

**Diseño de un gestor de contenidos a partir de la
selección y uso de patrones de arquitectura: caso
de estudio**

Fabian Esteban Martinez Lopez
Noviembre, 2018

Facultad de Ingeniería
UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
Especialización en Ingeniería de software

Índice general

I	CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN	9
1.	Marco Teorico	11
1.1.	Arquitectura de software	11
1.2.	Patrones Arquitectonicos	13
1.2.1.	Multicapas	13
1.2.2.	Modelo Vista Controlador	13
1.2.3.	Orientada a servicios	13
1.2.4.	Micro servicios	14
1.3.	Vistas arquitectonicas	15
1.3.1.	Vista lógica	15
1.3.2.	Vista de desarrollo	15
1.3.3.	Vista del proceso	15
1.3.4.	Vista física	15
1.4.	Gestores de Contenido	16
1.4.1.	Usos de los Gestores de Contenido	16
1.5.	Arquitectura de Negocio y Archimate	18
2.	Marco de Conceptual	21
II	DESARROLLO DE LA INVESTIGACIÓN	23
3.	Papel del gestor de contenidos en la organización	25
3.1.	Arquitectura de la organización	25
3.1.1.	IEEE Sección Colombia	25
3.1.2.	Misión	26
3.1.3.	Visión	26
3.1.4.	Objetivos Organizacionales	26
3.1.5.	Organigrama	26

3.1.6. Vista de negocio del gestor de contenido	28
3.2. Estado actual del gestor de contenidos	31
3.3. Definición de Requerimientos	32
4. Diseño del CMS para mejorar la sinergia con la organización	33
4.1. La propuesta	33
4.2. Vistas	38
4.2.1. Vista Logica	38
4.2.2. Vista de desarrollo	40
4.2.3. Vista de Procesos	42
4.2.4. Vista Fisica	44
III CIERRE DE LA INVESTIGACIÓN	51

Índice de figuras

3.1. Organigrama IEEE Sección Colombia	27
3.2. Punto de vista de cooperación de los integrantes de la organización para la publicación de eventos	29
3.3. Punto de vista de funciones de las personas encargadas de la gestión del contenido de IEEE Sección Colombia	30
3.4. Modulos del gestor de contenido actual	31
4.1. Propuesta general para nuevo CMS	34
4.2. Login a través de SAML y autorización lo jerarquía	35
4.3. Modulo de vista general	36
4.4. propuesta para gestión de contenido y proceso de notificaciones	37
4.5. Vista lógica: Casos de Uso	38
4.6. Vista de componentes	41
4.7. Diagrama de flujo del ingreso de una persona al gestor de contenido	45
4.8. Diagrama de flujo del ingreso a la sección de lista de publicaciones de contenido	46
4.9. Diagrama de flujo de la sección edición de las publicaciones .	47
4.10. Flujo del registro de un token para notificaciones PUSH . . .	48
4.11. Diagrama de flujo para la tarea programada de generación de notificaciones.	49
4.12. Despliegue del gestor de contenidos	50

Introducción

La organización IEEE Sección Colombia tiene como objetivo principal promover la ingeniería como herramienta de cambio social. Para esto, la organización lleva a cabo diferentes actividades como congresos, capacitaciones entre otras. Para el éxito de dichas actividades es necesario la participación de la comunidad académica y profesional. La organización cuenta diferentes canales de comunicación por los cuales divulgar estas actividades para promover su participación.

Siendo parte de la filosofía de la organización estos canales son administrados por voluntarios. Personas que dedican tiempo libre para apoyar el desarrollo de la organización. Dando como resultado en ocasiones los canales no sean efectivos ya que los voluntarios no siempre cuentan con el tiempo que requiere la organización, principalmente en labores transversales a las diferentes unidades de esta.

El presente documento plantea el diseño de un gestor de contenido. Permitiendo reducir la cantidad de procesos que lleva acabo los voluntarios y así ayudar a que la organización cumpla sus objetivos. Para definir el diseño se hizo un proceso de obtención de requerimientos a partir de la arquitectura actual de la organización. Dando como resultado la necesidad de mantener un software heredado y plantear procesos que reduzcan los tiempos de generación de contenido aumentando a su vez los canales por los que se distribuye dicha información. Con estos requerimientos se planteo una propuesta de arquitectura siguiendo patrones de arquitectura actualmente muy utilizados y se presentaron 4 vistas como parte de la etapa inicial del desarrollo del gestor de contenido.

Parte I

**CONTEXTUALIZACIÓN
DE LA INVESTIGACIÓN**

Capítulo 1

Marco Teorico

1.1. Arquitectura de software

Hoy en día hay una gran cantidad de definiciones de arquitectura de software que varían mucho entre cada autor cambiando los alcances y límites de esta. Siendo una de las definiciones más conocidas la de Clementes: “La arquitectura de software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se la percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para alcanzar la misión del sistema. La vista arquitectónica es una vista abstracta, aportando el más alto nivel de comprensión y la supresión o diferimiento del detalle inherente a la mayor parte de las abstracciones” [19]. Aunque hay un consenso general entre las definiciones y es que la arquitectura es la representación a grandes rasgos de su estructuras y componentes entre ellos como trata de definirlo el estándar 1471 “La fundamental organización de un sistema encarnado en sus componentes, las relaciones entre cada uno de ellos y los principios que guían el diseño y su evolución” [1].

El objetivo de la arquitectura es prenotar a partir de diagramas simples requerimientos, reglas de negocio y el comportamiento para así tener éxito en el desarrollo del software [16].

Para el desarrollo de aplicaciones web las arquitecturas de desarrollo clásicas requieren ciertas modificaciones ya que el ambiente en el que se ejecutan es diferente. En la arquitectura web ya hay unos componentes ya de finidos como lo son el navegador, el servidor de aplicaciones, el servidor de bases de datos y los objetos multimedia. Además de eso, cada componente puede llegar a tener su propia arquitectura. La representación gráfica de la

interacción entre los componentes principales de una aplicación web a través de un diagrama arquitectónico se puede considerar la arquitectura web de la aplicación [18]

En su libro "Clean Architecture" Martin Flower propone que independientemente de los tipos de arquitectura estos deben seguir los cuatro principios SOLID del modelo orientado a objetos los cuales son[15]:

- Responsabilidad simple (S): Cada componente de la arquitectura debe tener una única responsabilidad en la organización
- Abierto-cerrado (O): Todo artefacto de software debe estar abierto a la extensión y cerrado a la modificación.
- Sustitución de Liskov (L): Establece que si el comportamiento del objeto A puede estar contenido por el comportamiento del objeto B. Entonces el objeto A es subconjunto del objeto B
- Segregación de interfaces (I): Definido por el mismo Martin Flower la segregación plantea que es preferible tener muchas interfaces con funciones simples que una sola interfaz con muchas funciones.
- Inversión de dependencias (D): Plantea reducir el acoplamiento entre los diferentes componentes para que de esta manera la modificación de uno no afecte a los otros.

1.2. Patrones Arquitectonicos

Aparte de los principios SOLID en la industria han surgido una infinidad de propuestas de arquitectura para diferentes soluciones. A continuación se describen de manera general unas de las arquitecturas más comunes y utilizadas hoy en día.

1.2.1. Multicapas

Se basa en una distribución jerárquica de componentes, típicamente separado en tres capas, presentación, lógica y de datos pero desacuero a la complejidad de la aplicación puede haber más o menos capas [20]. Es una de las arquitecturas más usadas y que para aplicaciones web permite diferenciar claramente varias de estas capas a partir de los componentes web navegador (capa de presentación), servidor de aplicaciones (capa lógica) y servidor de base de datos (capa de datos). Aparte de las capas se plantean niveles que definen la forma física en que se distribuyen las aplicaciones [16].

1.2.2. Modelo Vista Controlador

Otro patrón muy usado en las arquitecturas web es el modelo vista controlador. Su uso principalmente es para el manejo de interfaces de usuario ya que separa lo que observa el usuario (vista), con lo que interactúa (controlador) y los procesos lógicos que ocurren a partir de esas interacciones [3]. Esta arquitectura también puede ser considerada dependiendo el nivel de profundidad un patrón de diseño ya que puede definir la estructura de las clases que integran un elemento del producto.

1.2.3. Orientada a servicios

La arquitectura orientada a servicios propone que la aplicación un conjunto de servicios a partir de una interfaz que permitan ser invocadas publicadas y descubiertas. [5]. El objetivo de la arquitectura orientada a servicios está en “separar la lógica de integración de negocio de la implementación, para que el desarrollador de integración pueda centrarse en ensamblar una aplicación integrada en lugar de hacerlo en los detalles de la implementación. Para lograrlo, se crean componentes de servicio que contienen la implementación de servicios individuales necesarios para los procesos de negocio. El resultado es una arquitectura de tres capas: lógica de integración de negocio, componentes de servicio e implementación” [4]

1.2.4. Micro servicios

La arquitectura a partir de micro servicios es bastante reciente (2010) y hoy en día se encuentra presente en el desarrollo de grandes compañías como Amazon, Uber o Netflix. Esta arquitectura propone la separación de los procesos lógicos en mini aplicaciones que se manejan de manera autónoma e interactúan entre ellas a partir de APIs pudiéndose independizar tecnologías, infraestructura e incluso aislando errores. Para el uso de micro servicios se requiere de un coreógrafo que coordine las interacciones entre los servicios [9]. Esto último requiere un gran coste a nivel de gestión por eso se considera que la arquitectura de micro servicios debe plantearse para modelos de negocio a gran escala [21]

1.3. Vistas arquitectonicas

Como se mencionó anteriormente la arquitectura viene ligada de una representación gráfica del sistema y por eso es necesario plantear un lenguaje de modelado que permita entender y a su vez implementar la arquitectura de manera correcta. Para esto [14] planteo ver la arquitectura de software a partir de 4 vistas inter relacionadas a partir de escenarios estas vistas son:

1.3.1. Vista lógica

Encargada de presentar los requerimientos funcionales de la aplicación y es usada para hacer pruebas funcionales.

1.3.2. Vista de desarrollo

Muestra la estructura del programa a nivel de código y es principalmente para los líderes técnicos y los programadores también permite definir pruebas a nivel unitario

1.3.3. Vista del proceso

Esta vista presenta el proceso de integración del sistema, su implementación e instalación se utiliza para evaluar procesos concurrentes o de escalabilidad.

1.3.4. Vista física

Muestra cómo se implementa el software dentro del hardware, elementos de comunicación y dispositivos requeridos para su implementación.

1.4. Gestores de Contenido

Un gestor de contenidos (Content Management System, CMS) es un set de tecnologías que ayudan a crear, personalizar, publicar y distribuir contenido a lo largo del ciclo de vida [20]. Los CMS fueron pensados para que las personas que poco conocimiento técnico pudieran tener la capacidad de generar contenido web sin tener que pasar por un equipo de desarrollo. En otras palabras, CMS brinda la oportunidad de publicar, editar y modificar contenido CMS por parte de usuarios no técnicos que no requieren conocimiento de programación o evitan la necesidad de codificación manual [6].

Hoy en día gracias al auge de internet existen una gran cantidad de CMS multi propósito para los cuales basta con unas pocas configuraciones para iniciar a editar e ingresar información. Estos CMS multi propósito se basan en el desarrollo de temas, widgets y plugins de terceros para brindar una personalización específica de acuerdo a la necesidad de los usuarios. Generando como dice [22] “que muchas empresas en el mismo dominio comercial tienden a publicar información similar organizada de manera similar”. Esto hace pensar en la posibilidad de los CMS con propósito específico puedan ser mejor para las empresas ya que se puede adaptar y citando a [7] “Las organizaciones dedicadas al diseño de sistemas están abocadas a producir diseños que son copias de las estructuras de comunicación de dichas organizaciones”. Así cada organización a nivel de sistemas le será más fácil manejarlos en cuanto estos se parezcan más a la estructura de comunicación de la organización.

1.4.1. Usos de los Gestores de Contenido

Las empresas pueden utilizar CMS para lograr muchos objetivos técnicos, comerciales y operativos. Los siguientes son algunos de los beneficios clave de adoptar un CMS:

- Reutilización de contenido y plantilla: CMS facilita la reutilización del contenido mediante la creación y publicación de fragmentos de contenido estructurados y modulares. Plantillas de autoría y publicación de contenido también pueden ser reutilizadas para crear varias secciones de sitio.
- Separación de contenido y presentación: CMS separa el contenido principal de su presentación. El contenido principal se puede almacenar en formato estándar y luego podemos agregar presentaciones (usando plantillas de presentación).

- **Publicación eficiente:** Los CMS pueden proporcionar publicación de contenido eficiente y más fácil en diferentes idiomas.
- **Los administradores pueden usar una interfaz de usuario más amigable** para modelar el flujo de trabajo de publicación con una mínima dependencia de los equipos técnicos.
- **Información coherente:** En un ecosistema de CMS se proporciona contenido coherente y que se puede encontrar a través del uso de metadatos, clasificación, taxonomía y arquitectura de información intuitiva. El contenido se puede usar en toda la empresa.
- **Duplicación de información reducida:** CMS almacena toda la información en un único repositorio, lo que reduce el riesgo de duplicación de información en las unidades de negocio. Esto ayuda a disminuir los costos de mantenimiento y las tasas de error.
- **Costos reducidos:** CMS puede ayudar a reducir los costos en múltiples frentes: reducción del costo de publicación debido a la reutilización del contenido, reducción del costo de publicación debido a la publicación eficiente y menor mantenimiento del sitio gracias a procesos de autoservicio y de negocios.
- **Aumento del tráfico del sitio:** podemos crear una experiencia de usuario atractiva e interactiva utilizando CMS. Esto aumentaría el tráfico del sitio y puede involucrar activamente a los usuarios del sitio.
- **Repositorio central:** CMS forma el repositorio de contenido centralizado. CMS puede consolidar contenido de varias fuentes empresariales y puede actuar como punto único de acceso para todos los sistemas empresariales.
- **Acceso de usuario de interfaz amigable para el negocio:** CMS ofrece una interfaz fácil de usar que una empresa puede usar para realizar diversas actividades críticas para el negocio, tales como publicación, creación de sitios, administración de flujos de trabajo, administración de sitios, acceso

1.5. Arquitectura de Negocio y Archimate

Con el auge de las tecnologías de información (TI) es cada día más importante el papel de la información en el modelo de negocio y es de particular importancia entender como estas tecnologías juegan un papel dentro de la organización [10]. Para entender el papel de la actual plataforma en la organización de estudio se decidió iniciar por evaluar la arquitectura de la organización y donde es fundamental el papel del gestor de contenidos. Para esto el estándar ArchiMate ofrece un lenguaje integrado para el modelado de arquitectura empresarial. Permitiendo la descripción y visualización de diferentes dominios de arquitectura, así como sus relaciones y dependencias subyacentes [13].

ArchiMate aboga por un enfoque más flexible en el que los arquitectos y otras partes interesadas puedan definir sus propios puntos de vista sobre la arquitectura empresarial. En este enfoque, las vistas se especifican por puntos de vista. Los puntos de vista definen abstracciones en el conjunto de modelos que representan la arquitectura de la empresa, cada uno dirigido a un tipo particular de partes interesadas y que aborda un conjunto particular de inquietudes. Los puntos de vista se pueden usar para ver ciertos aspectos de forma aislada y para relacionar dos o más aspectos.[2]

ArchiMate propone en total 14 puntos de vista para obtener una visión completa de la arquitectura de una organización.

- Punto de vista de cooperación: En este punto de vista se presentan las relaciones entre los diferentes actores del proceso que lleva a cabo la organización.
- Punto de vista de la función del negocio: La función que desempeñan cada uno de los actores dentro de la organización.
- Punto de vista de proceso: Describe los procesos de negocio que se llevan a cabo dentro de la organización.
- Punto de vista de cooperación en el proceso: Este punto de vista nos presenta la interacción entre los diferentes actores dentro de la ejecución de los procesos dentro de la organización.
- Punto de vista del producto: Con este punto de vista se visualiza el producto final que entrega la organización y los procesos que permiten su desarrollo y entrega.

- Punto de vista del comportamiento de la aplicación: Este punto de vista pretende mostrar el papel de las aplicaciones dentro de la organización a través de su comportamiento.
- Punto de vista de cooperación de aplicación: Este punto de vista presenta la interacción entre las diferentes aplicaciones que son manejadas dentro de la organización.
- Punto de vista de estructura de aplicación: Este punto de vista muestra las interacciones entre los usuarios la aplicación y los datos de la organización.
- Punto de vista de uso de la aplicación: Como y cuando se usa la aplicación es lo que nos muestra esta vista.
- Punto de vista de infraestructura: Presenta como se distribuyen físicamente las aplicaciones de la organización.
- Punto de vista de uso de la infraestructura: Muestra quien y en donde se le da uso a las aplicaciones.
- Punto de vista de implementación y entrega: Describe el proceso de implementación y despliegue físico de las aplicaciones de la organización.
- Punto de vista de la estructura de la información: Describe el modelo de datos de la organización.

Capítulo 2

Marco de Conceptual

- Arquitectura de software: “La fundamental organización de un sistemas encarnado en sus componentes, las relaciones entre cada uno de ellos y los principios que guían el diseño y su evolución” [1]
- Aplicación Web: Se puede definir como una aplicación en la cual el usuario por medio de un navegador realiza peticiones a una aplicación remota a través de internet.
- Patrones de Arquitectura: Se puede definir un patrón de arquitectura como una serie recomendaciones para la implementación de una arquitectura. Limita y define en la estructura del sistema a grandes rasgos
- Vistas arquitectónicas: Son los diagramas que representan la arquitectura del sistema dependiendo del punto de vista necesario.
- UML (Lenguaje unificado de modelado): Es un lenguaje utilizado principalmente para la representación de sistemas informáticos. Tiene herramientas para representación a diferentes niveles y vistas.
- ADL (Lenguaje descriptivo de modelo arquitectónico): lenguaje de modelado de software enfocado en alto nivel especialmente usado para el diseño de arquitecturas.
- Requerimientos: Son las características funcionales y no funcionales que debe tener un software para cumplir con los requisitos del usuario.
- Contenidos: Es la información que la organización necesita presentar a los usuarios de su página web, pueden ser noticias, invitaciones a eventos o información de la organización.

- Sistema Gestor de Contenidos: es una aplicación generalmente web que permite gestionar el contenido que ven los usuarios de un sitio web.
- API (Interfaz de programación de aplicación): Una API es una interfaz a través de la cual una aplicación expone sus servicios generalmente protocolo REST.
- SSO (Single Sign On): Es una propuesta para la autenticación en diferentes plataformas mediante un solo proceso de registro.
- SAML (Security Assertion Markup Language): Es un lenguaje de comunicación segura a partir de XML y típicamente utilizado para aplicaciones SSO.
- Notificaciones PUSH: Son un tipo de notificaciones que se presentan como ventanas emergentes en celulares y navegadores actuales.

Parte II

DESARROLLO DE LA INVESTIGACIÓN

Capítulo 3

Papel del gestor de contenidos en la organización

3.1. Arquitectura de la organización

3.1.1. IEEE Sección Colombia

El IEEE (Instituto de ingenieros eléctricos y electrónicos) es una organización internacional dedicada a promover la ingeniería como dice su eslogan “La mayor organización técnica profesional del mundo para el avance de la tecnología” [12] promueve el desarrollo de actividades de generación y divulgación investigación al rededor del mundo, además de proveer para sus miembros una diversidad de beneficios y espacios para hacer networking. Para desempeñar esta labor el instituto divide su trabajo a nivel regional en 10 regiones [11] y a su vez estas regiones se dividen en secciones que en el caso de Latinoamérica (región 9) se encuentra dividido en 35 regiones y una de ellas es IEEE Sección Colombia (La Sección). Esta sección está encargada de coordinar y organizar todas las actividades que promueve el instituto a nivel Colombia al igual que proveer un enlace entre los miembros Colombianos del instituto con este. Entre las labores que desempeña La Sección está el apoyo en la organización de eventos de divulgación científica, Actividades que promuevan la interacción entre sus miembros y generación de beneficios para los miembros de esta organización. Actualmente la página de La Sección es la plataforma para informar de estas actividades y eventos que organiza la sección.

Para elaborar la arquitectuda de la organización es necesario identificar su misión, visión, objetivos y la estructura organizacional.

3.1.2. Misión

El propósito central del IEEE es fomentar el desarrollo de la tecnología y la excelencia en beneficio de la humanidad.

3.1.3. Visión

IEEE será esencial para la comunidad técnica mundial y para los profesionales técnicos de todo el mundo, y será universalmente reconocido por las contribuciones de la tecnología y de los profesionales técnicos en la mejora de las condiciones globales.

3.1.4. Objetivos Organizacionales

- Expandir y permitir comunidades dinámicas, ágiles, flexibles y diversas para ayudar a personas de todo el mundo a compartir, colaborar, establecer redes, debatir y relacionarse entre sí.
- Proporcionar foros técnicamente vitales para la discusión, desarrollo y la diseminación de conocimiento autorizado relacionado con las tecnologías tradicionales, al mismo tiempo que concentramos más nuestros recursos en servir a los profesionales que trabajan en tecnologías emergentes y disruptivas.
- Dirigir los esfuerzos humanitarios en todo el mundo para utilizar la tecnología para resolver los problemas más desafiantes del mundo.
- Aprovechar la información relacionada con la tecnología de IEEE para proporcionar a los gobiernos, ONG y otras organizaciones y al público con recomendaciones innovadoras y prácticas para abordar los problemas de política pública.

3.1.5. Organigrama

El principal elemento de una organización son sus integrantes. En el caso de IEEE Colombia esta está constituida por dos comités el ejecutivo y el operativo y a su vez coordina otras unidades organizacionales en la imagen 3.1 se puede observar más detalle que personas desempeñan esa labor.

Comité Ejecutivo

Esta integrado por el presidente, el vicepresidente, el secretarios y el tesorero. Entre sus labores están la gestión y coordinación de los eventos

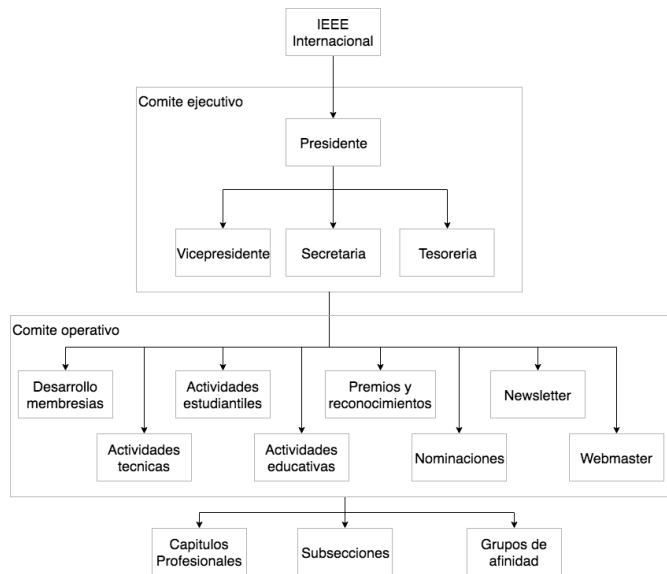


Figura 3.1: Organigrama IEEE Sección Colombia

desarrollados por la organización, brindar apoyo en gerencia y gestión de recursos en eventos organizados por otras unidades, definir el plan de trabajo y los objetivos anuales y gestionar, administrar y reporte de los recursos que administra la organización.

Comité Operativo

Como su nombre lo indica es el encargado de los procesos operativos de la organización como lo son el trabajo logístico, la gestión dentro de los eventos y los procesos de divulgación. Esta constituido por los equipos de actividades educativas, técnicas y estudiantiles. Una persona de premios y reconocimientos encargada de galardonar a los miembros de la organización por su labor en esta o en el que hacer de la ingeniería. Una persona encargada de nominaciones que coordina todos los procesos de elecciones de comités, tanto comité ejecutivo como los de las diferentes unidades dentro de la organización. Una persona encargada del desarrollo de membresias que brinda una labor de apoyo y gestión en pro de promover la compra y renovación de las membresias. El newsletter y el webmaster que se encargan de difundir todo el contenido generado por los diferentes integrantes de la organización incluyendo unidades organizacionales.

Otras Unidades

Finalmente en el organigrama (figura 3.1) se muestran los Capítulos profesionales, Subsecciones y Grupos de afinidad. Estos tres tipos de unidades tienen la particularidad respecto a las mencionadas anteriormente que manejan recursos económicos propios por lo que aunque se gestión y vigilancia están a cargo del comité ejecutivo estos cuentan con sus propios comités internos (ejecutivo y operativos).

3.1.6. Vista de negoció del gestor de contenido

Como se menciona anteriormente todas las unidades de la organización desempeñan un papel importante y cabe resaltar que en especial las actividades llevadas a cabo por el comité operativo en su mayoría requieren de divulgación ya sea que se desarrollen conferencias de temas de la industria (actividades profesionales), un congreso estudiantil (actividades estudiantiles) o un taller de capacitación a profesores (actividades educativas). Todos los comités acuden al web máster y el newsletter para poder divulgar dichas actividades. De esta manera, en la figura 3.2 se puede observar que los generadores de la información (capítulos, subsecciones, grupos de afinidad y comités) requieren de informar por medios no oficiales a los encargados de la distribución de dicha información (webmaster y newsletter). Estas dos personas en ocasiones pueden llegar a convertirse en cuellos de botella cuando hay una gran cantidad de actividades por se promocionadas ya que al igual que las personas en los otros comités el webmaster y el newsletter son voluntarios que comparten sus funciones dentro de IEEE con otras actividades.

De esta menra si se quiere reducir el impacto del cuello de botella se puede optar por permitirle a las diferentes unidades administrar su propio contenido. Este tarea genera nuevos problemas como asegurar la autenticidad de las personas que ingresan y que aunque el contenido sea diferente debe mantener armonía con la actual plataforma.

Los medios de publicación principales usados por la organización son el correo electrónico, la fanpage de facebook y la pagina oficial de la organización. Para lo cual el news lettter y el web master están encargados de su publicación y mantenimiento; adicionalmente, ellos tienen la labor de gestión IT y estar atentos a interactuar con los usuarios a través de los mismos medios como se puede observar en la figura 3.3.

El proceso actual de divulgación por medio de correo electrónico actualmente se hace a partir de listas que provee IEEE a el newsletter de acuerdo al tipo de publicación y requiere de herramientas especializadas para el en-

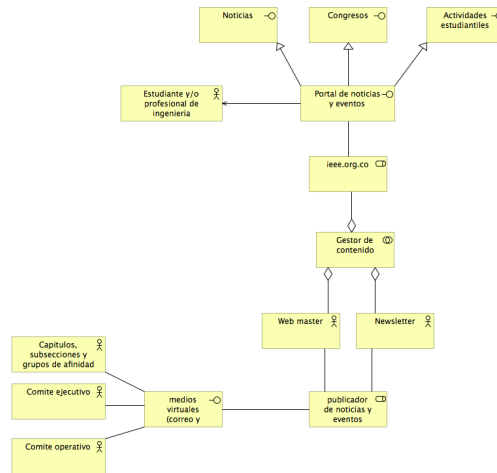


Figura 3.2: Punto de vista de cooperación de los integrantes de la organización para la publicación de eventos

vió y gestión de correos de manera masiva. En el proceso de investigación previo no se encontró la manera de automatizar de manera correcta este proceso por tal razón al momento del diseño no se tiene en cuenta el envío masivo de correos electrónicos.

Para el proceso de divulgación por medios sociales se asume que se hace solamente por la red social Facebook de esta manera se encontró una API ?? que expone métodos para hacer publicación de contenido en su web desde un tercero este caso el gestor de contenido propuesto.

Finalmente dentro de las herramientas actuales para divulgar noticias e información actualizada se encuentran las notificaciones PUSH este tipo de notificaciones se encuentran dentro de los navegadores actuales y permite la generación de ventanas emergentes informando de contenido nuevo de pagina a la que la persona se suscriba. Como apoyo al objetivo de generar más herramientas de divulgación se propone dentro del gestor de contenidos la administración de notificaciones PUSH??.

Teniendo en cuenta que las figuras 3.2 y 3.3 representan de manera modular y organizada la función del gestor de contenidos, se procede al análisis de la actual plataforma para a partir de esta definir más requerimientos que pueda resolver el modelo propuesto.

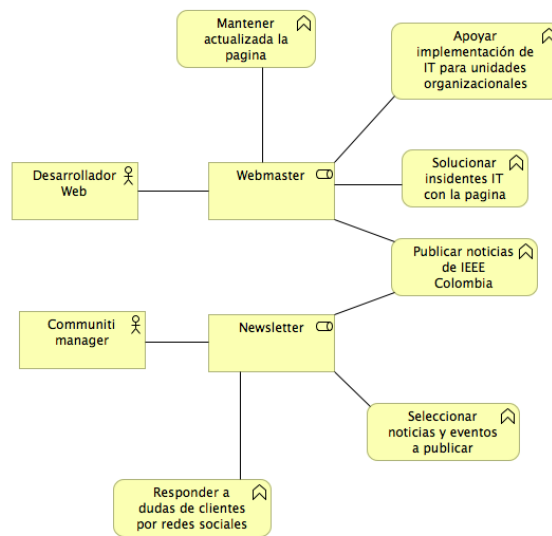


Figura 3.3: Punto de vista de funciones de las personas encargadas de la gestión del contenido de IEEE Sección Colombia

3.2. Estado actual del gestor de contenidos

Para poder generar una propuesta que permita mejorar los procesos de publicación de actividades y eventos en la organización es importante primero identificar las características y problemas del actual gestor de contenido.

Como se evidencia en la figura 3.2 el gestor actualmente es administrado por el webmaster y la publicación de los eventos la hace él, esto lleva a demoras en la publicación del contenido ya que es una labor de voluntariado y depende de la disponibilidad de tiempo del web master. Aunque el gestor actual tiene la opción de manejar multiples usuarios la gestión de estos usuarios tambien depende del webmaster ya que procesos como generación de nueva contraseña o registro no estan disponibles en la actual plataforma.

En la figura 3.4 Se observan los módulos con los cuenta el actual gestor de contenido en el cual no es claro donde se debe ingresar de acuerdo al contenido que se quiere gestionar. Módulos como secciones, modulo de eventos, categorías de eventos, gestor de menú, enlaces IEEE mundial, alianzas y eventos y membresia y servicios actualmente no se utilizan, adicionalmente otros como FAQ urls amigables no es claro su uso. IEEE Colombia y contactenos son actualizadas muy pocas veces y finalmente imagenes en el home, conferencias y eventos y noticias son las que más se acualizan pero muchas veces se requiere replicación de contenido en las tres.

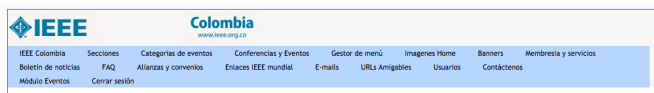


Figura 3.4: Modulos del gestor de contenido actual

Un dealle más de su funcionamientos es que si se quiere poner una imagen en la publicación es necesario primero subirla al servidor y luego tomarla del servidor.

Al igual que los gestores de contenido actuales en el contenido que se crea muchas veces no se ve igual en la edición como en la publicación.

Las publicaciones tambien se distribuyen por otros medios (facebook y correo electronico) pero esta labor es manual llevada acabo por el newsletter (3.3)

3.3. Definición de Requerimientos

El diagnóstico de la organización a nivel de arquitectura empresarial para permitió definir el contexto en el que se aplica la solución, esto quiere decir que se puede identificar a las personas de la organización que generan el contenido y las que lo administran de esta manera el contenido es generado principalmente por las unidades organizacionales las cuales generalmente tienen a uno o más miembros encargados de la labor de divulgación por tal razón es requerido tener un control del ingreso agrupado por miembros pertenecientes a cada respectiva unidad.

A partir de los mencionado en las dos secciones anteriores se definieron los siguientes requerimientos que debe tener el nuevo gestor de contenido propuesto:

- Debe ser aislada de la actual plataforma pero debe mantenerse los datos como se ingresan actualmente
- Debe permitir múltiples usuarios que sean miembros y ieee pertenecientes a los comites
- Debe tener un proceso de login para asegurar que las personas que publican sean de la organización
- Debe manejar roles
- Debe poder separarse el contenido de acuerdo a si es congreso, noticias o evento estudiantil
- Debe permitir generar notificaciones para navegadores
- Debe permitir generar Posts para facebook
- Un post de noticia genera una nueva noticia en la pagina
- Un post de un evento puede generar noticia, un banner en el home y un evento en la sección de eventos
- Un post de un evento estudiantil solo va en la sección de eventos estudiantiles

Capítulo 4

Diseño del CMS para mejorar la sinergia con la organización

4.1. La propuesta

Con los requerimientos obtenidos se procede al diseño de la solución. Para esto primero se define como un componente independiente a la actual plataforma como se puede observar en la figura 4.1. De esta manera, El gestor incluye base de datos para la configuraciones que requiera, debe contar con conectividad a la nube de IEEE.org para poder hacer login a través de protocolo SAML, a las APIs ofrecidas por facebook para poder automatizar la generación de publicaciones a los navegadores para el envío de notificaciones a través de mensajes PUSH para informar cada vez que hay un nuevo contenido. En la figura 4.1 se aprecian los elementos que se integran con el CMS propuesto.

De acuerdo a los requerimientos obtenidos y siguiendo el principio de responsabilidad única nombrado anteriormente se definen las responsabilidades que tiene el CMS como:

- Login seguro y sencillo.
- Visualizar contenido actual.
- Generación y actualización de contenido.
- Notificación de nuevo contenido.

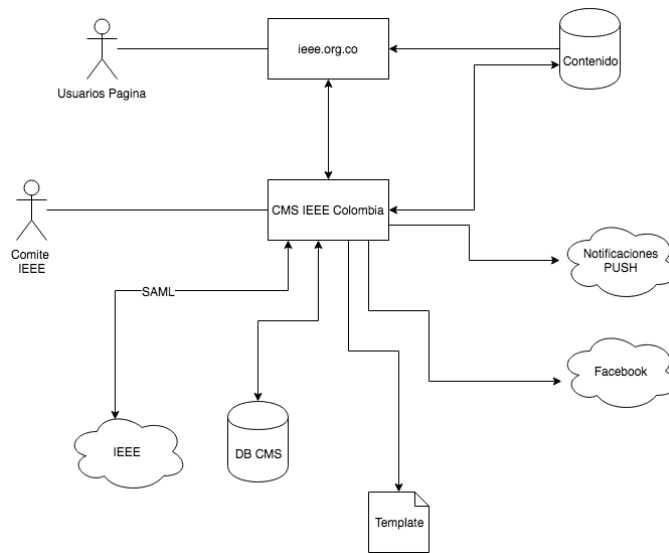


Figura 4.1: Propuesta general para nuevo CMS

Login seguro y sencillo

Para el login como se mencionó anteriormente se diseña un login utilizando protocolo SAML. Este protocolo permite implementar un concepto llamado Single Sign On que permite autenticar a una persona en varias instancias por medio de una sola identificación. Esta opción se tomó ya que para los miembros de IEEE en páginas como gmail y overleaf [?] se permite hacer login con el usuario propio de IEEE accediendo a beneficios diferentes sobre estos servicios. En el caso del gestor de contenido este login ayuda a verificar el estado de la membresía de quien lo usa obligando a que para acceder se requiera que sea un miembro activo, adicionalmente reduce la fricción del usuario con la plataforma ya que no requiere de recordar un usuario con una contraseña más.

Como se muestra en la figura 4.2. adicional al login con IEEE se agrega un módulo para autorización que permite validar de acuerdo su rol en la organización si puede o no publicar noticias en este para ello se propone unos roles dados por la estructura organizacional donde su posición puede publicar en ciertas secciones de la página o publicar cierto tipo de contenido.

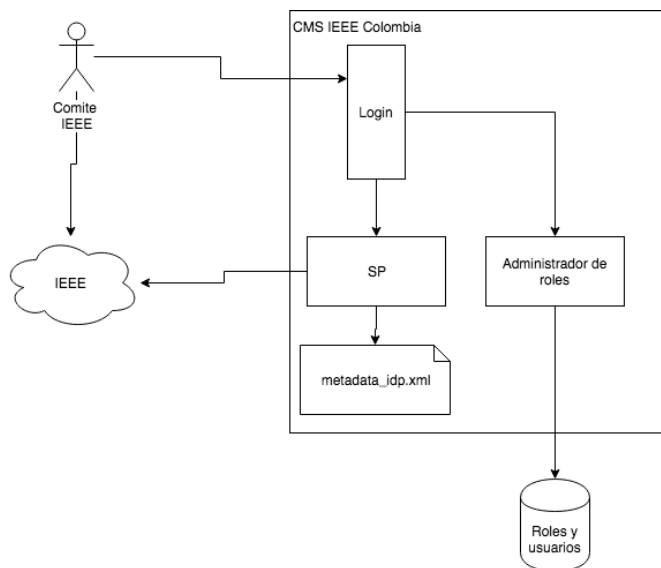


Figura 4.2: Login a través de SAML y autorización lo jerarquía

Visualización de contenido actual

En la figura 4.3 se presentan dos módulos que comprenden la página de inicio del CMS. En esta página se visualiza la lista del contenido actual de la página dividido en eventos y noticias. Adicionalmente en caso de manejar un rol administrador será posible ingresar a una sección donde se puedan administrar las cuentas que tienen permitida la generación de contenido.

Por último desde esta sección se accede al contenido existente para actualizarlo o eliminarlos y a las opciones para generar nuevo contenido.

Generación de contenido y notificaciones

En el último paquete agrupamos dos funcionalidades la edición de contenido y las notificaciones. La edición de contenido la separamos en dos elementos uno es la interfaz y el otro es el objeto encargado de la transformación. Mientras que en la parte de las notificaciones se tiene una cola de mensajes y un cliente de la respectiva herramienta de publicación.

La interfaz se plantea como una especie de fachada que permita independizar la información del proceso de almacenamiento, es decir que la tarea de definir como se va a almacenar la información de acuerdo a la estructura heredada será el componente transformador y así en futuras actualizaciones es este último el que se modificaría.

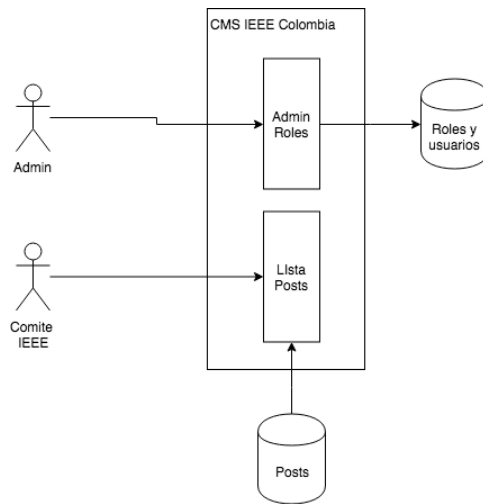


Figura 4.3: Modulo de vista general

Por otro lado se plantea que las publicaciones en terceros (Facebook y notificación PUSH) no se haga de manera directa evitando de esta manera concurrencia de información que pueda generar que se marque a la aplicación como SPAM.

La interfaz propuesta es un simple formulario donde los campos ya estan definidos de acuerdo al tipo de contenido que maneja la organización. De esta manera la información presentada en la pagina web ma a ser uniforme independiente de quien la genere.

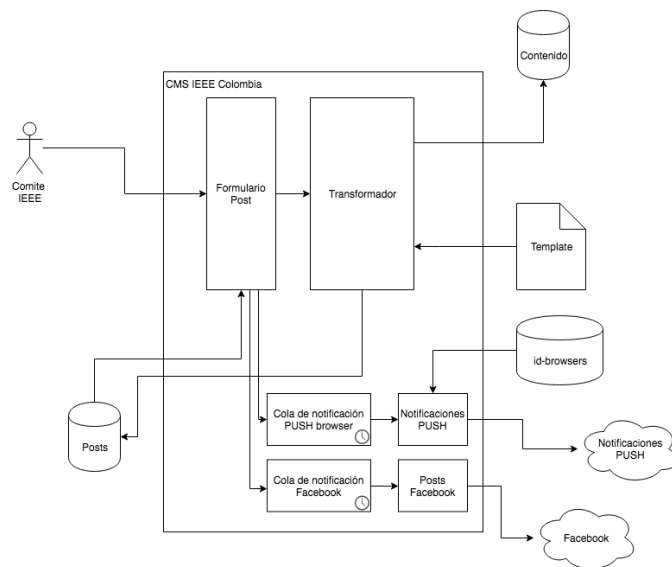


Figura 4.4: propuesta para gestión de contenido y proceso de notificaciones

4.2. Vistas

Teniendo en cuenta el modelo propuesto finalmente se procede a diseñar las diferentes vistas que cumplen este. Para ello se definen cuatro diagramas de acuerdo a las vistas de arquitectura propuestas.

4.2.1. Vista Logica

La primera vista presenta los requerimientos de manera que sean claros para tanto los usuarios como los el equipo de desarrollo. En la figura 4.5 se muestra un diagrama de casos de uso dividiendo los requerimientos de acuerdo a casos en los que la herramienta se desean implementar. De esta manera, se logra ver el papel de los actores con el proceso e identificar las responsabilidades en las que se pueden dividir los componentes.

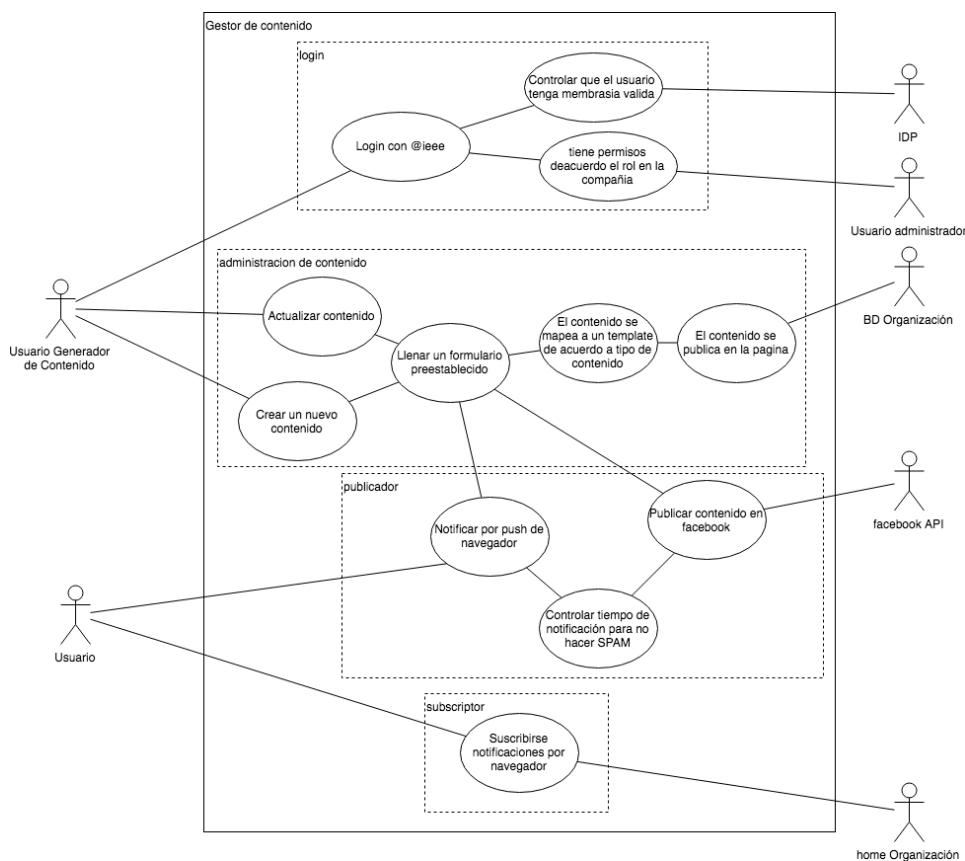


Figura 4.5: Vista lógica: Casos de Uso

En este diagrama se presentan siete actores que son los elementos externos al gestor de contenido con el que este, de esta manera:

- **Usuario Generador de contenido:** Este actor representa a un usuario de alguna de las unidades de la organización que desarrolla algún tipo de actividad que quiere reportar ya sea de tipo noticia o evento. Este usuario podrá ingresar a la plataforma a través de un login, podrá actualizar las publicaciones y podrá crear nuevas
- **Usuario:** es el consumidor del contenidos son todas la personas que ingresan a la pagina interesadas en las actividades y noticias de la organización. Si el usuario quiere estar al tanto de las ultimas actualizaciones de la pagina podrá suscribirse al las notificaciones PUSH para eso se registrar un token asociado a su navegador.
- **IDP: Identiti Provider** es el encargado de validar la identidad de usuario generador de contenido a través de protocolo SAML
- **Usuario Administrador:** Este es el usuario que tiene como labor administrar los roles de los usuarios generadores de contenido.
- **DB Organización:** Es la base de datos donde se ingresa el contenido que luego es consumido por la pagina web.
- **Home Organización:** Para permitir a los usuarios suscribirse a las notificaciones PUSH es necesario embeber un servicio de registro en la pagina principal de la organización.
- **API Facebook:** Es una API que expone Facebook para poder automatizar algunas de las tareas que normalmente se hacen desde Facebook.

Adicionalmente se agruparon casos de uso con una funcionalidad o responsabilidad especifica. Definiendo los siguientes conjuntos de casos de uso: login, administración de contenido, publicador, subscriptor.

- **El login:** agrupa todos los casos de uso relacionados con el proceso de ingreso seguro a la plataforma de administración de contenido.
- **El administrador de contenido:** se encarga de actualizar y crear el contenido, es el encargado de entender las publicaciones existentes y crear nuevas.
- **El publicador:** tiene la tarea publicar el contenido a través de facebook y notificaciones PUSH.

- El subscriptor: se embebe dentro de la pagina principal de la organización y su objetivo obtener los datos de los navegadores para poder hacer publicaciones PUSH.

4.2.2. Vista de desarrollo

Para la vista de desarrollo se presenta un diagrama de componentes, como se ve en la figura 4.6. Este diagrama me permite visualizar la solución a nivel de paquetes y componentes de software de esta manera separar en tres paquetes de acuerdo a las vistas por las que pasa el usuario (login, lista de contenido y gestor de contenido) y dos paquete de procesos que no son manejados por el usuario del gestor (notificador y registrador).

Login

Esta compuesto por tres artefactos encargados de administrar el login de manera efectiva, esta compuesto por los siguientes artefactos:

- User Agent SAML: se encarga de la comunicación con el IDP de IEEE para hacer el proceso de autenticación mediante SSO
- Rol Administrator: Encargado de autorizar a un usuario que existe en IEEE y adicionalmente haga parte de alguna de las unidades de la organización.
- Session Manager: es un administrador de sesión que mediante cookies en el navegador gestiona la sesión para todas las vistas.

Lista de contenido

Tiene como función el cargar el contenido que se encuentra actualmente en la pagina. Este contenido debe tener ciertas características como el tipo de contenido y estar organizado desde la ultima publicación, esta compuesto por los siguientes artefactos:

- List View: que es la vista desde la cual se visualiza la lista de publicaciones.
- Post List: Provee una interfaz de consulta entre un bojeo de datos y lo que necesita visualizar el usuario.

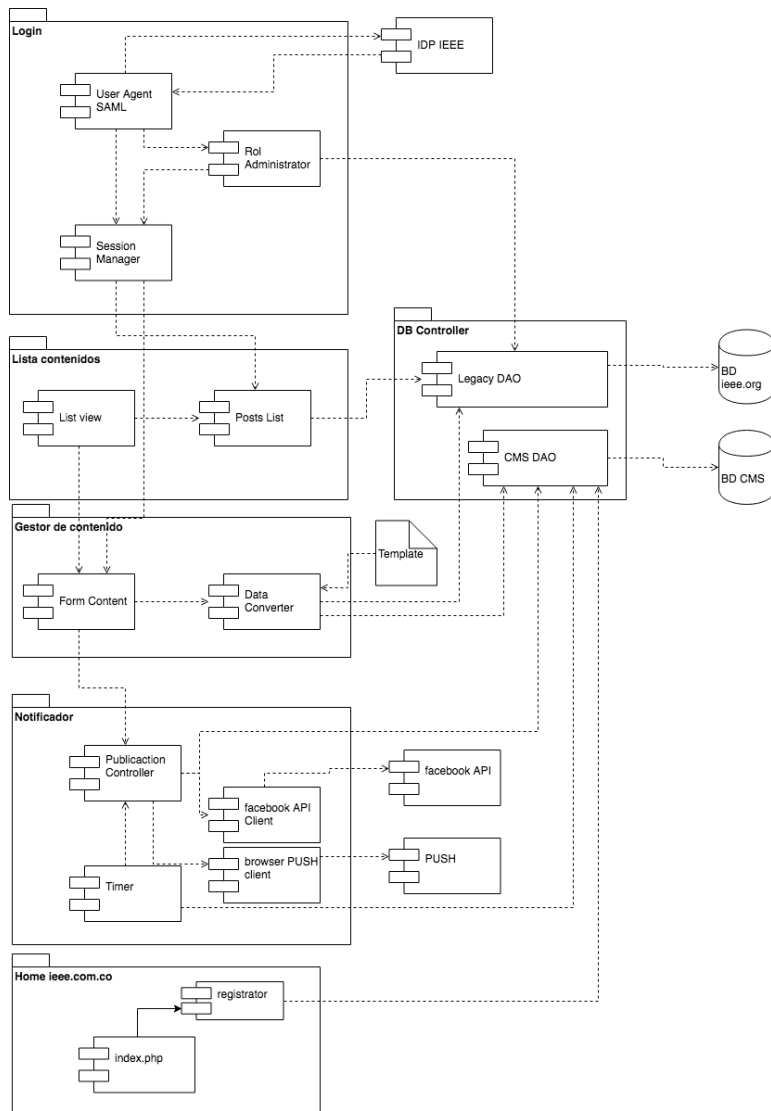


Figura 4.6: Vista de componentes

Gestor del contenido

Siendo el corazon del modelopropuesto esta integrado por dos elementos la vista como un formulario (Form Content) y el gestor de datos (Data Converter).

- Form Content: Presenta al usuario un formulario simple donde se in-

gresa solamente el contenido que este requiera modificar de acuerdo al tipo de contenido que necesite publicar

- **Data Converter:** Es el encargado de consultar, actualizar y crear el contenido en la base de datos de la organización. Para esto, es necesario que conozca el esquema de base de datos y adicionalmente a partir de un template construye el contenido de la publicación y de esta manera poder mantener un mismo estilo de publicación pero que sea definido por la organización.

Notificador

Son los artefactos que van a permitir al gestor las tareas adicionales de publicar en medios de terceros. Para este proyecto se contemplaron dos medios (publicación en Facebook y notificación PUSH en navegadores). Esta integrado por los siguientes artefactos:

- **Publication Controller:** Es el orquestador del proceso de publicación, este es activado por un evento de reloj y la idea es que valide si ya se envió la publicación por los canales, de lo contrario enviarla.
- **Cientes de Facebook y PUSH:** son dos componentes encargados de entender el protocolo de comunicación con los canales y de esta manera enviarles la publicación.
- **Timer:** es el artefacto encargado de consumir el Publication Controller periódicamente de acuerdo a una configuración preestablecida.

Registrador

Es un artefacto que se embebe dentro de la pagina principal de la organización para que cuando un usuario de la pagina ingrese pueda suscribir su navegador y de esta manera poder enviarle notificaciones PUSH.

4.2.3. Vista de Procesos

Para representar de manera clara como se usan los artefactos para cumplir con los requerimientos se construyo en la vista de proceso los diagramas de flujo que identifican las diferentes etapas en el uso del gestor, en la figuras 4.8, 4.8 y 4.9 presentan el comportamiento de un usuario al ingresar a las vistas del gestor.

En el caso de las figuras 4.11 y 4.10 presenta el proceso de notificación por los navegadores y facebook y el proceso de registro de un navegador en la pagina del home de la organización.

Login

Como se menciona anteriormente el proceso de Login se hace usando protocolo SAML. Para esto al ingresar al recurso del gestor (ieee.org.co/CMS) y no se encuentra una sesión iniciada este enseguida redirige al IDP y de acuerdo a su respuesta redirige a la vista de la lista de publicaciones o al home de la organización. En esta caso no hay una vista para el usuario por parte del gestor y la ejecución del proceso se hace de manera indirecta por la redirección entre los diferentes componentes.

Lista de publicaciones

El flujo de la figura 4.8 representa el comportamiento de la vista de la lista de publicaciones, donde se puede acceder a opciones de crear , actualizar y eliminar contenido. Para el caso del inicio cuando se carga la lista y la eliminación el artefacto Post List tiene la capacidad de ejecutar esas tareas. En el modelo propuesto no se contempla pero es necesario la implementación de una paginación en esta sección del gestor.

Edición de contenido

En la figura 4.9 se representa el proceso de publicar un contenido. Entonces, si la sesión se encuentra activa valida que dentro de la url vaya como variable el id de la publicación a actualizar de lo contrario asume que es una nueva. El formulario va a tener variaciones de acuerdo a si es una publicación de tipo noticia o si es un evento

- Noticia: Esta compuesta solamente por el titulo, una resumen y el cuerpo de la noticia.
- Evento: este puede ser de tipo estudiantil o no, pero independiente de este adicional a los elementos de la noticia también tiene lugar y fecha del evento y un banner asociado para poner en el home de la pagina.

Registro para PUSH

Como una funcionalidad adicional que se propone para el gestor esta el envío de notificaciones PUSH a los navegadores. Esto se hace mediante el uso

de workers que son herramientas disponibles en los navegadores para ejecutar procesos en paralelo, de esta manera si se quiere hacer una notificación PUSH se requiere de que el navegador este suscrito y que haya compartido un par de llaves con el backend de la aplicación y así mantener una comunicación segura [17]. La figura 4.10 representa el proceso.

Diagrama de flujo de notificaciones

Teniendo ya compartidas las llaves es posible establecer omunicación entre los navegadores y el servidor mediante notificaciones PUSH. El diagrama de la figura 4.11 representa el proceso de notificación endonde a partir de una tarea programada que se encuantra configurada en base de datos se envía notificación PUSH para las publicaciones que no se hayan notificado. Esto se hace con la idea de noa saturar al usuario con notificaciones en caso de que se generen muchas publicaciones en un corto tiempo. Adicional a las nootificaciones PUSH el gestor tambien publica en Facebook mediante una API que ofrecen ellos llamad Graph API que permite hacer publicaciones en la fanpage por desde un tercero (el gestor de contenido) [8]

4.2.4. Vista Fisica

En esta sección observamos donde se desplegarian fisicamente los componentes que componen todo el gestor de contenido para ellos en la figura 4.12 podemos observar fisicamente tres elementos fundamentales los browser del usuario, el servidor host donde se aloja tanto la pagina de la organización como el gestor de contenidos y finalmente la base de datos. Adicionalmente en diagrama vemos componentes de terceros que se van a utilizar como los browser de los usuarios a los que se les envia notificaciones push, los servidores de facebook con los que se comuocaria el API y el IDP con el que los usuarios generadores de contenido validan su identidad.

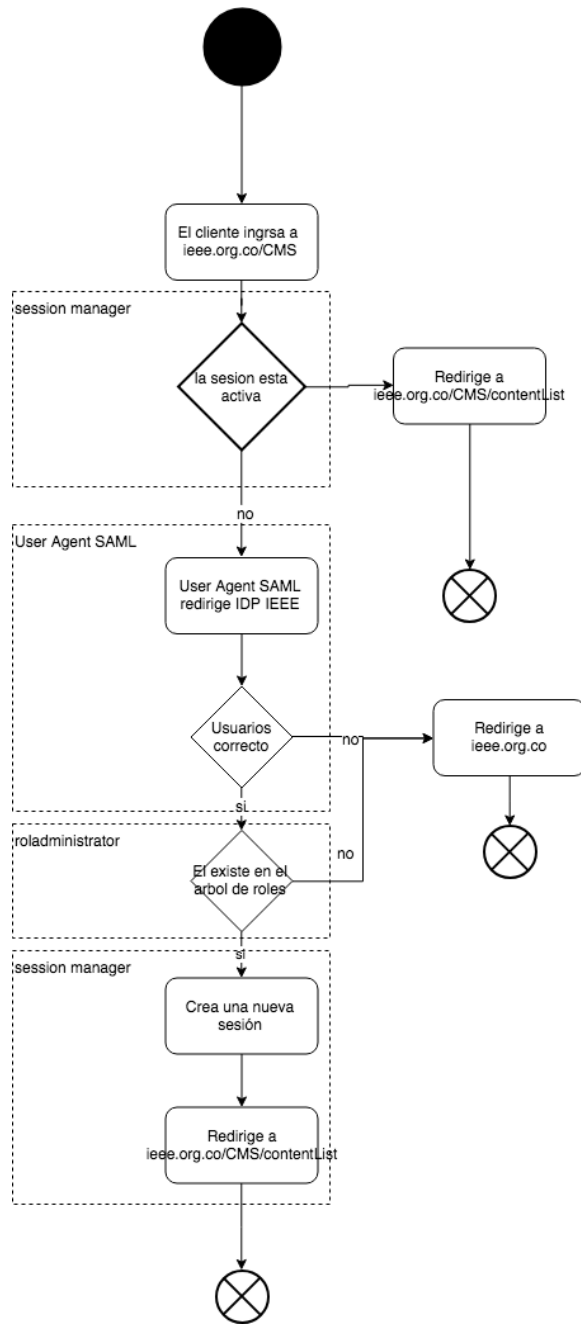


Figura 4.7: Diagrama de flujo del ingreso de una persona al gestor de contenido

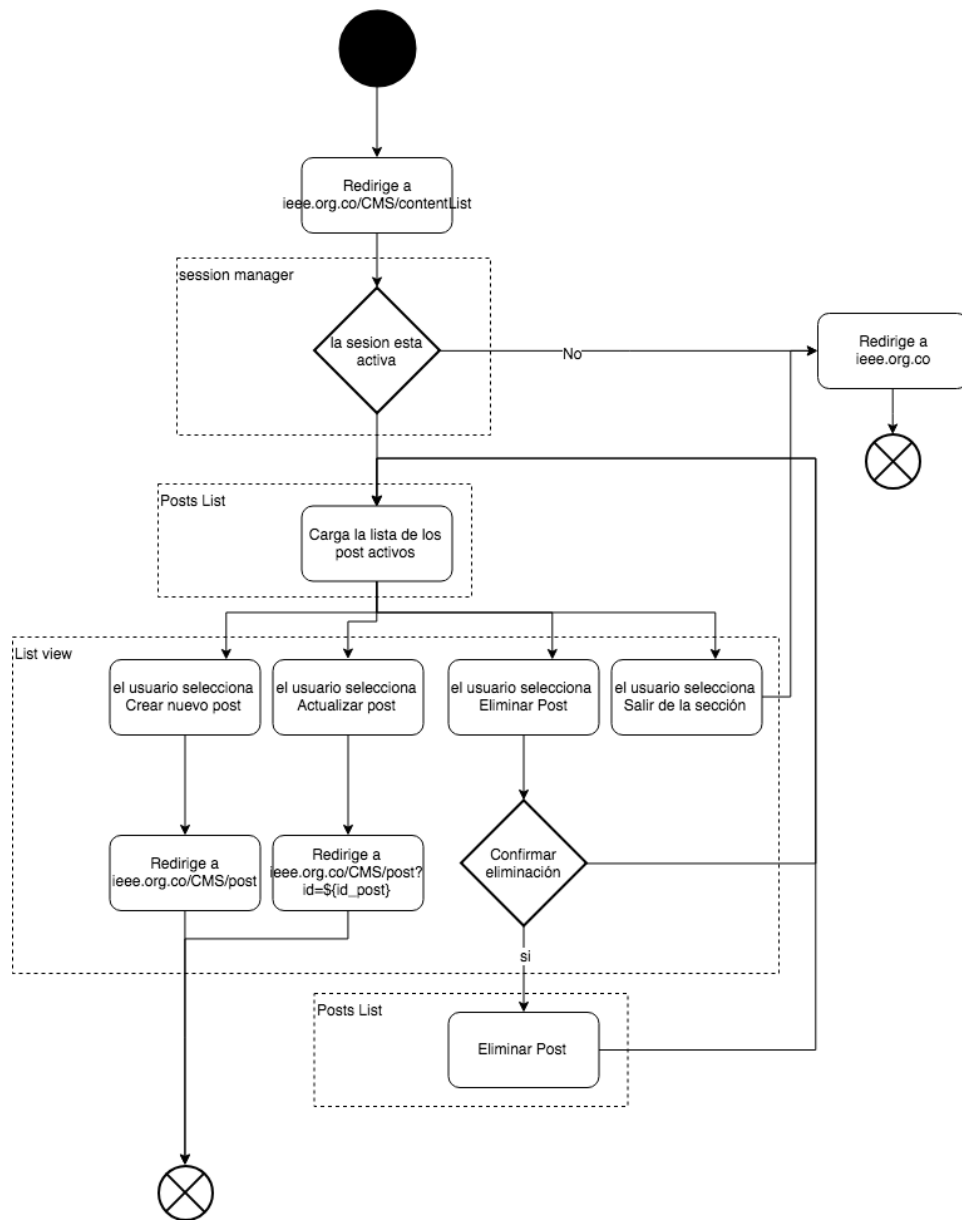


Figura 4.8: Diagrama de flujo del ingreso a la sección de lista de publicaciones de contenido

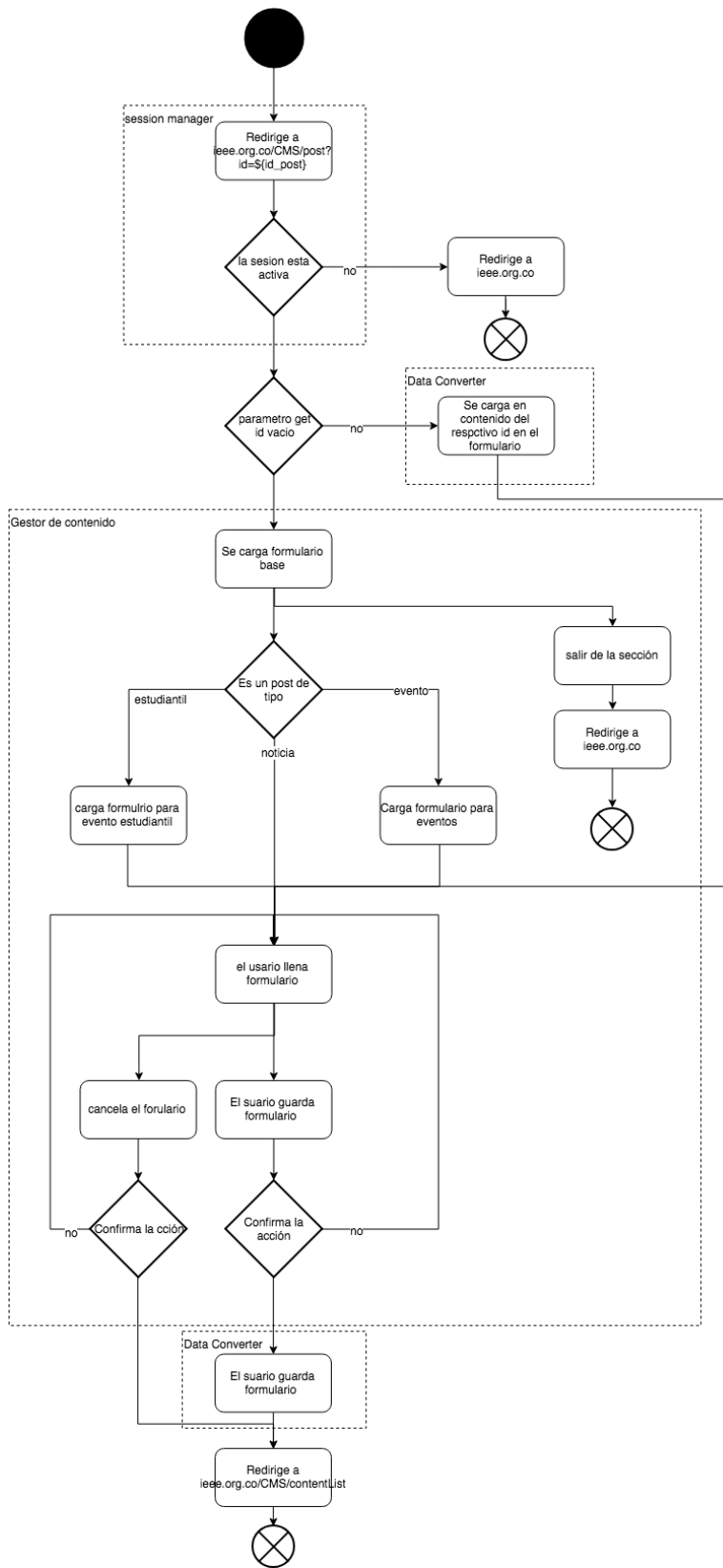


Figura 4.9: Diagrama de flujo de la sección edición de las publicaciones

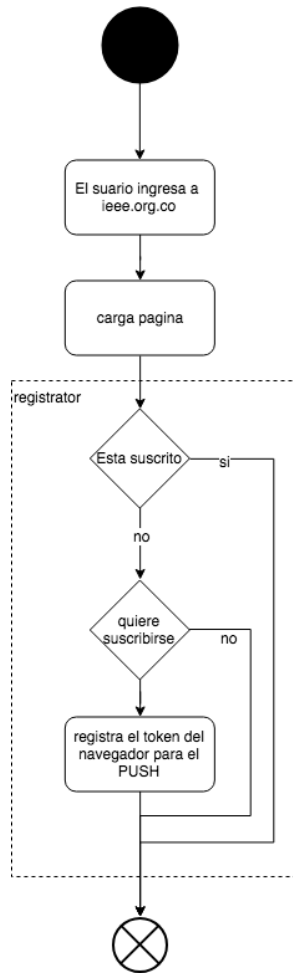


Figura 4.10: Flujo del registro de un token para notificaciones PUSH

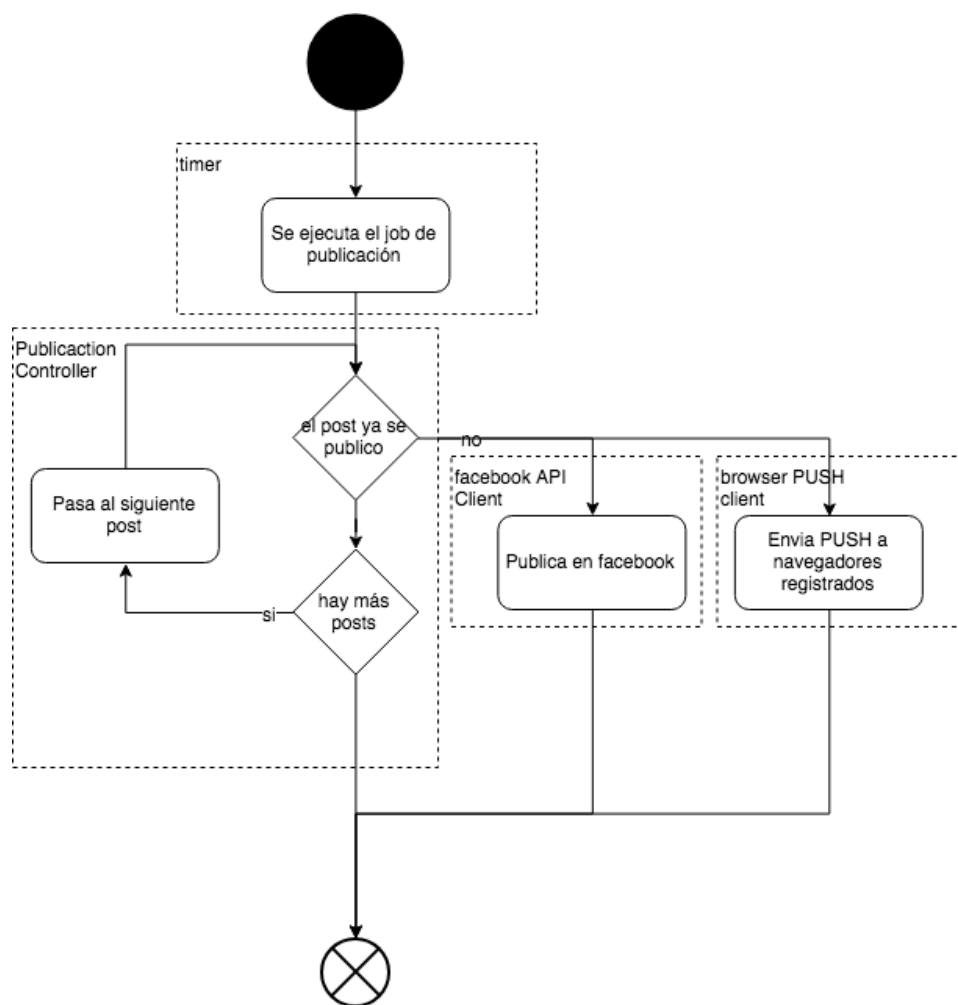


Figura 4.11: Diagrama de flujo para la tarea programada de generación de notificaciones.

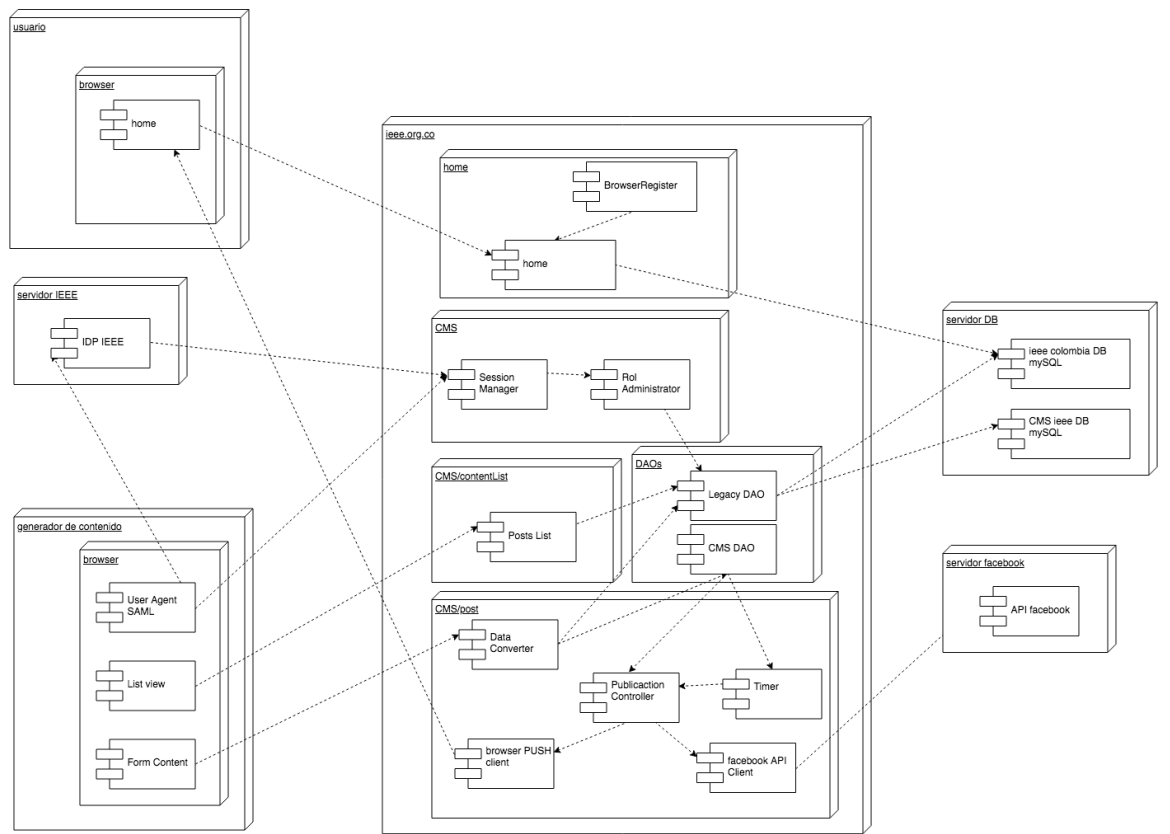


Figura 4.12: Despliegue del gestor de contenidos

Parte III

CIERRE DE LA INVESTIGACIÓN

Conclusiones

El uso de herramientas como Archimate para el diagnóstico de la organización permite tener más claro como el gestor de contenido puede afectar el desarrollo de la organización y de esta manera ayudar al fortalecimiento de la organización.

A partir de los diagramas de secuencia resulta evidente como debe ser e comportamiento del gestor de contenido con esto se espera que facilite el posterior desarrollo del proyecto.

Aun teniendo una base de requerimientos importantes el proceso de diseño en esta ocasión tiene un gran componente de creatividad y de entender como los diferentes patrones y estrategias de arquitectura pueden ayudar a desarrollar un producto que su principal característica debe ser la mantenibilidad.

La propuesta plantea diferentes retos a los actuales gestores de contenido ya que propone una independencia total de la estructura de la estructura de la página.

Trabajos Futuros

Con el desarrollo de este modelo el siguiente paso es la generación del gestor para esto se debe tomar una investigación más profunda de la estructura de los datos en la base de datos de la organización y de esta manera poder definir de manera más clara el transformador. Adicionalmente el diagnostico inicial de la actual plataforma plantea la necesidad de reestructurarla para de esta manera resolver algunos problemas encontrados en esta como son la falta de patrones que la hacen difícil de mantener, el uso de librerías obsoletas que la hacen inseguros y un diseño des actualizado a los estándares actuales.

Bibliografía

- [1] *std. 1471*. IEEE, 2000.
- [2] *ArchiMate® 2.0 Specification*. The Open Group, 2012.
- [3] Miguel Angel Alvarez. Desarrolloweb. <https://desarrolloweb.com/articulos/que-es-mvc.html>.
- [4] IBM Knowledge Center. Arquitectura orientada a servicios. <https://www.ibm.com/support/knowledgecenter/es/SSV2LR/com.ibm.wbpm.wid.main.doc/prodoverv>
- [5] Miguel Angel Ramos Barros Javier Calvarro Nelson César de la Torre Llorente, Unai Zorrilla Castro. *Guía de Arquitectura N-Capas orientada al Dominio con .NET*. Microsoft Ibérica S.R.L., 2010.
- [6] Mohamed Ariff Ameen Affiliation Chan Chung Hoong. Intuitive content management system. *International Conference on Computer, Communication, and Control Technology*, pages 541–543, 2015.
- [7] Melvin E. Conway. How do committees invent? *Datamation*, pages 28–31, 1968.
- [8] Facebook. Publicación /post-id. <https://developers.facebook.com/docs/graph-api/reference/v3.2/post>.
- [9] Jose Luis Gomez. Arquitectura de microservicios. <https://blog.sarenet.es/arquitectura-microservicios/>.
- [10] M. Ide, T. Kishida, M. Aoyama, and Y. Kikushima. A goal-oriented design methodology of it-driven business architecture. In *2014 21st Asia-Pacific Software Engineering Conference*, volume 2, pages 11–14, Dec 2014.
- [11] IEEE. Ieee regional world map. <https://www.ieee.org/communities/regional-world-map.html>.

- [12] IEEE. [ieeee.org. https://www.ieee.org/about/index.html](https://www.ieee.org/about/index.html).
- [13] H. Jonkers, E. Proper, M. M. Lankhorst, D. A. C. Quartel, and M. Iacob. Archimate(r) for integrated modelling throughout the architecture development and implementation cycle. In *2011 IEEE 13th Conference on Commerce and Enterprise Computing*, pages 294–301, Sept 2011.
- [14] Philippe B. Kruchten. The 4+1 view model of architecture. *IEEE Software*, pages 42–50, 1995.
- [15] Robert C. Martin. *Clean Architecture*. Prentice Hall, 2017.
- [16] Juan Tahuiton Mora. *Arquitectura de software para aplicaciones Web*. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, 2011.
- [17] Mozilla. Push api. <https://developer.mozilla.org/es/docs/Web/API/Push-API>.
- [18] Raheela Nasim. Applications, software architectures: A comparative study for web based. *IJCSI International Journal of Computer Science Issues*, pages 125 – 127, 2014.
- [19] Carlos Billy Reynoso. *Introducción a la Arquitectura de Software*. UNIVERSIDAD DE BUENOS AIRES, 2004.
- [20] Shailesh Kumar Shivakumar. *Enterprise Content and Search Management for Building Digital Platform*. the IEEE Computer Society, 2017.
- [21] Didier T. ¿qué considerar antes de involucrarse con microservicios? <http://www.psl.com.co/blog/que-considerar-antes-de-involucrarse-con-microservicios.html>.
- [22] Zainudin Zuhri Teduh Dirgahayu, Novi Setiani. Information requirements engineering for specific content management systems. *2014 IEEE Conference on Open Systems (ICOS)*, pages 54–59, 2014.