p>El sistema muestra una pantalla con las opciones a seleccionar por el actor:</p> <p>Buena.</p> <p>Mala.</p> <p>Regular.</p> <p>Salir.</p>
El usuario autenticado, administrador o revisor seleccionan la calificación.	
	El sistema registra la calificación seleccionada y regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones

	de OA.
Flujo Alterno “Salir”	
	Regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.
Prototipo de interfaz	
 <p>The screenshot shows a window titled 'Repositorio de Objetos de Aprendizaje'. Inside the window, there is a list of three quality ratings: 'Mala', 'Buena', and 'Regular'. At the bottom right of the window, there is a button labeled 'Salir'.</p>	
Flujo Normal de Eventos	
Sección 5 “Descargar OA”	
Acción del Actor	Respuesta del Sistema
El usuario autenticado, usuario anónimo, administrador o revisor seleccionan descargar OA.	
	Descarga el OA y regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.
Flujo Normal de Eventos	
Sección 6 “Agregar comentario”	
Acción del Actor	Respuesta del Sistema
El usuario autenticado, administrador o revisor seleccionan agregar comentario al OA.	
	El sistema muestra una ventana con:

	<p>Campo de texto para introducir el comentario. Permite las opciones subir y salir.</p>
<p>El usuario autenticado, administrador o revisor seleccionan subir el comentario.</p>	
	<p>Verifica que el campo no este vacío. El sistema registra el comentario y regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.</p>
<p>Flujo Alterno “Salir”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>Regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA.</p>
<p>Prototipo de interfaz</p>	
	
<p>Poscondiciones</p>	<p>Se visualiza correctamente el listado de OA.</p>

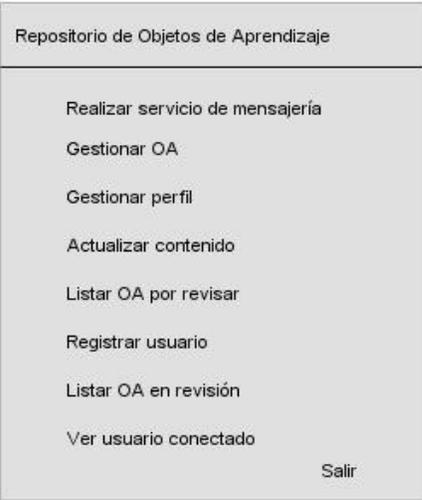
Caso de uso “Ver usuarios conectados”

<p>Flujo Normal de Eventos</p>	
<p>Sección “Ver usuarios conectados”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>

<p>El usuario anónimo, usuario autenticado, administrador o revisor decide ver usuarios conectados.</p>	<p>Muestra un listado de los usuarios que están online. Permite salir.</p>
<p>El usuario anónimo, usuario autenticado, administrador o revisor seleccionan el usuario a ver.</p>	
	<p>Cada usuario va a mostrar: Nombre. Apellidos Usuario. Dirección de correo. Número telefónico. Brinda opciones: Agregar usuario al móvil. (Ver sección 1: Agregar usuario al móvil). Realizar llamada telefónica. (Ver sección 2: Realizar llamada). Enviar SMS al móvil del usuario. (Ver sección 3: Enviar SMS). Permite Regresar.</p>
<p>Flujos Alternos</p>	
<p>Flujo Alterno “Salir”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>Regresa a la portada del sistema (ver caso de uso: Mostrar portada del sistema) en el caso de estar autenticado, sino regresa a la interfaz externa de la aplicación (ver figura Mostrar interfaz externa de la aplicación que se encuentra al final de las descripciones de casos de uso). Termina el caso de uso.</p>
<p>Flujo Alterno “Regresar”</p>	

Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar todos los usuarios.
<p>Prototipo de interfaz</p> <div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <p style="text-align: center;">Repositorio de Objetos de Aprendizaje</p> <hr/> <p>Nombre: Yanelis Apellidos: Batista Reyes Usuario: ybreyes Dirección de correo: ybreyes@estudiantes.uci.cu Número telefónico 58180208</p> <p style="text-align: right;">Regresar</p> </div>	
Flujo Normal de Eventos	
Sección 1 “Agregar usuario al móvil”	
Acción del actor	Respuesta del sistema
El usuario anónimo, usuario autenticado, administrador o revisor decide agregar usuario al móvil.	Termina el caso de uso.
Flujo Normal de Eventos	
Sección 2 “Enviar SMS a un usuario”	
Acción del actor	Respuesta del sistema
	Termina el caso de uso.
Flujo Normal de Eventos	
Sección 3 “Llamar a un usuario”	
Acción del actor	Respuesta del sistema
	Termina el caso de uso.
Poscondiciones	Se muestra el usuario correctamente.

Caso de uso “Mostrar portada”

Flujo Normal de Eventos	
Sección “Mostrar portada ”	
Acción del Actor	Respuesta del Sistema
El administrador, revisor o usuario autenticado, entran al sistema.	
	1.1 Muestra una pantalla con las funcionalidades que brinda la aplicación dependiendo del rol que tenga. Permite Salir.
Flujo Alternativo “Salir”	
Acción del Actor	Respuesta del Sistema
	Sale de la vista actual y regresa a la interfaz externa de la aplicación (ver figura Mostrar interfaz externa de la aplicación que se encuentra al final de las descripciones de casos de uso).
<p>Prototipo de interfaz</p> 	
Poscondiciones	Se muestra la portada del sistema correctamente.

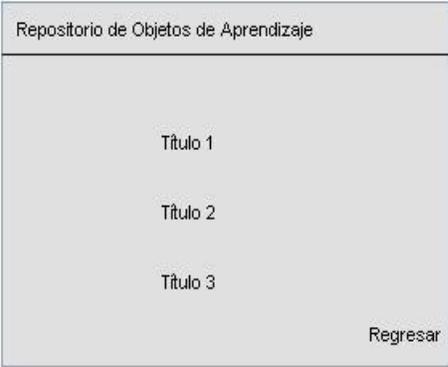
Caso de uso “Listar OA en revisión”

Flujo Normal de Eventos	
Sección “Listar OA en revisión ”	
Acción del Actor	Respuesta del Sistema
El revisor decide listar los OA que se encuentran en revisión.	
	Muestra un listado de OA. Permite Regresar.
El revisor selecciona del listado un OA ver.	
	Muestra la información que contiene el OA. Permite salir.
Flujo Alterno “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa a la interfaz externa del sistema (Ver caso de uso: Mostar portada del sistema).
Flujo Alterno “Salir”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar OA en revisión.
Prototipo de interfaz	

<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <p>Repositorio de Objetos de Aprendizaje</p> <hr/> <p>Título 1</p> <p>Título 2</p> <p>Título 3</p> <p style="text-align: right;">Regresar</p> </div>	
Poscondiciones	Se visualizan los OA en revisión correctamente.

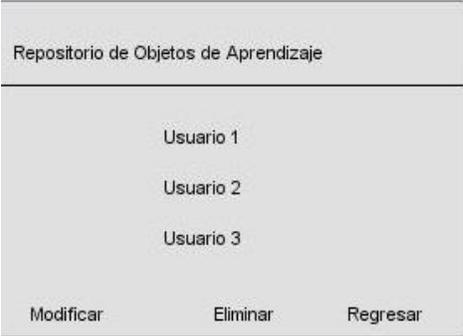
Caso de uso “Listar OA por revisar”

Flujo Normal de Eventos	
Sección “Listar OA por revisar ”	
Acción del Actor	Respuesta del Sistema
El revisor decide listar los OA que no han sido revisados.	
	Muestra un listado de OA sin revisar. Permite salir.
El revisor selecciona del listado un OA ver.	
	Muestra la información que contiene el OA. Permite regresar.
Flujo Alterno “Salir”	
Acción del Actor	Respuesta del Sistema
	Regresa a la interfaz externa del sistema (Ver caso de uso: Mostar portada del sistema).
Flujo Alterno “Regresar”	

Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección listar publicaciones de OA en revisión.
<p>Prototipo de interfaz</p> 	
Poscondiciones	Se visualizan los OA por revisar correctamente.

Caso de uso “Gestionar perfil”

Flujo Normal de Eventos	
Sección “Gestionar perfil”	
Acción del Actor	Respuesta del Sistema
El administrador decide gestionar perfil.	
	<p>Muestra un listado de usuarios registrado en la aplicación. Y brinda las opciones:</p> <p>Si decide modificar perfil (ver sección 1: modificar perfil).</p> <p>Si decide eliminar perfil (ver sección 2: eliminar perfil).</p> <p>Permite salir.</p>
Flujo Alternativo “Salir”	

Acción del Actor	Respuesta del Sistema
	Muestra la portada del sistema (ver caso de uso: Mostrar portada del sistema).
<p>Prototipo de interfaz</p> 	
Sección 1: “Modificar perfil”	
Acción del Actor	Respuesta del Sistema
El administrador, revisor o usuario selecciona modificar perfil.	
	Muestra una pantalla con los datos que contiene el perfil del usuario. Permite: Guardar nuevos datos, regresar.
El usuario, administrador y revisor entran los nuevos datos y da guardar.	
	Guarda los nuevos datos en el sistema. Regresa al paso 1.1 del flujo normal de evento de la sección gestionar perfil.
Flujo Alternativo “Regresar”	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección gestionar perfil.

Sección 2: “ Eliminar perfil”	
Acción del Actor	Respuesta del Sistema
1.1 El administrador decide eliminar perfil.	
	Se elimina el perfil de la base de datos. Regresa al paso 1.1 del flujo normal de evento de la sección gestionar perfil.
Poscondiciones	Se establece el gestionar perfil sin problema.

Caso de uso “Actualizar contenido”

Flujo Normal de Eventos	
Sección “Actualizar contenido del sistema”	
Acción del Actor	Respuesta del Sistema
El actor accede al vínculo contenido.	
	El sistema actualiza todo el contenido Guarda el nuevo contenido a la base de datos, sustituyendo al ya existente. Regresa a la interfaz externa de la aplicación (ver figura Mostrar interfaz externa de la aplicación que se encuentra al final de las descripciones de casos de uso) en el caso de no estar autenticado, sino muestra la portada del sistema (ver caso de uso: Mostrar portada del sistema). Termina el caso de uso.
Poscondiciones	Se actualizan los datos de la aplicación correctamente.

Caso de uso “Gestionar OA”

Flujo Normal de Eventos

Sección “Gestionar OA”

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

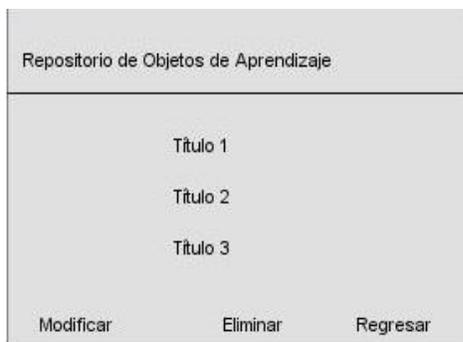
El administrador accede decide gestionar OA.	
	Muestra listado de OA. Y brinda las opciones dependiendo del rol: Si decide modificar OA (ver sección 1: modificar OA). Si decide eliminar OA (ver sección 2: eliminar OA). Permite salir.

Flujo Alternativo “Salir”

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

	Muestra la portada del sistema (ver caso de uso: Mostrar portada del sistema).
--	---

Prototipo de interfaz



Sección 1: “Modificar OA”

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

El administrador, revisor o usuario selecciona modificar	
--	--

OA.	
	Muestra una pantalla con la información que contiene el OA. Permite: Guardar nuevos datos, regresar.
El usuario, administrador y revisor entran los nuevos datos y selecciona guardar datos.	
	Guarda los nuevos datos en el sistema. Regresa al paso 1.1 del flujo normal de evento de la sección gestionar OA.
Flujo Alternativo "Regresar"	
Acción del Actor	Respuesta del Sistema
	Regresa al paso 1.1 del flujo normal de evento de la sección gestionar perfil.
Sección 2: "Eliminar OA"	
Acción del Actor	Respuesta del Sistema
El administrador decide eliminar OA.	
	Se elimina el OA de la base de datos. Regresa al paso 1.1 del flujo normal de evento de la sección gestionar OA.
Poscondiciones	Se establece el gestionar OA sin problema.

Caso de uso "Registrar usuario":

Flujo Normal de Eventos	
Sección "Registrar usuario"	
Acción del Actor	Respuesta del Sistema
El administrador decide registrar un	

usuario a la aplicación.	
	<p>El sistema muestra los datos de un usuario para ser registrado.</p> <p>El sistema guarda los datos del usuario y regresa a la portada del sistema (ver caso de uso: Mostrar portada del sistema).</p> <p>Termina el caso de uso.</p>
Poscondiciones	Se registra el usuario en la aplicación.

Prototipo de interfaz

Repositorio de objetos de aprendizaje

Nombre

Apellidos

Usuario

Contraseña

Dirección de correo

número de teléfono

Repositorio de Objetos de Aprendizaje

Autenticar usuario

Recuperar contraseña

Actualizar contenido

Ver usuarios conectados

Listar publicaciones de OA

Salir

Figura 21-Interfaz externa de la aplicación.

Anexo 2 Diagramas de clases del análisis



Figura 22-Diagrama de clases del análisis “Recuperar contraseña”.

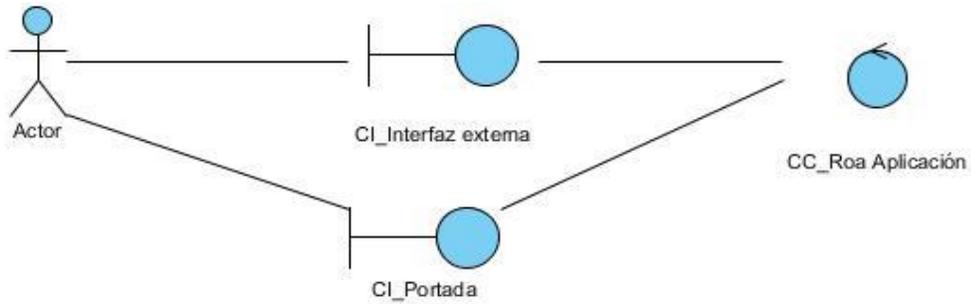


Figura 23-Diagrama de clases del análisis “Mostrar portada”.

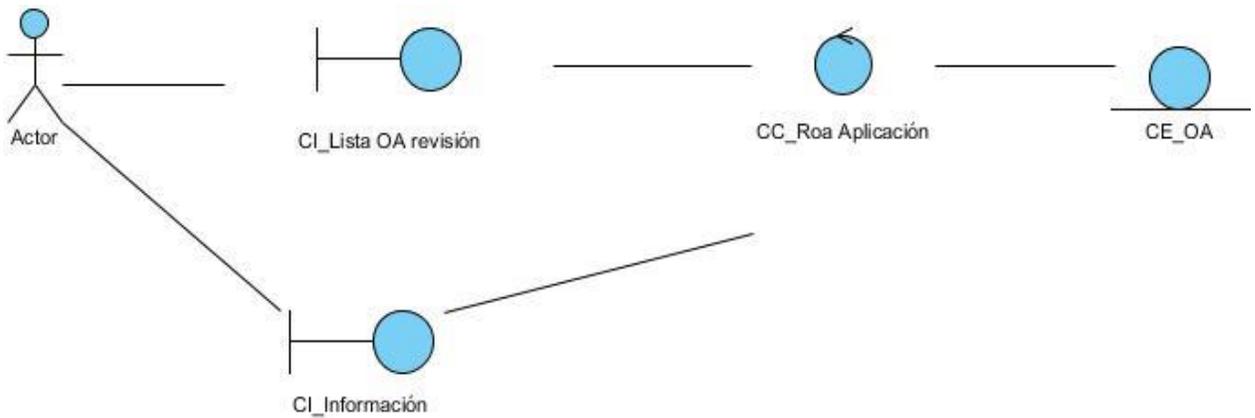


Figura 24-Diagrama de clases del análisis “Listar OA (en revisión)”.

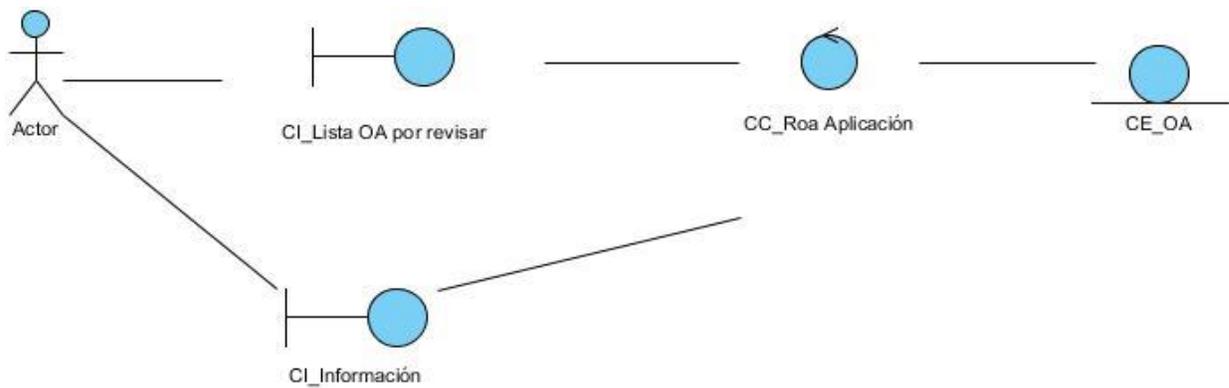


Figura 25-Diagrama de clases del análisis “Listar OA (por revisar)”.

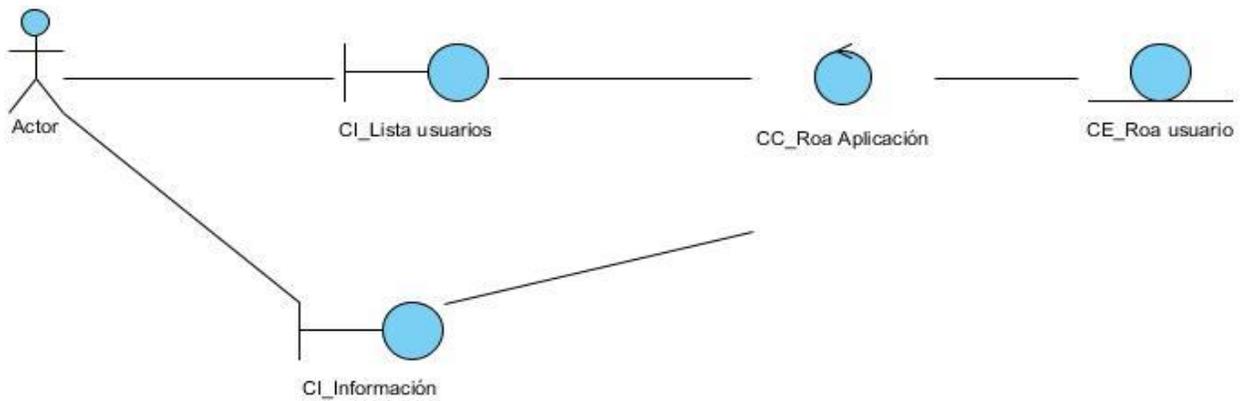


Figura 26-Diagrama de clases del análisis “Ver usuarios conectados”.



Figura 27-Diagrama de clases del análisis “Actualizar contenido”.

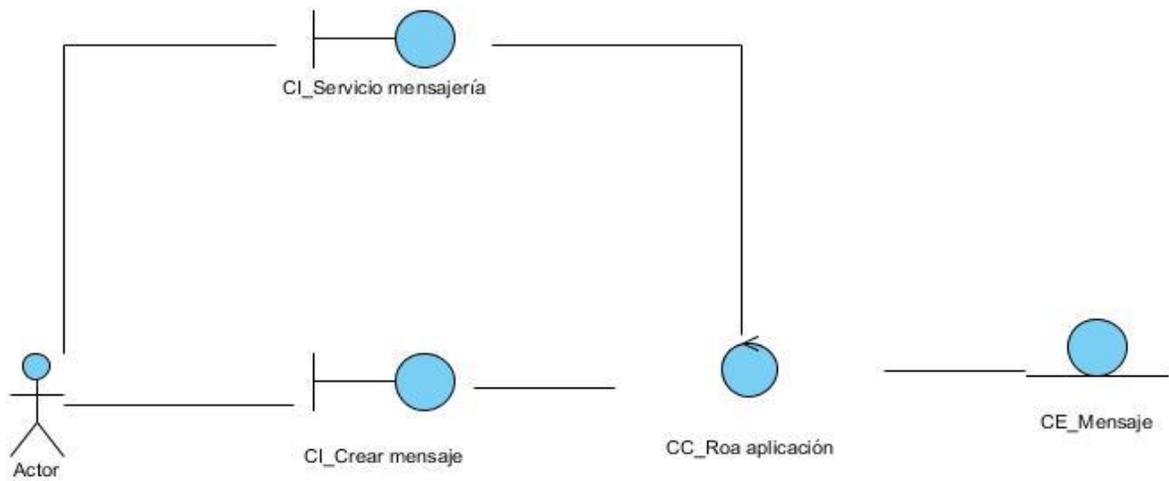


Figura 28-Diagrama de clases del análisis “Servicio de mensajería (Crear mensaje)”.

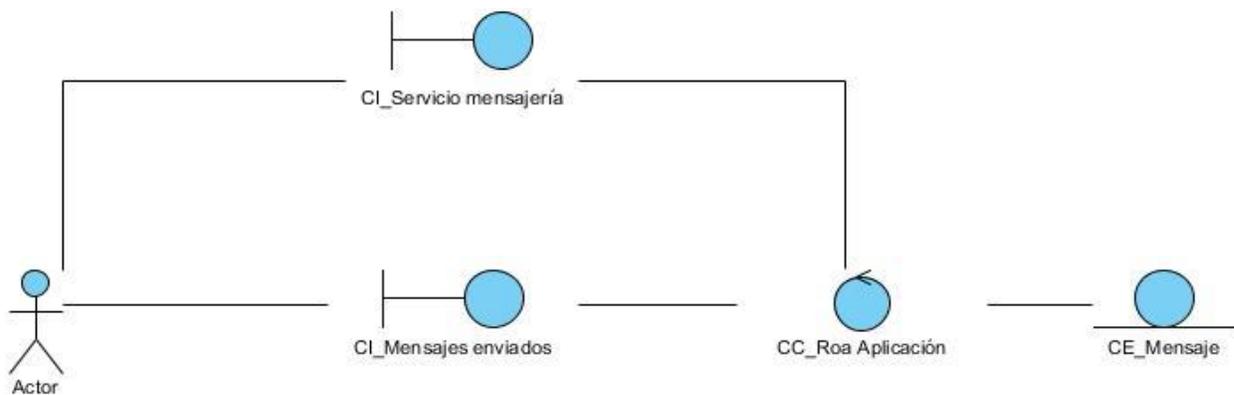


Figura 29-Diagrama de clases del análisis “Servicio de mensajería (Mensajes enviados).”

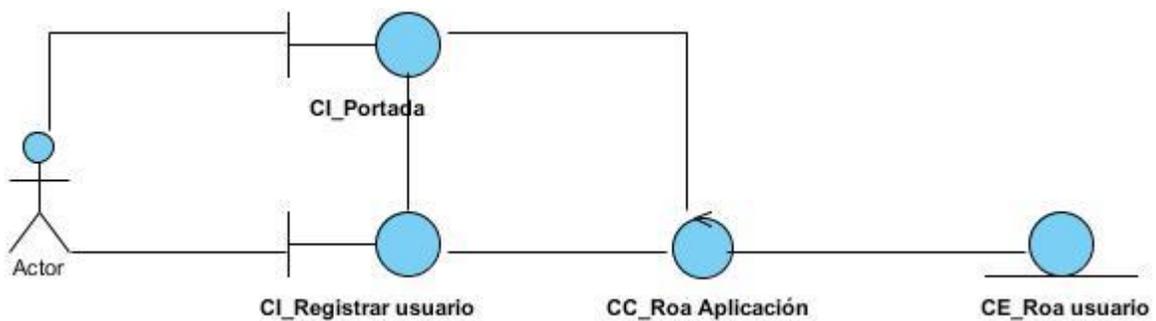


Figura 30-Diagrama de clases del análisis (Registrar usuario).

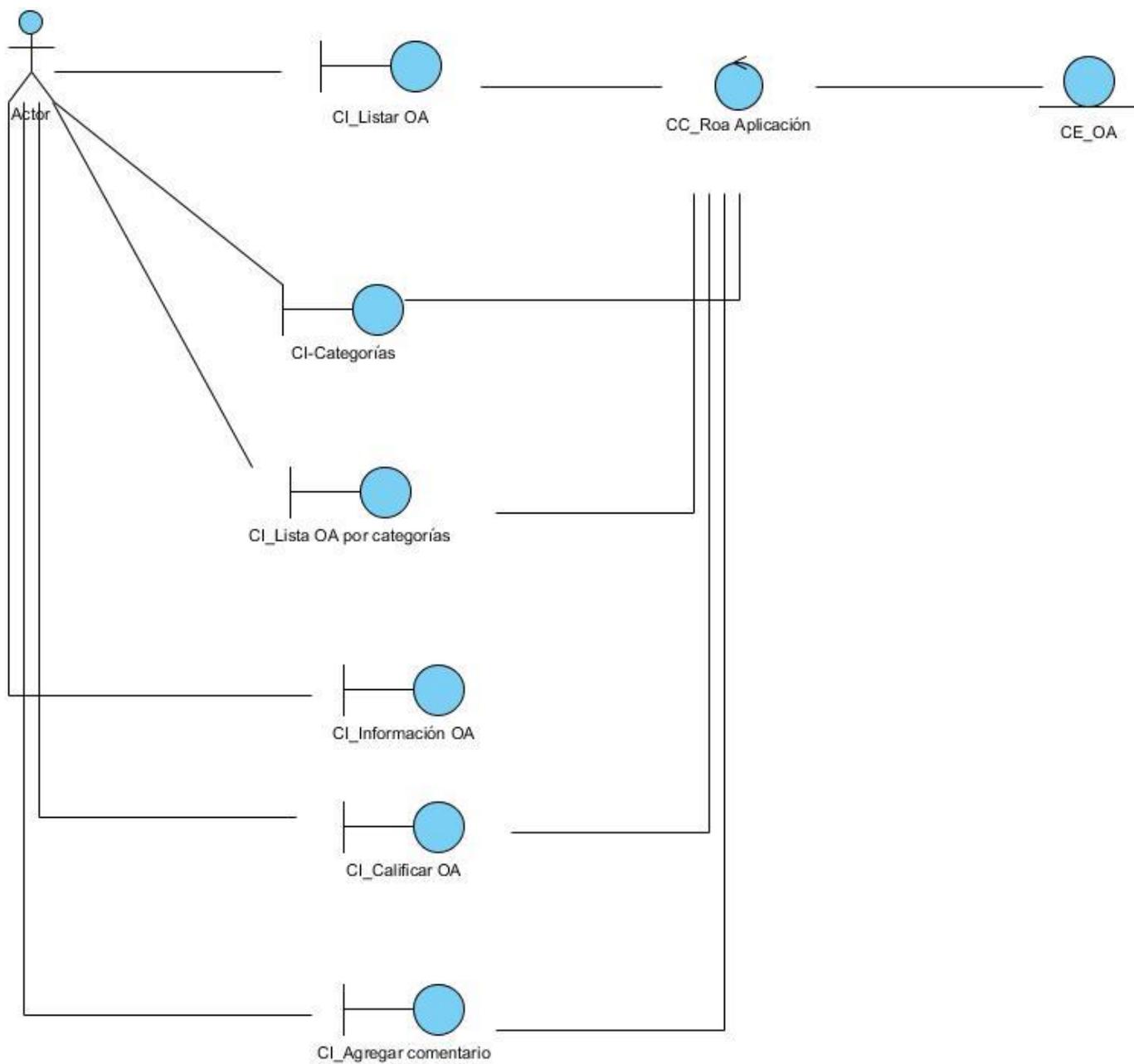


Figura 31-Diagrama de clases del análisis “Listar OA (por categorías)”.

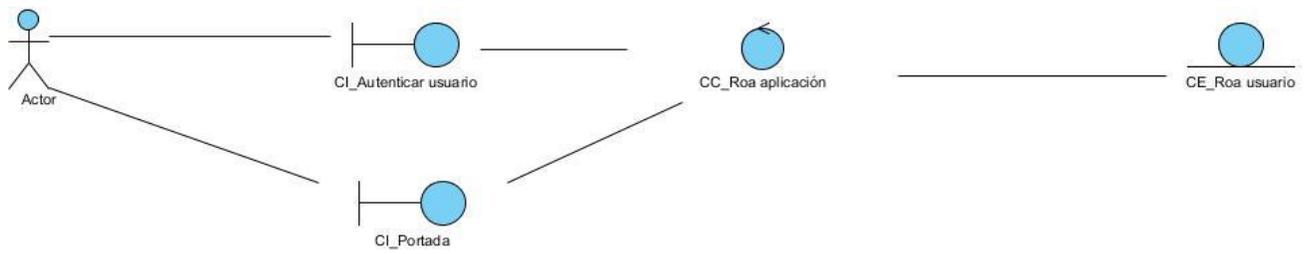


Figura 32-Diagrama de clases del análisis “Autenticar usuario”.

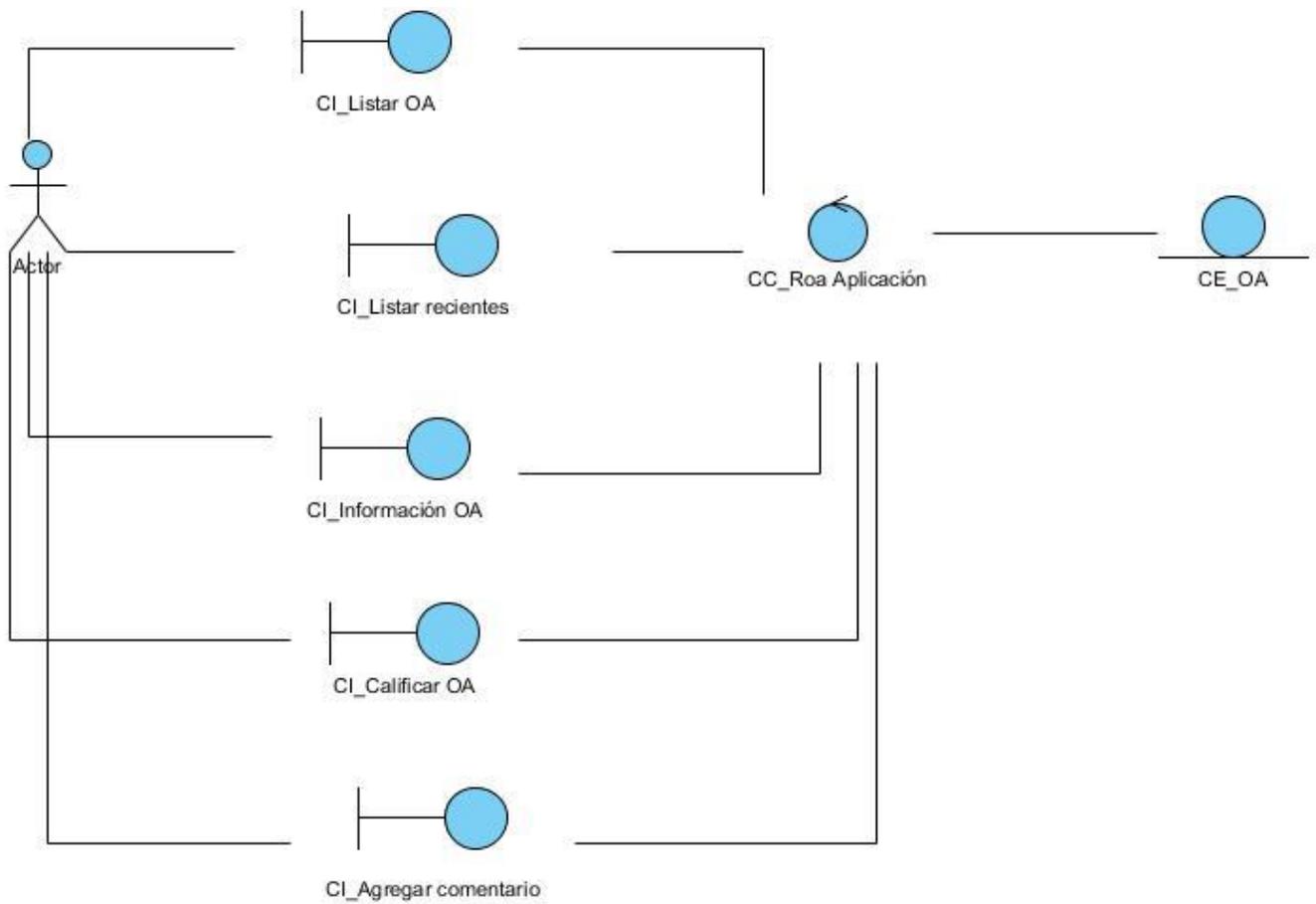


Figura 33-Diagrama de clases del análisis “Listar OA (recientes)”.

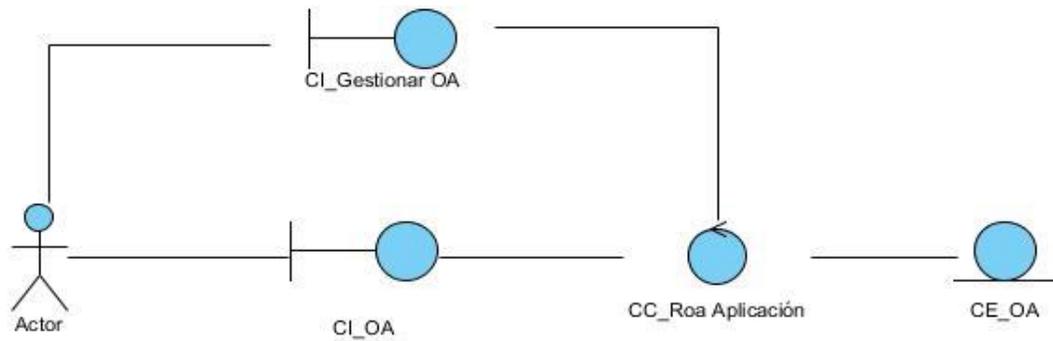


Figura 34-Diagrama de clases del análisis “Gestionar OA”.

Anexo 3 Diagramas de colaboración

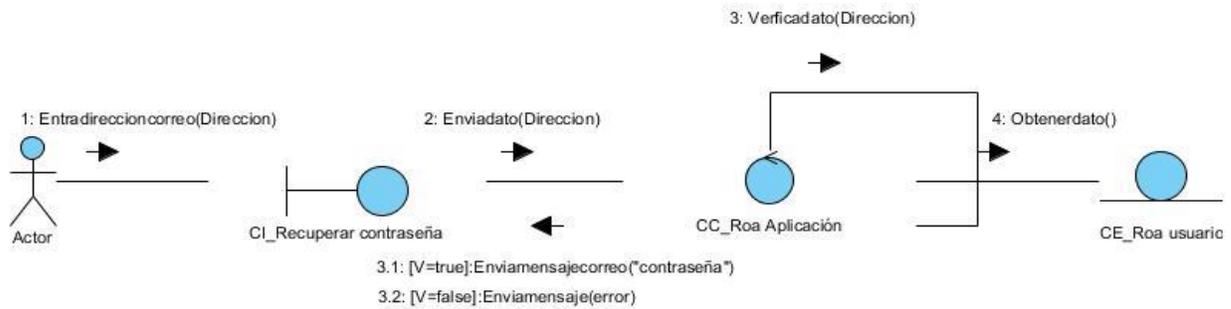


Figura 35-Diagrama de colaboración “Recuperar contraseña”.

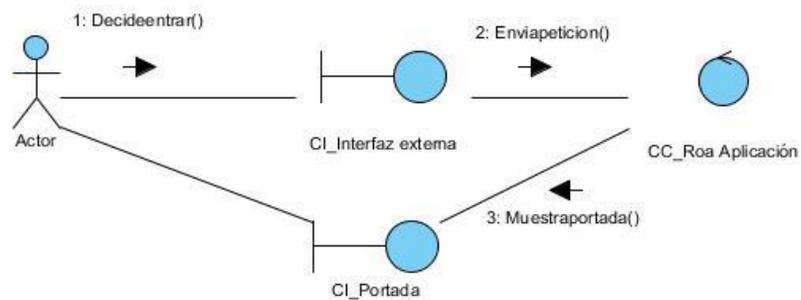


Figura 36-Diagrama de colaboración “Mostrar portada”.

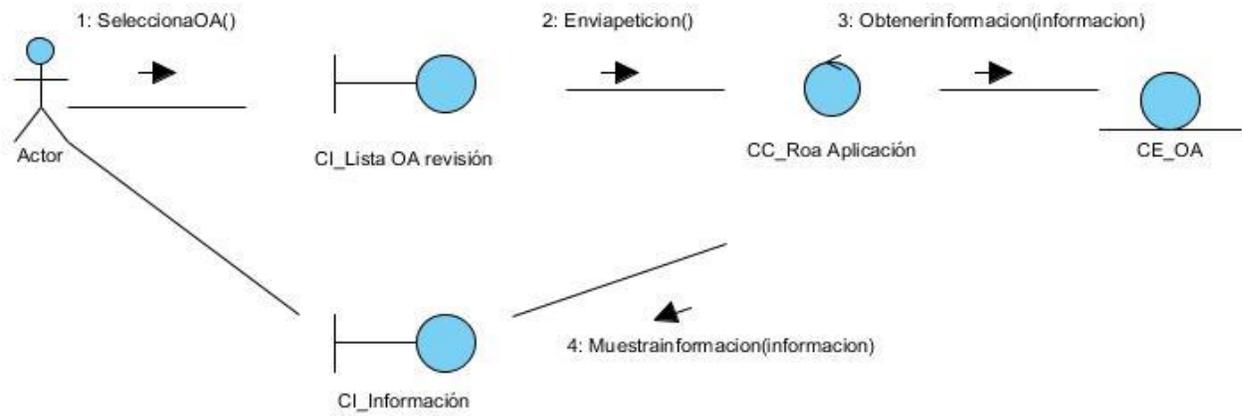


Figura 37-Diagrama de colaboración “Listar OA (en revisión)”.

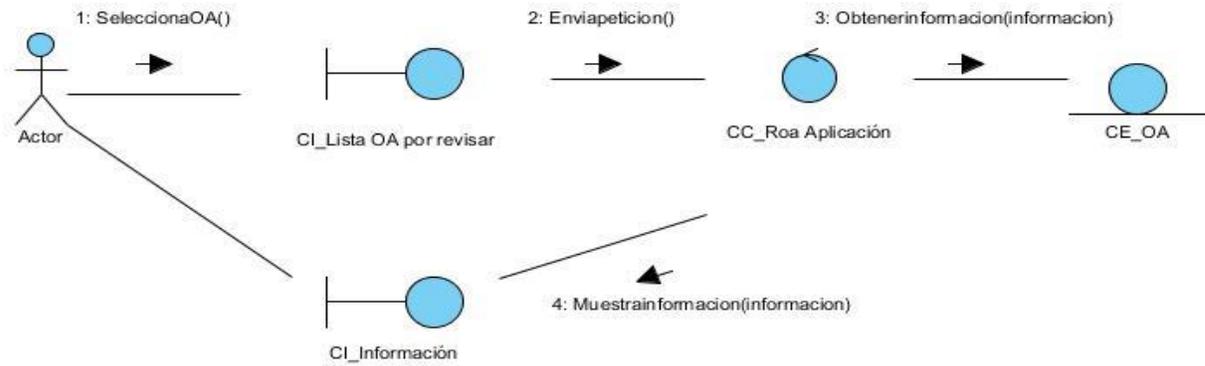


Figura 38-Diagrama de colaboración “Listar OA (por revisión)”.

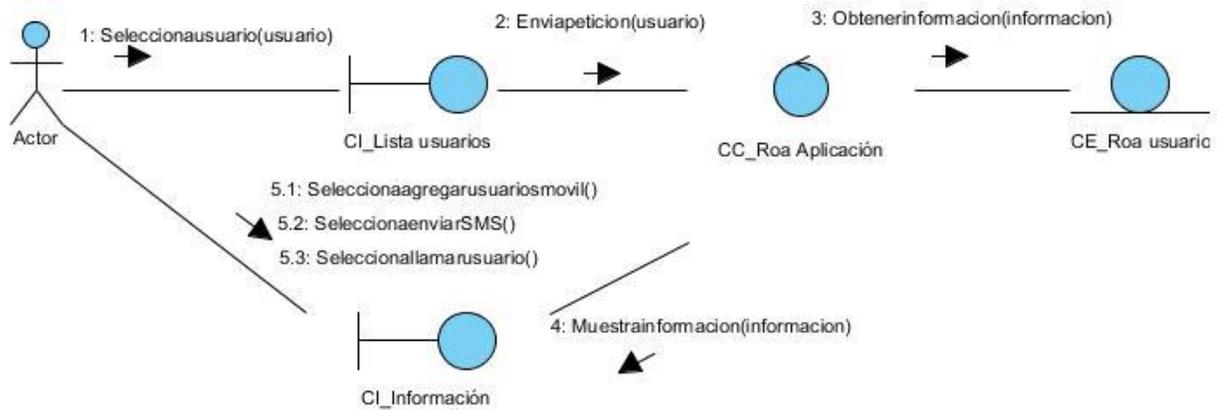


Figura 39-Diagrama de colaboración “Ver usuarios conectados”.

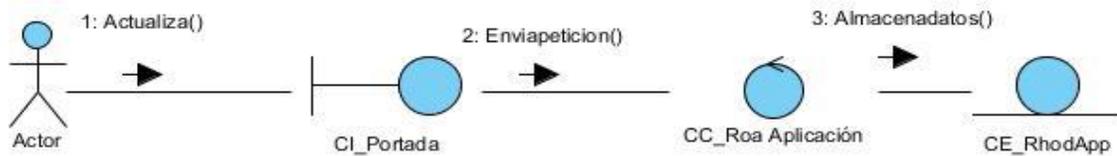


Figura 40-Diagrama de colaboración “Actualizar contenido”.

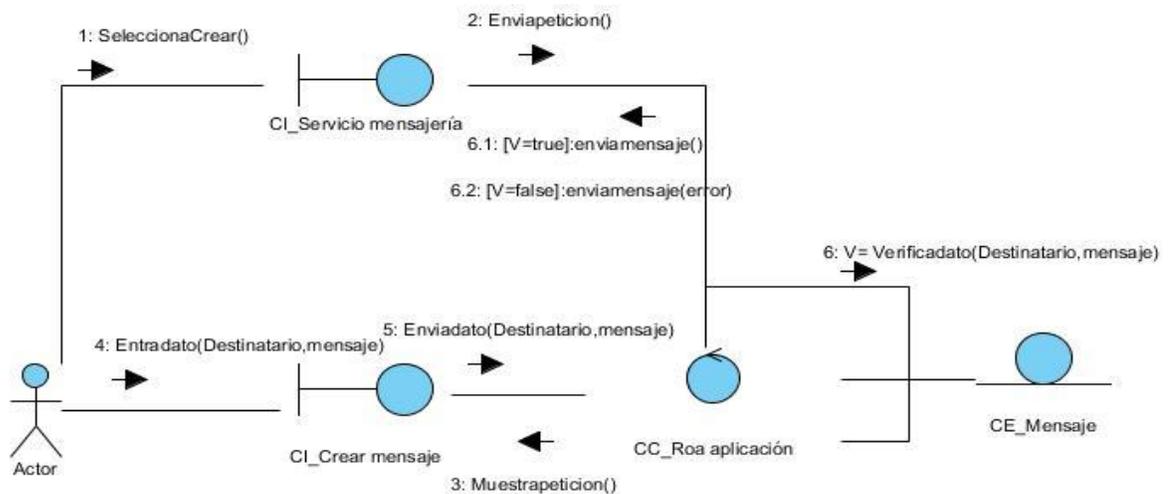


Figura 41-Diagrama de colaboración “Servicio de mensajería (crear mensaje)”.

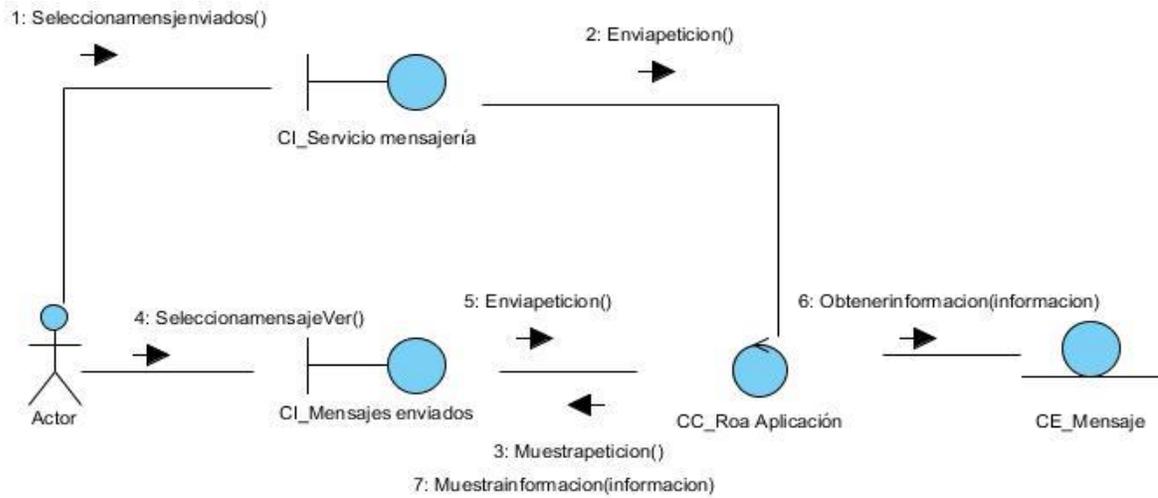


Figura 42-Diagrama de colaboración “Servicios de mensajería (mensajes enviados)”.

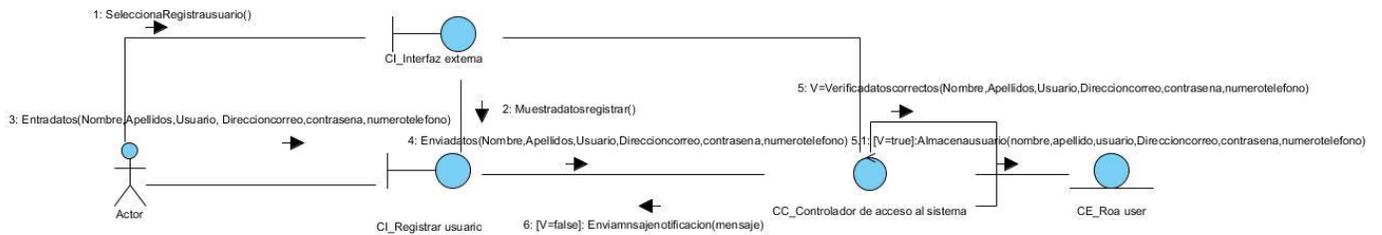


Figura 43-Diagrama de colaboración “Registrar usuario”.

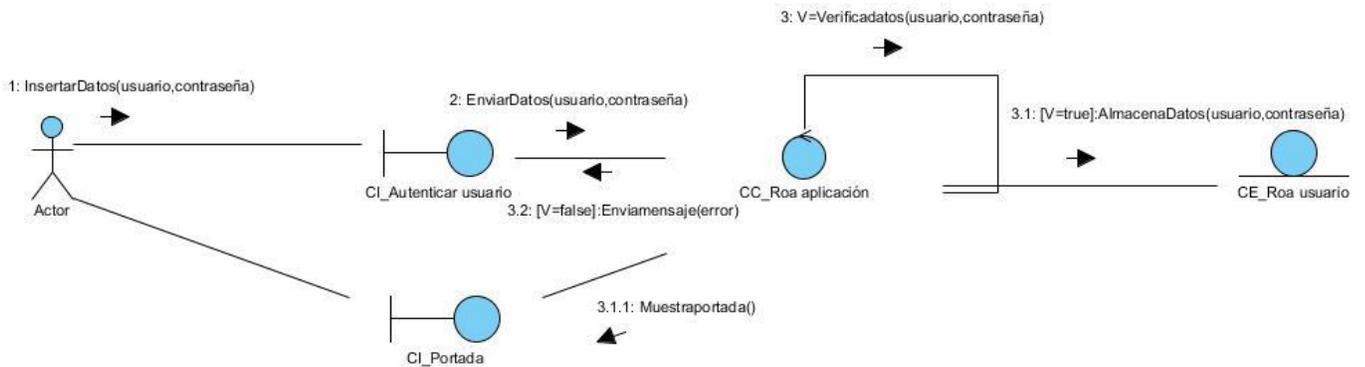


Figura 44-Diagrama de colaboración “Autenticar usuario”.

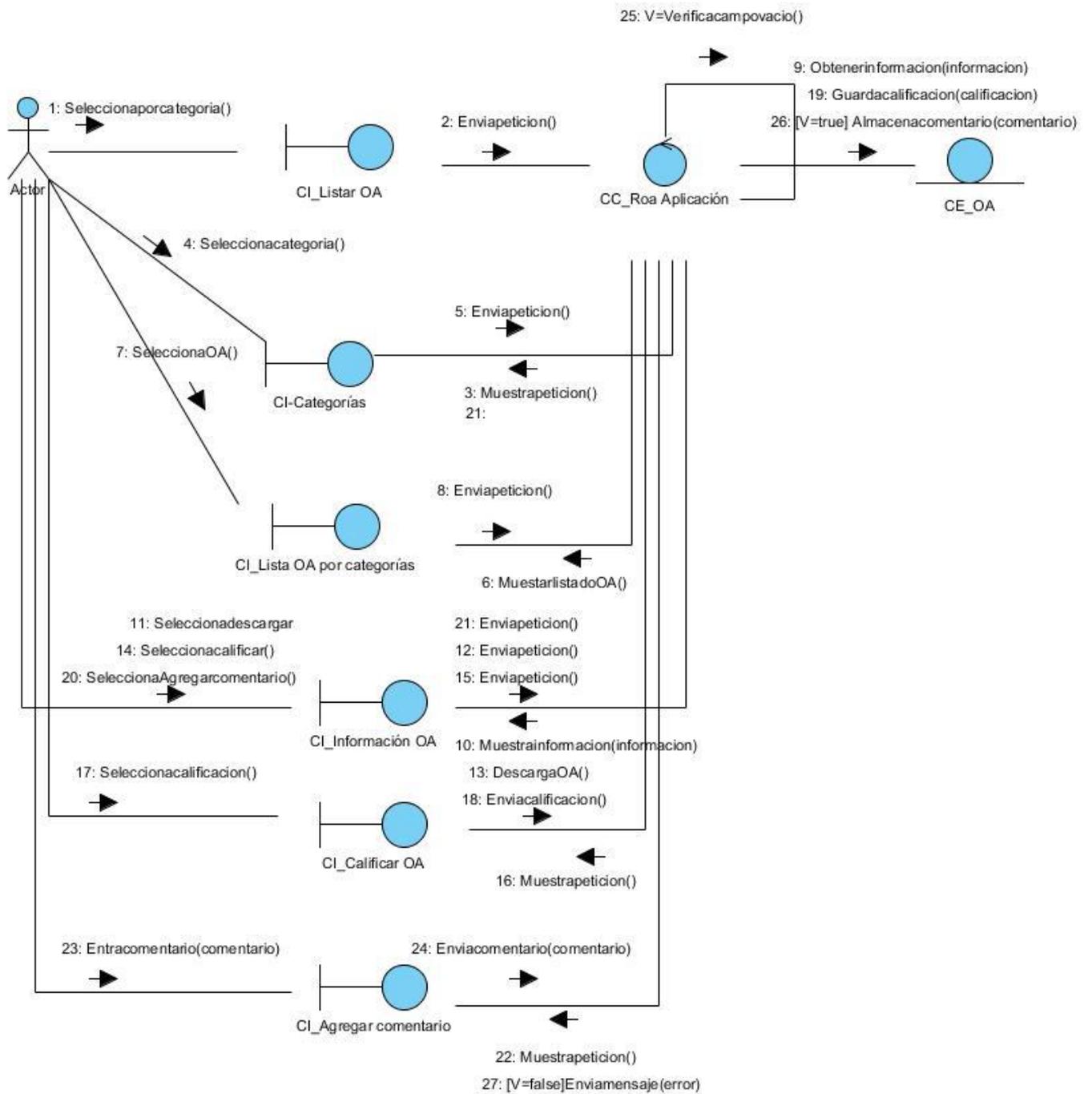


Figura 45-Diagrama de colaboración “Listar OA (por categorías)”.

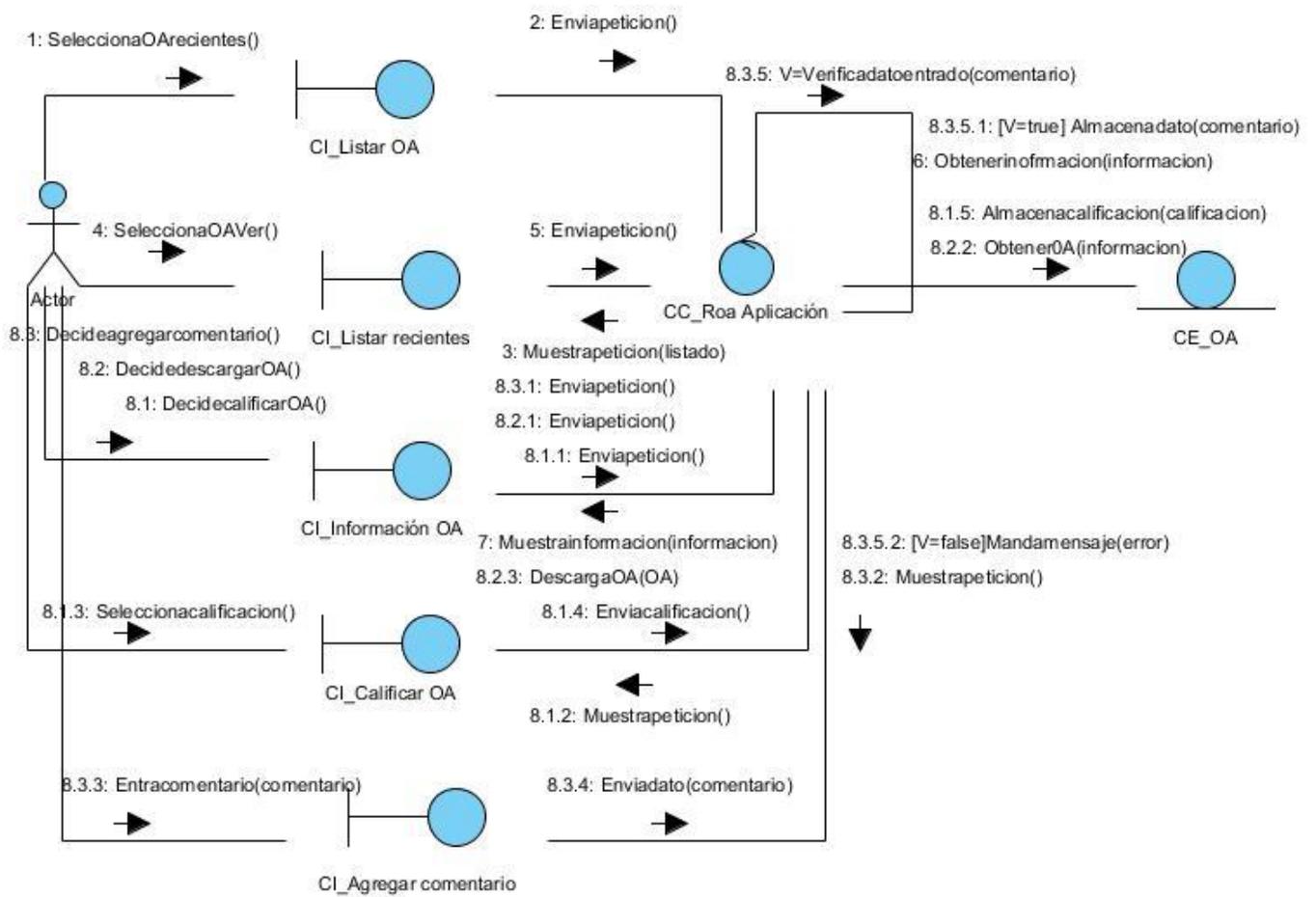


Figura 46-Diagrama de colaboración “Listar OA (recientes)”.

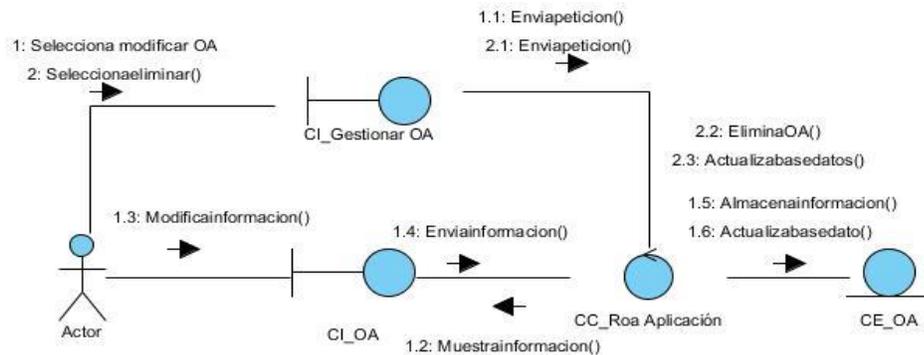


Figura 47-Diagrama de colaboración “Gestionar OA”.

Anexo 4: Diagrama de secuencia

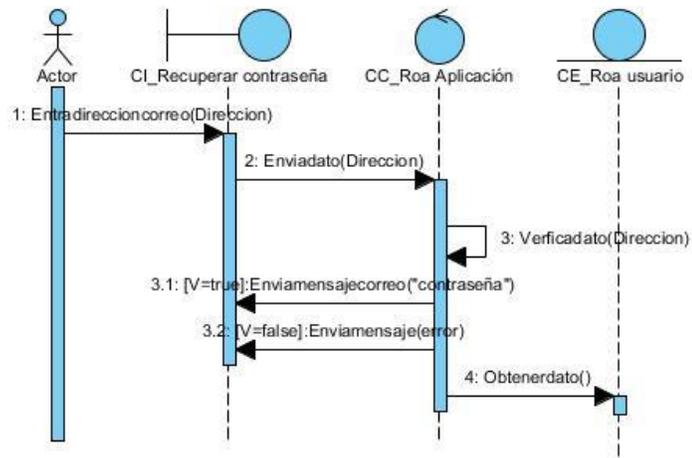


Figura 48-Diagrama de secuencia “Recuperar contraseña”.

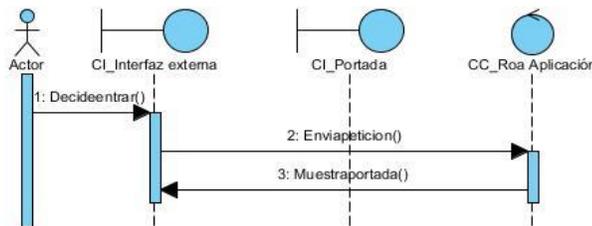


Figura 49-Diagrama de secuencia “Mostrar portada”.

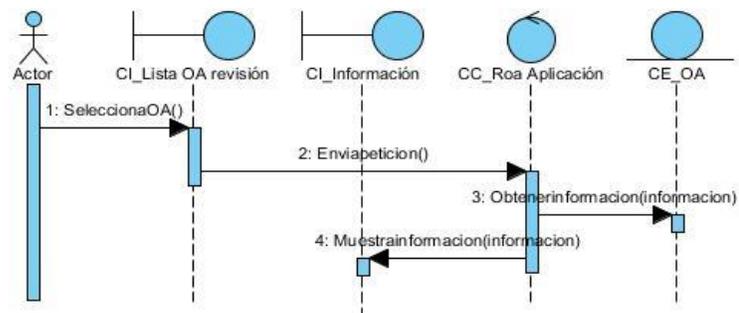


Figura 50-Diagrama de secuencia “Lista OA (en revisión)”.

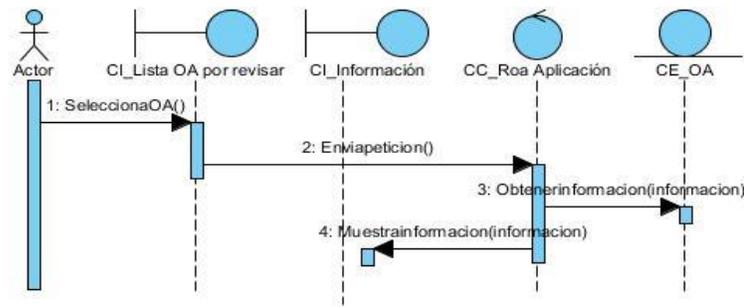


Figura 51-Diagrama de secuencia “Lista OA (por revisar)”.

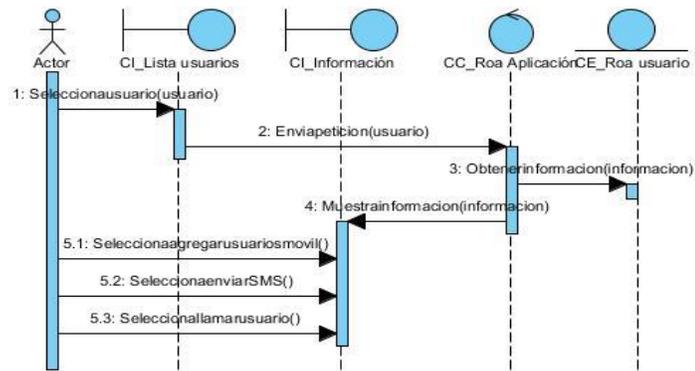


Figura 52-Diagrama de secuencia “Ver usuarios conectados”.

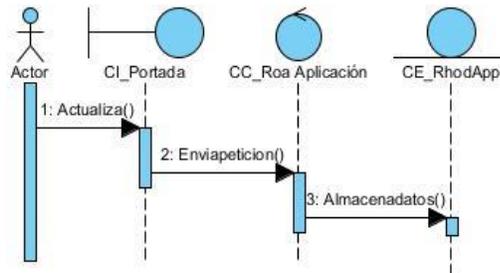


Figura 53-Diagrama de secuencia “Actualizar contenido”.

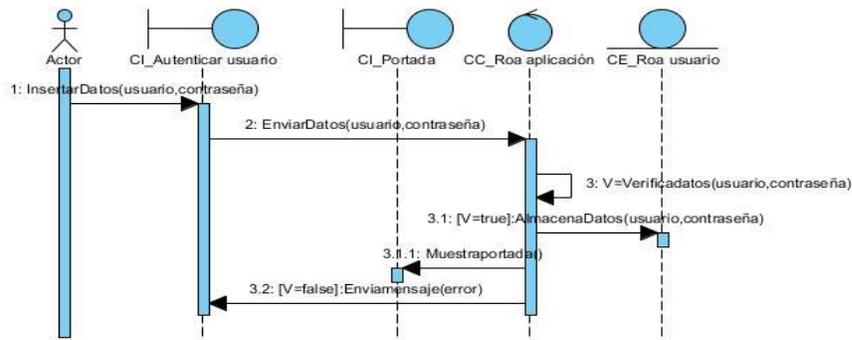


Figura 54-Diagrama de secuencia “Autenticar usuario”.

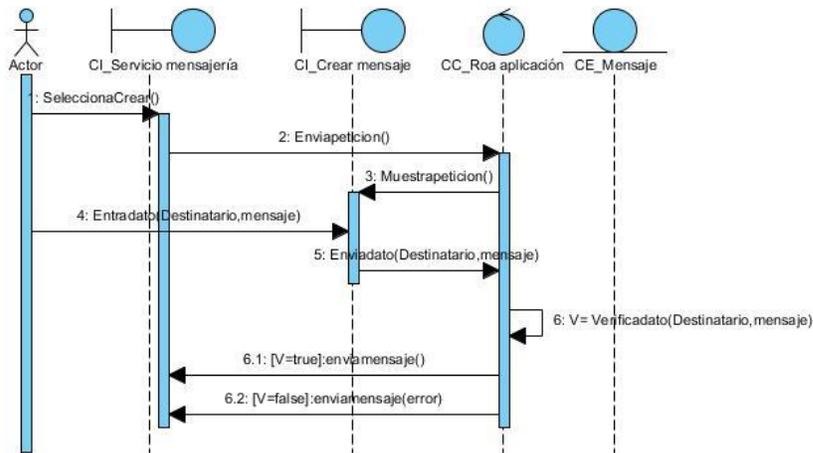


Figura 55-Diagrama de secuencia “Servicio de mensajería (Crear mensaje)”.

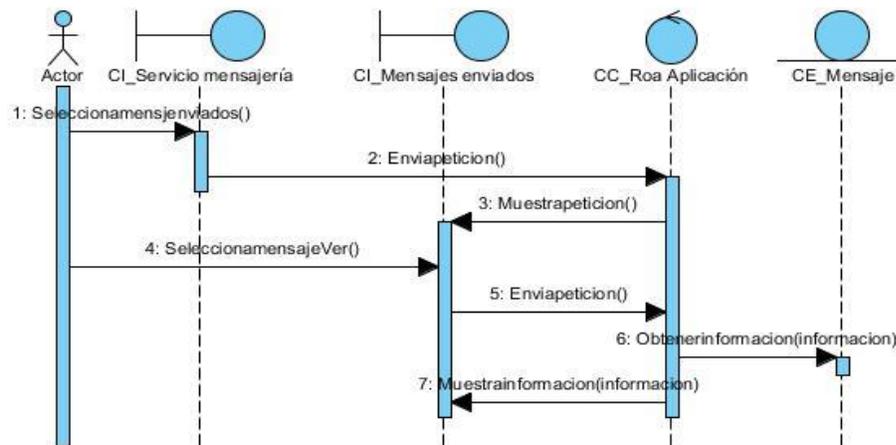


Figura 56-Diagrama de secuencia “Servicio de mensajería (Mensajes enviado)”.

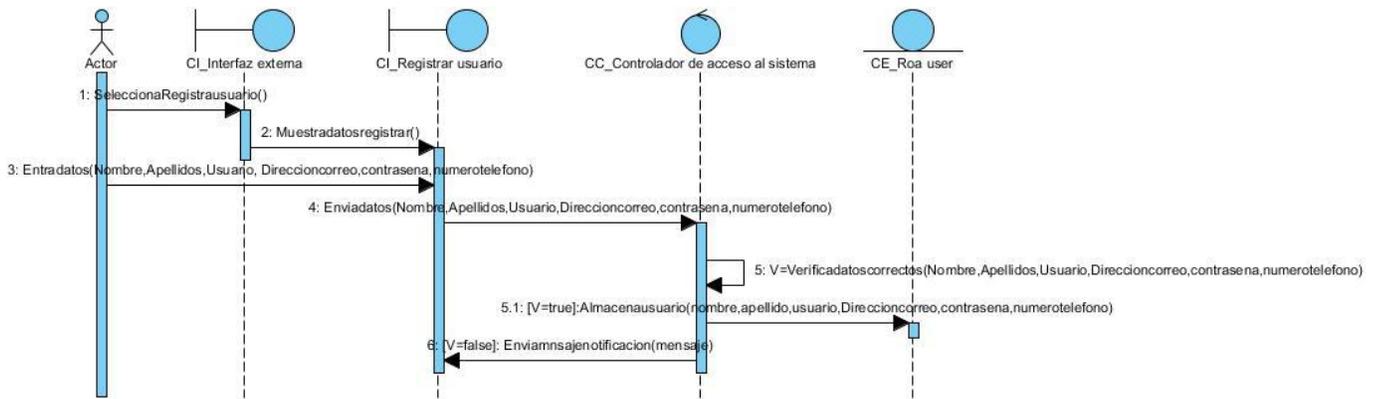


Figura 57-Diagrama de secuencia “Registrar usuario”.

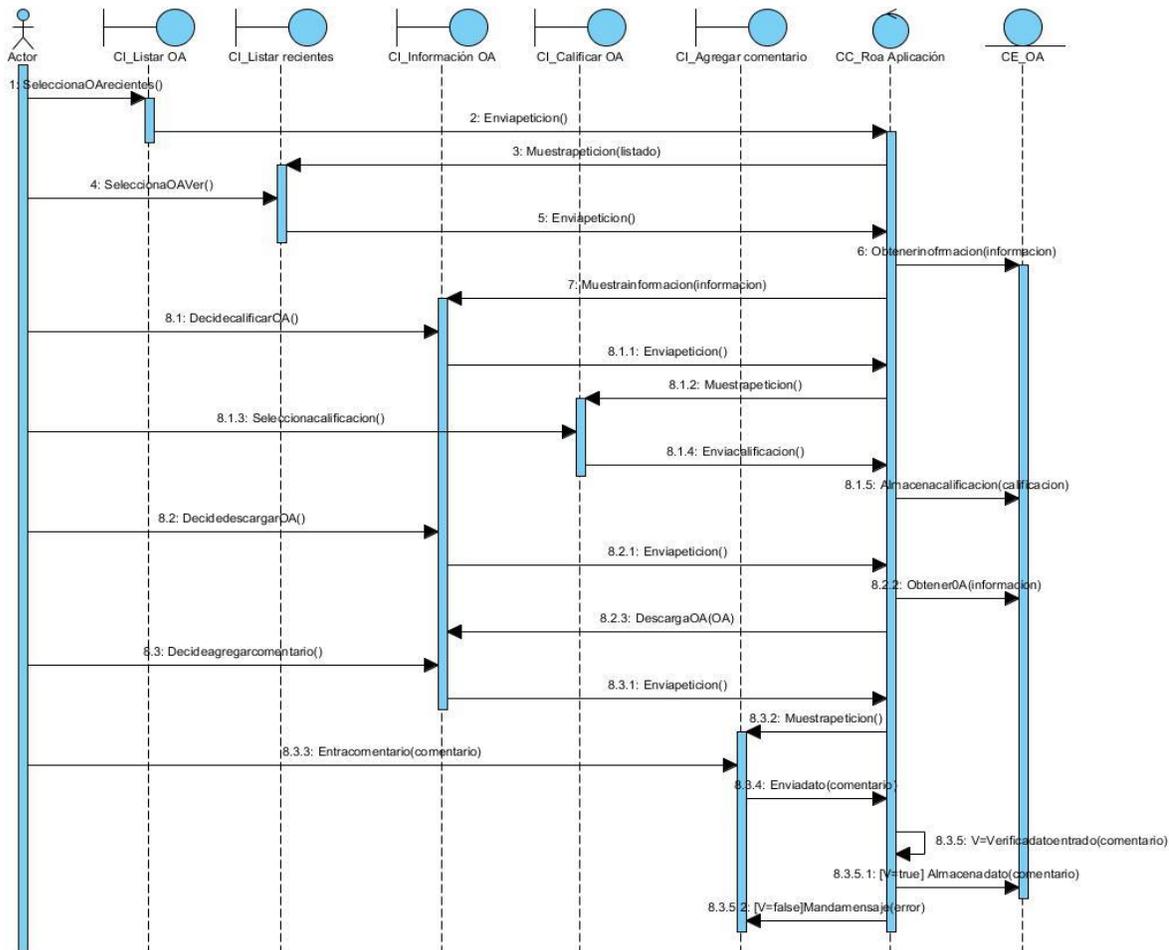


Figura 58-Diagrama de secuencia “Listar OA (recientes)”.

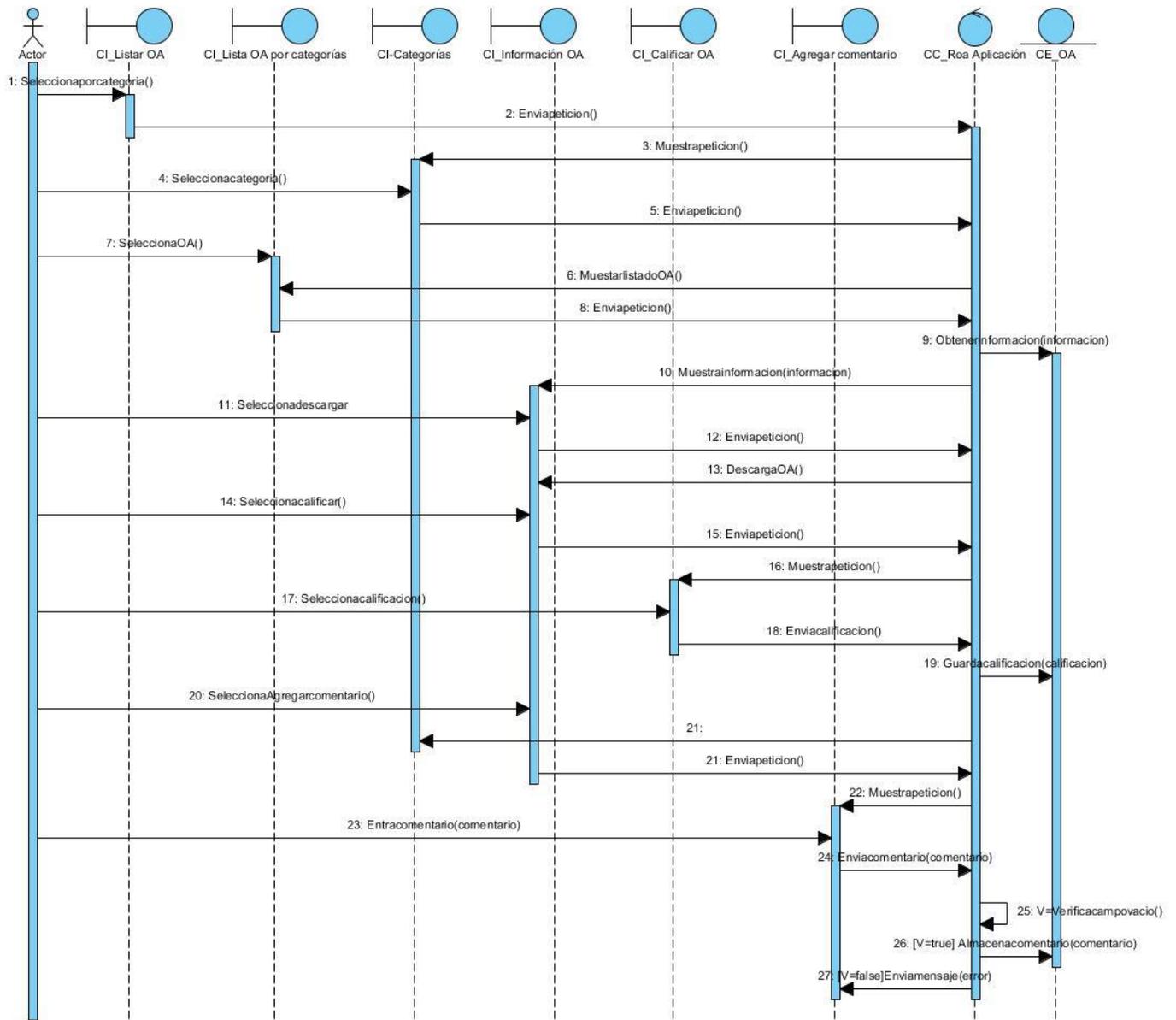


Figura 59-Diagrama de secuencia “Listar OA (por categoría)”.

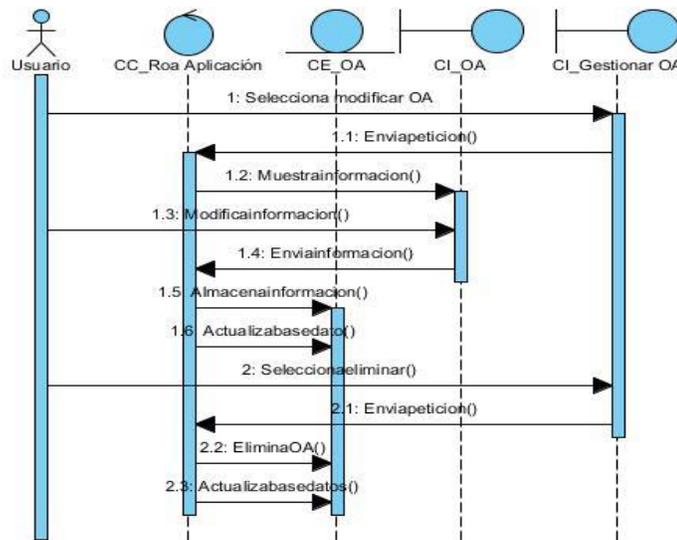


Figura 60-Diagrama de secuencia “Gestionar OA”.

Anexo 5: Resultado de la validación de los requerimientos

Revisores:

R1 Cindy Santos Salgado

R2 Irina Ivis Santiesteban Pérez

R3 Yuleisy Gonzalez Pérez

Especificidad

Para la evaluación de la especificidad de los requisitos cada revisor emitió su criterio, lo que arrojó como resultado que los revisores no consideraron específico dos de los requisitos funcionales escritos, por lo que fue necesario modificarlos para un mejor entendimiento de los mismos.

Para obtener el resultado final de los restantes indicadores es necesario indicar en la columna Resul, si está correcto (C) o incorrecto (I) el requerimiento de acuerdo al indicador, al menos dos revisores deben coincidir con un criterio.

Tabla 17: Resultado de la completión

Requerimientos	Revisores			Resultados
	R1	R2	R3	
RF ₁	I	C	C	C
RF ₂	I	I	C	I
RF ₃	C	C	I	C
RF ₄	I	C	C	C
RF ₅	I	C	C	C
RF ₆	I	C	C	C
RF ₇	C	I	C	C
RF ₈	C	I	C	C
RF ₉	C	C	C	C
RF ₁₀	C	I	C	C
RF ₁₁	C	C	C	C
RF ₁₂	C	C	C	C
RFN ₁	C	C	I	C
RFN ₂	I	C	C	C
RFN ₃	C	I	I	I
Total correctos				13
Total incorrectos				2

Tabla 18: Resultado de la corrección

Requerimientos	Revisores			Resultados
	R1	R2	R3	
RF ₁	C	C	I	C
RF ₂	C	C	I	C
RF ₃	C	I	C	C

RF ₄	C	C	I	C
RF ₅	I	C	C	C
RF ₆	I	C	I	I
RF ₇	C	C	C	C
RF ₈	C	I	C	C
RF ₉	C	C	C	C
RF ₁₀	I	C	C	C
RF ₁₁	I	C	I	I
RF ₁₂	C	C	C	C
RFN ₁	C	I	C	C
RFN ₂	C	C	C	C
RFN ₃	C	I	I	I
Total correctos				12
Total incorrectos				3

Tabla 19: Resultado de la comprensión

Requerimientos	Revisores			Resultados
	R1	R2	R3	
RF ₁	C	C	C	C
RF ₂	C	C	C	C
RF ₃	C	C	C	C
RF ₄	C	I	I	I
RF ₅	C	I	C	C
RF ₆	C	C	I	C
RF ₇	I	C	C	C
RF ₈	I	C	I	I
RF ₉	C	C	I	C
RF ₁₀	I	C	C	C
RF ₁₁	C	I	C	C

RF ₁₂	I	C	I	I
RFN ₁	C	I	C	C
RFN ₂	C	I	C	C
RFN ₃	I	I	C	I
Total correctos				11
Total incorrectos				4

Tabla 20: Resultado de la consistencia interna

Requerimientos	Revisores			Resultados
	R1	R2	R3	
RF ₁	C	C	C	C
RF ₂	I	C	C	C
RF ₃	C	C	C	C
RF ₄	C	C	C	C
RF ₅	C	I	C	C
RF ₆	C	C	I	C
RF ₇	I	C	C	C
RF ₈	I	C	C	C
RF ₉	C	C	I	C
RF ₁₀	I	C	C	C
RF ₁₁	C	I	C	C
RF ₁₂	C	C	C	C
RFN ₁	C	I	C	C
RFN ₂	C	I	C	C
RFN ₃	C	I	C	C
Total correctos				15
Total incorrectos				0

Tabla 21: Resultado de la consistencia externa

Requerimientos	Revisores			Resultados
	R1	R2	R3	
RF ₁	C	C	C	C
RF ₂	C	I	C	C
RF ₃	C	C	C	C
RF ₄	C	C	C	C
RF ₅	C	I	C	C
RF ₆	C	C	I	C
RF ₇	C	I	C	C
RF ₈	I	C	C	C
RF ₉	C	C	I	C
RF ₁₀	I	C	C	C
RF ₁₁	C	I	C	C
RFN ₁	C	I	C	C
RFN ₂	C	I	C	C
RFN ₃	I	C	C	C
Total correctos				15
Total incorrectos				0

Tabla 22: Resultado de la estabilidad

Requerimientos	Revisores			Resultados
	R1	R2	R3	
RF ₁	C	C	C	C
RF ₂	C	I	C	C

RF ₃	C	C	C	C
RF ₄	C	C	C	C
RF ₅	C	C	C	C
RF ₆	C	C	I	C
RF ₇	I	C	C	C
RF ₈	C	C	I	C
RF ₉	C	C	I	C
RF ₁₀	I	C	C	C
RF ₁₁	C	I	C	C
RFN ₁	C	I	C	C
RFN ₂	C	I	C	C
RFN ₃	I	C	C	I
Total correctos				14
Total incorrectos				1

Anexo 6: Resultado de la validación de los casos de uso

Revisores:

R1 Cindy Santos Salgado

R2 Irina Ivis Santiesteban Pérez

R3 Yuleisy Gonzalez Pérez

Tabla 23. Resultado de la completitud

Preguntas	Revisores			Resultado
	R1	R2	R3	
P1	SI	SI	SI	SI
P2	SI	NO	NO	NO

P3	NO	SI	SI	SI
P4	SI	SI	SI	SI
Total correctos				3
Total incorrectos				1

Tabla 24. Resultado de la comprensibilidad

Preguntas	Revisores			Resultado
	R1	R2	R3	
P1	SI	SI	SI	SI
P2	SI	SI	NO	SI
P3	SI	SI	SI	SI
P4	SI	NO	SI	SI
P5	SI	SI	SI	SI
Total correctos				5
Total incorrectos				0

Tabla 25. Resultado de la concisión

Preguntas	Revisores			Resultado
	R1	R2	R3	
P1	SI	SI	SI	SI
P2	SI	SI	NO	SI
P3	SI	SI	SI	SI
Total correctos				3
Total incorrectos				0

Tabla 26. Resultado de no trivialidad

Preguntas	Revisores			Resultado
	R1	R2	R3	
P1	SI	SI	SI	SI
P2	SI	NO	NO	NO
Total correctos				1
Total incorrectos				1