

Desarrollo de Mashup de Noticias: El caso de la Universidad Nacional de la Patagonia Austral

Development of news mashup: the case in the National University of Austral Patagonia

Emiliano Agustín Agüero, Vanina Luciana Gola, Graciela Vidal y Sandra Casas
ProyectoMashupWeb@gmail.com, gvidal@uarg.unpa.edu.ar, sicasas@uarg.unpa.edu.ar
GISP, Instituto de Tecnología Aplicada

Unidad Académica Río Gallego, Universidad Nacional de la Patagonia Austral
Avenida Gregores y Piloto Lero Rivera s/n - Río Gallegos - Santa Cruz - Argentina

Recibido: 09/12/2020. Aceptado: 25/06/2021

RESUMEN

El desarrollo de aplicaciones web Mashup ha adquirido relevancia en los últimos siete años. Entre las herramientas y tecnologías disponibles para estas aplicaciones híbridas se destacan los componentes feed (ATOM y RSS), que permiten obtener datos de diferentes fuentes y formatos (XML y JSON). El objetivo del presente trabajo fue estudiar y aplicar técnicas y estrategias para la extracción, composición e integración de contenidos provenientes de diversos feed relacionados a periódicos digitales. Para lograr este objetivo procedimos a: (i) estudiar y seleccionar herramientas y componentes para el desarrollo de mashup de noticias; (ii) definir, analizar y seleccionar esquemas de diseño de visualización/interfaz de mashup; y (iii) prototipar y evaluar la implementación de mashup de noticias. A partir de los resultados se desarrolló un portal de noticias en el cual se pueden visualizar las novedades de la Universidad Nacional de la Patagonia Austral, publicados por periódicos digitales locales. Además, se elaboró una guía de recomendaciones para la creación de este tipo de mashup.

Palabras clave: Web Mashup; Feeds; RSS; ATOM; web 2.0.

ABSTRACT

The development of web Mashup apps has acquired relevance especially during the last 7 years. Among the tools and technologies available for these hybrid apps, the feed components (ATOM and RSS) stand out. These allow us to obtain data from different sources and formats (XML and JSON). The main purpose of this work was to study and apply techniques and strategies for the extraction, composition and content integration from diverse feeds related to digital newspapers. In order to do this we proceeded to, (i) study and select tools and components to develop news' mashup; (ii) define, analyse and select mashup interface and display layout diagrams; and (iii) prototype and evaluate the news' mashup implementation. From the results obtained a news portal was created in which we can visualize National University of Austral Patagonia updates, published in local digital newspapers. Furthermore, a guide with recommendations for the creation of this type of mashup was developed.

Keywords: Web Mashup; Feed; RSS; ATOM; web 2.0.



INTRODUCCIÓN

El término Mashup en el idioma inglés, significa mezclar, y sus orígenes se remontan a la industria musical, cuando se buscaba elaborar una nueva canción basada en la combinación de fragmentos de varias canciones (Hernández Murguía, 2007) Sin embargo, en el ámbito del desarrollo web, el término se refiere a las aplicaciones web combinadas o híbridas; las cuales extraen información o datos de diversas fuentes, obteniendo como resultado, un nuevo sitio o aplicación web (Marroquín Borja, 2016).

Los *Mashups* se han convertido en uno de los pilares de la Web 2.0 debido a su sencillez y a su fácil acceso para desarrolladores y usuarios, y actualmente, son considerados como uno de los avances más importantes en el desarrollo de aplicaciones web; puesto que facilitan la combinación de una multiplicidad de datos que existen en Internet de manera sencilla pero visualmente muy atractiva y robusta (Cordón, 2007).

El desarrollo Mashup ha sido abarcado desde diferentes enfoques por distintos autores. Algunos consideran este tipo de desarrollo como una actividad sencilla, ya que integrar una nueva aplicación a la nube no es una tarea compleja, dado que existen librerías y herramientas tecnológicas que permiten hacerlo (Fichter, 2009).

Sin embargo, desde otro punto de vista, el desarrollo de Mashup ha revelado que existen problemas con respecto a la tarea de combinación, debido a que se requiere analizar los requerimientos de integridad de datos, la capacidad de acceso a los recursos y las herramientas que faciliten un acceso eficiente, es decir, que poseen la capacidad de desacoplar los datos para poder utilizar los mismos para hacer el “Mashing” (Koschmider et al., 2009).

Este trabajo propone estudiar y aplicar técnicas y estrategias para la extracción, composición e integración de contenidos proveniente de periódicos digitales. Para ello se desarrollará un portal de noticias para la UNPA, que integre la información que se publique en relación a la institución, de distintos periódicos digitales de la región. Esta integración y composición implica varios procesos, la conexión a estos sitios en la web, pero además diversos filtrados que extraigan solo las noticias de la universidad y ordenen según diversos criterios. De esta forma, se reusará el contenido y serán desplegados en un portal que los integre de manera organizada y que automáticamente los mantenga actualizados.

Por último, se procede a indicar la estructura del documento. La Sección 2 abarca el concepto de Web Mashup, sus tipos, sus elementos y componentes, mientras que en la Sección 3, se describe la metodología usada para la implementación del portal, abarcando desde la aplicación de la guía de desarrollo utilizada, herramientas de diseño e implementación de la aplicación. Luego se procede con los resultados y las conclusiones finales.

1. WEB MASHUP

La web social o web 2.0 se resume en el paso de las tradicionales webs a otras destinadas a los usuarios. De la concepción estática de los sitios web tradicionales se evoluciona a otros marcados por un cambio constante que, en muchas ocasiones, procede tanto de los gestores del mismo como de los mismos usuarios, que adquieren los roles de editores e incluso generadores de contenidos (Tejedor et al., 2011). La web 2.0 se concibe, como una plataforma cuyas principales diferencias con la 1.0 se sintetizan en los siguientes puntos:

Tabla 1: Diferencias Web 1.0 y Web 2.0, O'Reilly, 2005

Web 1.0	Web 2.0
DoubleClick Ofoto Akamai mp3.com Britannica Online personal websites evite domain name speculation page views screen scraping publishing content management systems directories (taxonomy) stickiness	Google AdSense Flickr BitTorrent Napster Wikipedia blogging upcoming.org and EVDB search engine optimization cost per click web services participation wikis tagging («folksonomy») syndication

En el escenario que inaugura la web social, también llamada web semántica, aparecen los “Mashups”. Un Mashup es una terminología que significa combinar. Es una forma de integración y reutilización que ocurre cuando una aplicación web es usada o accedida desde otra aplicación, con el fin de reutilizar su contenido y/o funcionalidad (Nieto Peñalver, 2012). La nueva aplicación permite crear nuevos servicios simples, visualizados en una única interfaz gráfica. También son conocidos como aplicaciones web híbridas, ya que permiten combinar de forma innovadora, dos o más fuentes de información que permiten crear una nueva aplicación web.

El contenido utilizado en un Mashup es típicamente usado por terceros a través de una interfaz pública o usando un API. Otros métodos que constituyen el origen de sus datos incluyen: sindicadores web (RSS o Atom) (Casado Manzanero, 2009). Normalmente, un *Mashup* mejora la presentación visual de la información, ofrece valor agregado a sus usuarios combinando la información de diferentes fuentes o ambas cosas (Nieto Peñalver, 2012).

Los Mashup, han alcanzado un notable éxito en ámbitos como el inmobiliario, publicitario, televenta, hotelería, etc. Sin embargo, en el ámbito ciberperiodístico la aplicación de estas singulares “combinaciones” de contenidos y de programas, está menos desarrollada. En este sentido, es posible hacer alusión a un nuevo tipo de convergencia mediática más allá de la concepción habitual del multimedia. El “Periodismo Mashup” introduce severas transformaciones más afines a las directrices de la web social y más centradas en las necesidades concretas de los internautas (Tejedor, 2007).

1.1 Clasificaciones de un Mashup

Los Mashups, se pueden clasificar en dos tipos, basados desde perspectivas diferentes.

Orientado hacia el navegador (browser): Este tipo, está más enfocado en la mezcla o composición de información con imágenes en el navegador, principalmente usando JavaScript como lenguaje de programación para lograrlo.

Orientado hacia el servidor (Mashup empresarial): En este otro tipo, la integración y manipulación de la información suceden en ambos lados: servidor y navegador. Su uso principal es interactuar con información de diferentes sistemas generando vistas necesarias para la toma de decisiones (Hernández Murguía, 2007).

Por otra parte, se señala que el cambio para los sitios de los diversos medios de comunicación en internet "radica en un desplazamiento de aplicaciones puramente virtuales concebidas con finalidades comerciales, publicitarias al terreno informativo". Esto podría explicar porque, actualmente, las páginas web de los diferentes sitios informativos añaden contenidos y aplicaciones, ya sean escritas, audiovisuales, nuevos diseños, etc., de terceros para potenciar estos mismos y así provocar que el contenido entregado al usuario sea de la mejor forma posible (Tejedor, 2007).

1.2 Componentes de un Mashup

Un componente de Mashup es cualquier segmento de datos, lógica de aplicación y/o interfaz de usuario que puede ser reutilizada y que es accesible ya sea local o de forma remota. En este contexto existen otras piezas de software que pueden ser consideradas también como componentes, tales como protocolos RSS, Atom, Screen scraping o servicios web (Tinajero Díaz, 2016).

RSS: Son las siglas de *Really Simple Syndication*. Conocido también como "tomar notas de forma simple". Una RSS es una tecnología que está basada en una estructura XML que permite compartir contenido de una web determinada, como así también distribuir contenidos de una web, sin necesidad de un navegador. De esta forma, esos contenidos pueden ser mostrados en una aplicación o en otra página web. Los formatos de extensión para RSS son .rss o .xml y representan objetos de tipo MIME, application/rss+xml. Es importante destacar que la primera versión de RSS, también es conocida como RDF, por tanto, algunos canales RSS se etiquetan como TSS 1.0 o RDF. La extensión en este caso es .rdf.

Para poder recibir noticias RSS, se precisa que el sitio o blog en internet, ofrezca el servicio RSS. Esto es algo totalmente normal en los blogs, debido a que lo tienen habilitado por defecto.

El RSS se identifica fácilmente por un logotipo habitualmente naranja con tres arcos y/o el texto RSS, que, para poder leerlo, se debe disponer de un programa lector RSS o RSS reader, el cual puede tratarse desde un programa instalado en la computadora, el propio navegador de internet, o simplemente un programa cliente de correo electrónico o un lector online como por ejemplo Feedly o Inoreader.

Una vez instalado el programa RSS o mediante la aplicación online en la web, se deberán dar de alta aquellas páginas de las que se desea recibir actualizaciones. Al darse de alta, se reciben directamente las noticias y los nuevos artículos de aquellas páginas web de interés y que se hayan incluido en los Feeds (canales o fuentes) deseados, con la ventaja de tener las páginas web reunidas en un único lugar.



El alimentador RSS facilita la gestión y publicación de información y noticia en sitios y blogs. Es una forma estandarizada de distribución de la información de las páginas web a los lectores de las páginas. Esta información se distribuye a través de las fuentes RSS o Canales RSS. Gracias a esto, los lectores pasan a tener una herramienta útil para mantenerse informado sobre las noticias y webs que le resultan de interés, conservando y almacenando toda la información en un solo lugar que se actualiza de manera automática.

Además, el RSS ofrece al lector un considerable ahorro en tiempo en la lectura de información y noticias, ya que sólo con abrir el lector RSS (ya sea un programa, el navegador o un lector web online), el usuario podrá ver cuáles son las últimas actualizaciones y noticias que han publicado las diferentes páginas web a las que está suscrito (Menéndez, 2018).

Atom: Esta tecnología está basada en el lenguaje XML, que permite estructurar y distribuir contenidos web, tal y como ocurre en RSS. No obstante, Atom no se basa ni se corresponde con ninguna versión de RSS. El formato del documento XML es similar, y tiene un mismo objetivo, ya que se trata de poder distribuir información de una forma web a través del documento XML. Las mejoras que incorpora con respecto a la tecnología RSS, es que un documento Atom tiene capacidad para contener más información, siendo esta última más compleja y consistente. Debido a esta mejora en la capacidad, aumenta el grado de dificultad de esta tecnología. El tipo de extensión que maneja es .atom o .xml y construye un objeto tipo MIME, application/atom+xml. Existen dos estándares que son RFC 4287 y RFC 5023. Los símbolos que utilizan las páginas para indicar que disponen de canales Atom para permitir la distribución de contenidos a un lector de Atom son similares a los de RSS, aunque indican que se trata de Atom y no de RSS (Vara Mesa, 2000).

Los elementos de información que forman un feed Atom se conocen como entradas Atom. Un proveedor de contenido publica un feed de Atom al ponerlo a disposición a través de una URL en Internet, actualizándolo con nuevos elementos. Las páginas web pueden mostrar los elementos en la fuente Atom, y los usuarios de la web pueden obtener los elementos de la fuente utilizando un lector de fuentes o un navegador web. Un Feed Atom podría usarse como parte de un Mashup, que es una aplicación web que combina el contenido de varias fuentes de datos para que los usuarios puedan experimentar y comprender los datos de una nueva manera. En un Mashup, los datos del Feed Atom pueden ser manejados por un widget, que es una aplicación de script que se ejecuta en una página web.

Screen scraping: Se traduce al español como: “raspado de pantalla”, esta es una técnica de programación, que toma la presentación de información que se visualiza por pantalla. Normalmente se extrae texto, aunque también se puede seleccionar información gráfica. El programa Screen scraping, accede a través de ingeniería inversa a la página del proveedor, y procesa el contenido HTML, recopilando la información que se desea extraer. Para utilizar esta tecnología, no es necesario que el proveedor del contenido realice alguna tarea adicional, tal y como ocurriría con RSS o Atom. Esta técnica está en desuso, debido a los inconvenientes que presenta: Las webs no la autorizan, tiene mucha complejidad, posee mucha sobrecarga de los servidores, y también propensión a fallos inesperados. En algunos casos puede ser la única opción de recuperar contenidos, aunque la tendencia a la hora de crear nuevos *Mashup* es utilizar API (Menéndez, 2018).

API: Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. Una API significa interfaz de programación de aplicaciones.

Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones, y proporcionan oportunidades de innovación, lo cual es ideal al momento de diseñar herramientas y productos nuevos (o de gestionar los actuales).

A veces, las API se consideran como contratos, con documentación que representa un acuerdo entre las partes: si una de las partes envía una solicitud remota con cierta estructura en particular, esa misma estructura determinará cómo responderá el software de la otra parte.

Debido a que simplifican la forma en que los desarrolladores integran los elementos de las aplicaciones nuevas en una arquitectura actual, las API permiten la colaboración entre el equipo comercial y el de TI. Las necesidades comerciales suelen cambiar rápidamente en respuesta a los mercados digitales en constante cambio, donde la competencia puede modificar un sector entero con una aplicación nueva. Para seguir siendo competitivos, es importante admitir la implementación y el desarrollo rápidos de servicios innovadores. El desarrollo de aplicaciones nativas de la nube es una forma identificable de aumentar la velocidad de desarrollo y se basa en la conexión de una arquitectura de aplicaciones de microservicios a través de las API.

Las API son un medio simplificado para conectar su propia infraestructura a través del desarrollo de aplicaciones nativas de la nube, pero también le permiten compartir sus datos con clientes y otros usuarios externos. Las API públicas representan un valor comercial único porque simplifican y amplían la forma en que se conecta con sus partners y, además, pueden rentabilizar sus datos (un ejemplo conocido es la API de Google Maps) (Comunidad Red Hat, 2007).



Figura 1 - Arquitectura Cliente-Servidor

En la Figura 1 se ilustra como el proveedor de contenidos facilita una API al servicio de páginas híbridas. El servidor une los contenidos propios de la página con los datos recibidos del proveedor, y genera una página híbrida que pone a disposición del cliente.

1.3 Categorías de un Mashup

Los *Mashups* forman parte de un ecosistema muy grande que está creciendo a un ritmo acelerado y prueba de esto es que existe un sitio web llamado “programmableweb.com”. Este sitio es un directorio web de API’s y de *Mashups* que pueden ser inspeccionados. Además, se puede conocer las últimas noticias referentes a los nuevos *Mashups* que están surgiendo. El sitio programmableweb.com tiene clasificadas las APIs y *Mashups* en categorías de tal forma que sea más fácil para los usuarios poder buscar una determinada API o *Mashup* dependiendo del dominio de la aplicación.

En la Tabla 1 se detalla la cantidad de mashups más populares que existen por categoría, consultado en Junio del año 2020 en el sitio de programmableweb.com.

Tabla 2- Lista de APIs más populares – Junio de 2020, programmableweb

MASHUP POR CATEGORÍA: JUNIO 2020		
N°	Categoría	Cantidad
1	Cartografía	2.881
2	Búsqueda	1.156
3	Social	1.117
4	eCommerce	840
5	Fotos	831
6	Videos	723
7	Viajes	596
8	Música	576
9	Mensajería	537
10	Móvil	499
11	Referencia	425
12	Blogs	403
13	Servicios de Noticias	401
14	Visualizaciones	349
15	Deportes	336
16	Humor	334
17	Telefonía	325
18	Bienes raíces	270
19	Herramientas	245
20	Widges	243
21	Eventos	211
22	Juegos	204
23	Transporte	190
24	Inglaterra	180
25	Gobierno	167

Los nombres de categorías son un buen instrumento para los desarrolladores a la hora de buscar Mashups en un dominio dado, y se puede deducir que existen tantas APIS's que pueden ser usadas en aplicaciones web convencionales, así como en entornos empresariales; a esto nos referimos como Entorno. Además, existen diferentes dominios en los cuales pueden aplicarse los Mashups, por ejemplo, aplicaciones móviles, telecomunicaciones, procesos, y algunas más genéricas; a esto nos referimos con Dominio. En la composición de un Mashup, pueden existir

también componentes de datos, lógica y de interfaz de usuario; a esto se refiere como Composición (Tinajero Díaz, 2016).

2. MATERIALES Y MÉTODOS

El desarrollo del Mashup de noticias UNPA, se realizó mediante un proceso iterativo, en el cual se aplicaron de forma simultánea tres estrategias: La guía para el desarrollo Mashup (sección 2.1), el diseño (sección 2.2) y la implementación (sección 2.3). Estas acciones se fueron retroalimentando antes de pasar a la implementación. Una vez terminada una iteración, se retomaba como punto de partida a la guía y el diseño, realizando el análisis necesario e implementando las mismas, como puede observarse en la Figura 2.

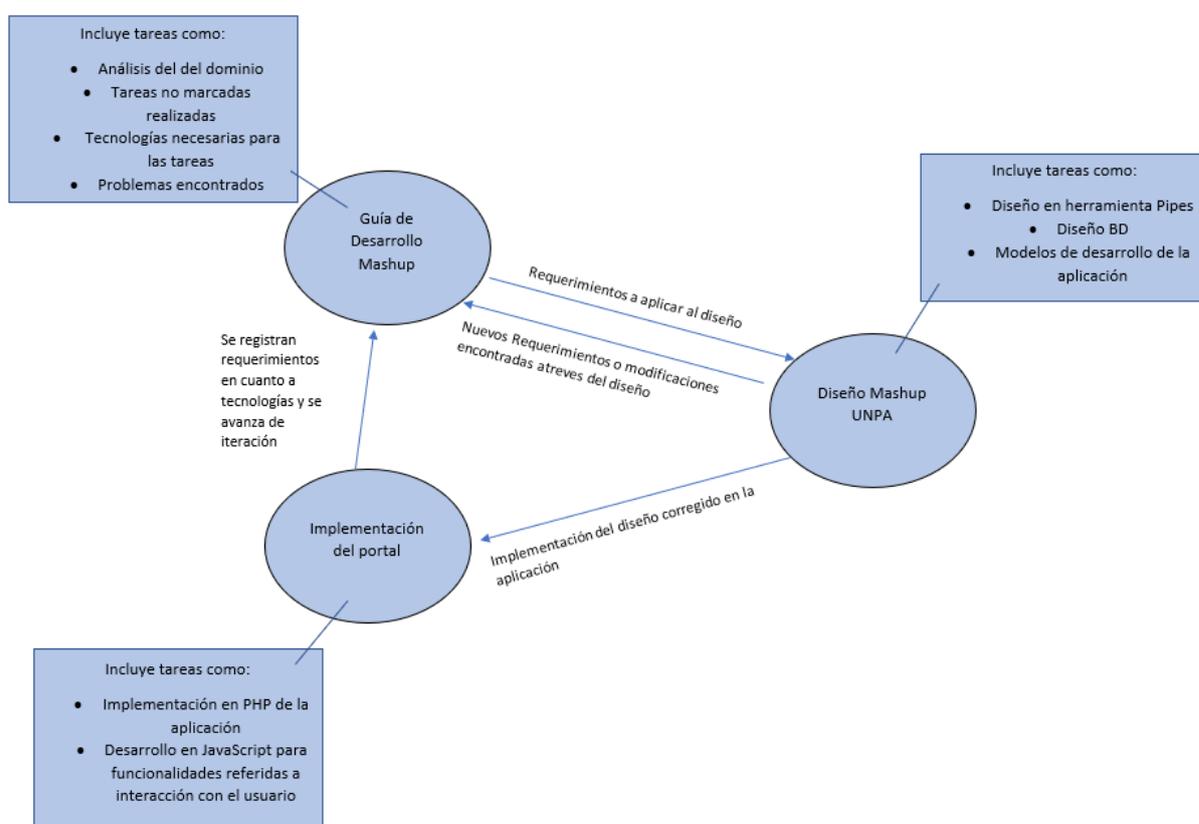


Figura 2 – Pasos y estrategias aplicadas

A continuación, se mencionan las técnicas y herramientas aplicadas en este proyecto.

2.1 Guías para el desarrollo de Mashup

Para el trabajo de campo, se utilizó la guía de desarrollo “Elucidating the Mashup Hype: Definition, Challenges, Methodical Guide and Tools for Mashups” (Koschmider et al., 2009), que comprende seis actividades que el desarrollador debe seguir en la construcción del Mashup. Aun así, fue necesario adaptarla, incorporando las actividades, obstáculos y acciones realizadas y necesarias. La misma fue aplicada en iteraciones de 15 días cada una, y está compuesta por siete etapas, las cuales son:

Indicar el dominio del problema y definir los requerimientos: En esta etapa, se describen los requerimientos que se han de implementar, haciendo un análisis del dominio del problema.

Identificar el entorno de TI: Selección de recursos tecnológicos utilizados para la implementación de la aplicación.

Identificar los requisitos técnicos: Recursos necesarios para crear la solución a los requerimientos.

Identificar el conjunto de tecnología: Recursos necesarios que se deben tener instalados previamente, que sirven para dar solución a los requerimientos.

Mantenimiento del mashup: En esta etapa se define cómo se realiza el control de versiones del aplicativo.

Identificar obstáculos: Se registran los problemas que fueron hallados en la iteración específica.

Acciones adicionales y necesarias: Instrumentos o requerimientos que no estaban definidos, pero que fueron necesarios para llevar adelante la iteración.

Las acciones de la guía fueron registradas y documentadas en una planilla.

2.2 Modelado del Mashup de Noticias: PIPES

Para el modelado del diseño del mashup de noticias se utilizó la herramienta PIPES (<https://pipes.digital>). Pipes es considerado un editor de programación visual especializado en Feeds, un shell visual, o simplemente como un configurador de Feeds.

El modelado se basa en el concepto de “tubería” o “PIPES”. Este término abstracto se refiere a la conexión existente entre los bloques, los cuales son operaciones o funciones que se aplican a los datos de entrada generando salidas. Los bloques tienen distintos propósitos, como buscar, crear Feeds, permitiendo su manipulación de diversas maneras, ya sea filtrando, extrayendo, fusionando u ordenando los datos. El resultado del diseño final es denominado tubería.

Los formatos de entrada Pipes son compatibles con los canales RSS, Atom y JSON, y puede raspar documentos HTML y fusionarlos con archivos de texto normales. El formato de salida predeterminado de una tubería es RSS.

Las operaciones utilizadas en el proyecto fueron las siguientes tres:

- **Feed:** Esta operación, suele ser el punto de partida de una tubería, en donde toma como entrada de usuario, la url que coloca para descargar. Un Feed puede ser del tipo RSS, Atom o JSON. El Pipes almacena en caché todas las descargas durante algunos minutos.
- **Filtrar:** Filtra del Feed de entrada a través de una palabra clave, extrayendo solo los elementos que la contienen. Busca la palabra en el título de los elementos, la descripción (si existe) y el contenido, distinguiendo entre mayúsculas y minúsculas.

- **Combinar:** Esta operación, une dos o más fuentes de entrada en una sola fuente de salida, en el que se obtiene como resultado una asociación de las etiquetas correspondientes a cada elemento y además añade el título y las descripciones de los feeds utilizados en esta operación. El orden de clasificación sigue el orden de entrada: la fuente de salida tiene primero todos los elementos de la primera fuente de entrada, luego todos los elementos de la segunda fuente de entrada, y así sucesivamente.

2.3 – Herramientas para la implementación del Portal

Las herramientas que se utilizaron en la implementación del Portal, fueron las siguientes:

PHP: Es un lenguaje de código abierto específico para el desarrollo de aplicaciones Web y con el cual se desarrolló este proyecto. La versión de PHP utilizada fue la 7.4.3. La página web del sitio es: <https://www.php.net/>

Apache TomCat: Este es un servidor web HTTP el cual tiene como función permitir el acceso de todos los usuarios a la aplicación. La versión de Apache TomCat utilizada fue la 2.3. La página web del sitio es: <http://tomcat.apache.org/>

My SQL Server: Esta aplicación cumple con la función de alojar la base de datos para poder acceder al histórico de las noticias sobre el campus. La versión de My SQL Server utilizada fue la 5.5.9. La página web del sitio es: <https://www.mysql.com/>

Angular: Su función es facilitar la creación y programación de aplicaciones web en una sola página. La versión de Angular utilizada fue la 7. La página web del sitio es: <https://angular.io/>.

JavaScript: Este es un lenguaje de programación ligero e interpretado que nos permite la ejecución o compilación justo a tiempo. La versión de Java Script ECMAScript utilizada fue la 2018. La página web del sitio es: <https://ecma-international.org/ecma-262/9.0/>.

Ajax: Su funcionalidad nos permite poder, aplicar un conjunto de técnicas de desarrollo web para aplicar Java Script y XML de manera Asíncrona. La versión de Ajax utilizada fue la 3.5.1.

RSS & Atom Feeds for PHP: Librería utilizada que permite consumir RSS y ATOM Feeds a partir de php. La versión utilizada fue la 1.2. La pagina web del repositorio es <https://github.com/dg/rss-php>

3. RESULTADOS

Los resultados obtenidos a lo largo del proyecto fueron los siguientes:

3.1 Aplicación de la guía de desarrollo Mashup

En la Tabla 3, se presenta el detalle de la aplicación de las Guías de Desarrollo de Mashup, por etapas y en cada una de las seis iteraciones ejecutadas.



Etapas	Iteración 1	Iteración 2	Iteración 3	Iteración 4	Iteración 5	Iteración 6
Indique el dominio del problema y defina	Mostrar feed en pagina	Realizar modificaciones en la página, crear base de datos y almacenar noticias.	Agregar filtrados de "Contenido" y "Descripción" en la página. Utilización de editor de programación visual especializado en feeds, que permita filtrar, extraer, fusionar y ordenar. Buscar intersecciones que existen entre los distintos campos de Diarios. Filtrar por nombre de Diario y Unidad Académica.	Completar el modelo PIPES, Agregar el código de nuevos diarios	Revisar operaciones Pipes, modificar tamaño de título, optimización al método combinar, modificar PIPES	Sacar "Unidades académicas" en la página. Realizar apartado de contacto. Pasar PIPES en una sola imagen. Subir Proyecto al Servidor
Identifique el entorno de TI, especialmente	Los diarios utilizados son: Ahora Calafate y Diario Nuevo Día	Se suman nuevos diarios: Santa Cruz Uno, El Espejo Diario, Radio San Jorge	No hay cambios	Se suman nuevos diarios: Patagonia anexo, Radio Noticias, Portal Uasj, Santa Cruz al momento, Prensa obrera, El espejo diario	No hay cambios	No hay cambios
Identificar los requisitos técnicos, especialmente	Servidor Apache, Visual Studio Code, Computadora	Se agrega la utilización de base de datos sql	Se agrega utilización de herramienta pipes	No hay cambios	No hay cambios	No hay cambios
Identifique el conjunto de tecnología, especialmente :	PHP, HTML5, Archivos RSS, Archivos RSS/ATOM	Php my admin, JavaScript	Pipes.digital	No hay cambios	No hay cambios	No hay cambios
Mantenga su mashup, especialmente	Control de versiones a través de carpeta compartida en drive	No hay modificaciones en control de versiones	No hay cambios	No hay cambios	No hay cambios	No hay cambios
Obstáculos	Analizar archivos xml	Errores en la obtención de información en los feeds ya que son muy diferentes el uno del otro	No existieron Problemas	Problemas y errores con los diferentes RSS de los nuevos diarios, ya que son muy diferentes unos con otros	No hay cambios	No hay cambios
Acciones realizadas no marcadas en guía y que fueron necesarias	Agregado de imágenes a las noticias obtenidas	Filtrado a partir de un buscador	Agregado de Carrusel que muestra noticias actuales	Método unificador que combina RSS	Ninguna	Análisis de interacción futura con el usuario

Tabla 3- Etapas e iteraciones de la aplicación de las guías para el desarrollo mashup

3.2 Modelado del PIPES

La Figura 3, presenta el modelo diseñado con PIPES del Mashup de Noticias UNPA. Los primeros seis bloques del diagrama (A, B, C, D, E, F, G, H) corresponden a las operaciones acceso a los sitios webs de los siguientes diarios digitales: Ahora Calafate, Radio Noticias, Patagonia Nexo, Portal UASJ, Prensa Obrera y Santa Cruz al Momento, de los cuales se extrae el Feed, y combina su información en el bloque "1A", ya que todos pertenecen a la categoría de RSS Puro. Sin embargo, los últimos dos bloques (G, H) que corresponden a los sitios webs de Diario nuevo día Local y Diario nuevo día Regional, los cuales pertenecen a la categoría de

tipo ATOM, por lo cual sus datos deben ser extraídos y combinados en un bloque diferenciado (1B).

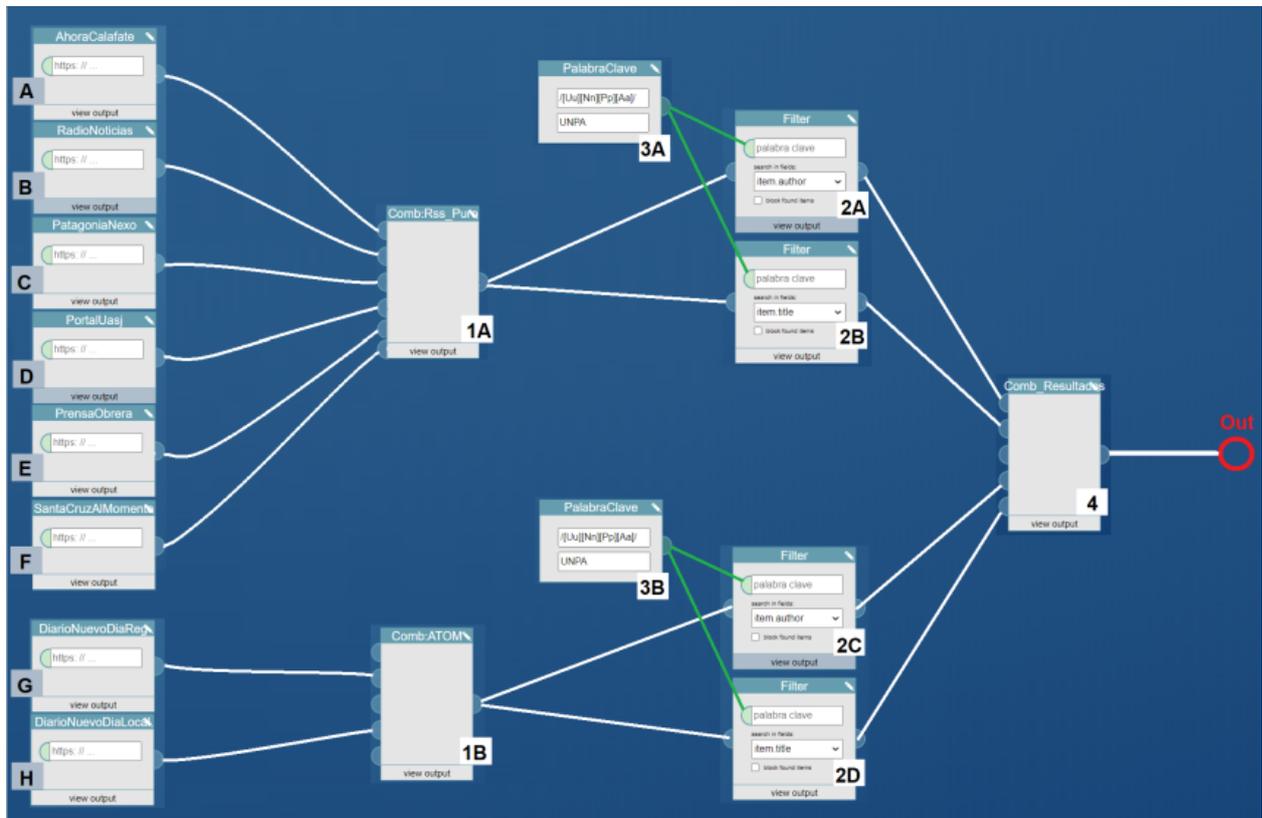


Figura 3 - Modelado del PIPES

Las salidas de los bloques combine (1A, 1B), se constituyen en las entradas de bloques que filtran ciertos elementos. Los bloques que filtran (2D, 2B), extraen los elementos que coinciden con la palabra clave: “UNPA” y “Universidad Nacional de la Patagonia Austral”, teniendo en cuenta tanto las mayúsculas como las minúsculas (3A). También el filtrado puede realizarse a partir del diario del cual proviene la noticia (2C, 2A) de la misma manera se obviarán las mayúsculas y las minúsculas (3B).

Por último, los resultados que se extraen de ambos filtros, son relacionados en una misma operación combinar (4), por lo que la salida del PIPES devuelve como resultado aquellas noticias referentes a la UNPA, ya sea de periódicos digitales que trabajan con XML, RSS o ATOM.

3.3 Implementación de Sitio Web Mashup

La Figura 4, presenta el esquema general de la implementación realizada. En el portal se distinguen dos capas de componentes y funciones: las que corresponden al Back-End y al Front-End.

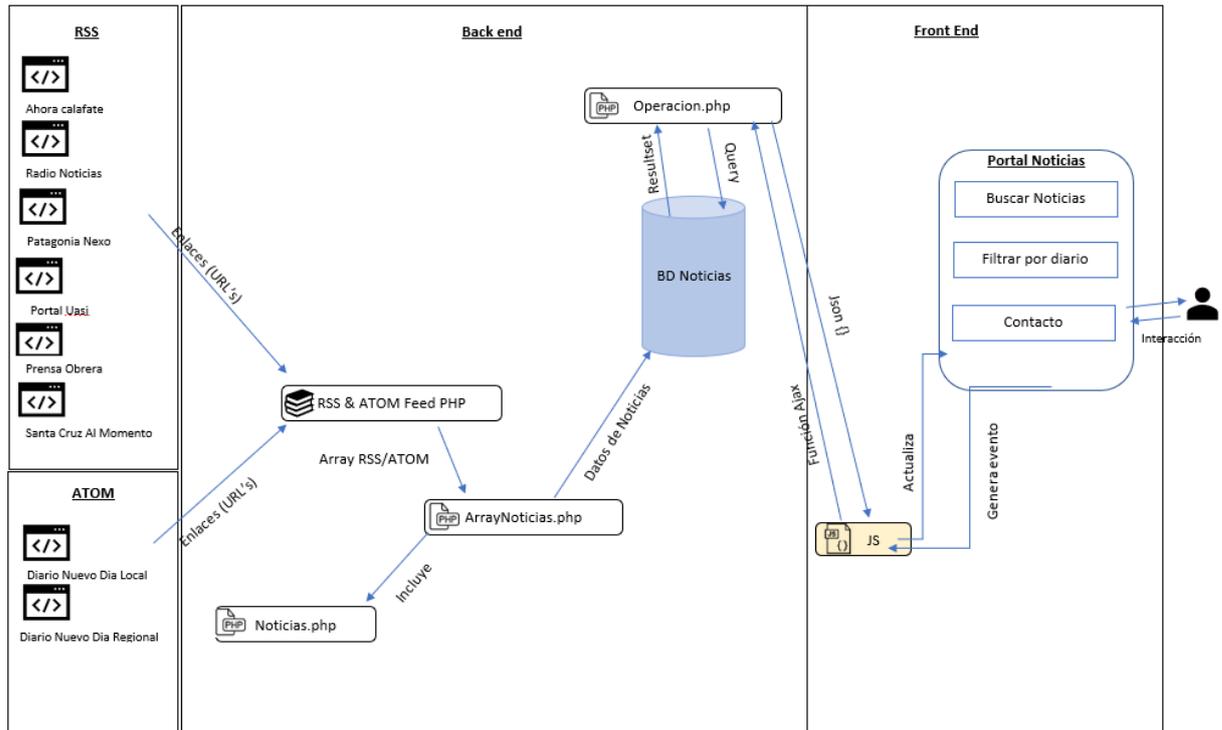


Figura 4 - Diagrama funcional de la App. separado por contextos

Uno de los contextos a analizar es el Back-End, las cuales son las actividades realizadas del lado del servidor, es decir las tareas de base de datos y servidores de aplicaciones que el usuario no puede visualizar (Caballero Valdivia, 2016). Dentro de este contexto se ubican las funcionalidades internas de la aplicación, comenzando con la obtención de los Feeds a través de la librería mencionada anteriormente (Sección 2.3), la cual transforman estos XML respectivos de cada portal en arreglos dependiendo del tipo RSS o ATOM. Estos arreglos son utilizados en ArrayNoticias.php, donde se incluyen las clases necesarias para representar las noticias con sus correspondientes atributos, la funcionalidad de esta clase es combinar estos arreglos en uno solo uniendo los datos que son de importancia para el portal de noticias y guardar las mismas en la base de datos.

Por último, se encuentra la base de datos con sus respectivas características, que pueden observarse en la Figura 5. Esta tiene como funcionalidad almacenar el histórico de las mismas, y por último los archivos PHP que contienen las operaciones SQL que son utilizados por el otro contexto y que se invocan a través de JavaScript.

Mashup
Id (Primary)/(int)
link (nvarchar(50))
fecha (date)
titulo (nvarchar(50))
autor (nvarchar(30))
descripcion (nvarchar(25))
Imagen(nvarchar(50))

Figura 5 – Tabla de Base de Datos Mashup

Dentro de la Figura 4, se encuentra el otro contexto el cual se denomina Front-End , el mismo implica el uso de las tecnologías con las que interactúa directamente el usuario (Caballero Valdivia, 2016). Dentro de este contexto se encuentra la parte grafica de la aplicación web en la cual se puede acceder a las ultimas noticias publicadas por los periódicos digitales regionales. Estas interacciones se realizan a través de JavaScript el cual establece un nexo mediante JQuery enviando una petición a los correspondientes elementos del Back-End.

A continuación, se reflejarán imágenes de la funcionalidad de la aplicación. En la Figura 6, se presenta la funcionalidad de la aplicación para iniciar sesión de usuario a partir de sus cuentas de redes sociales tales como Facebook o Google (gmail).

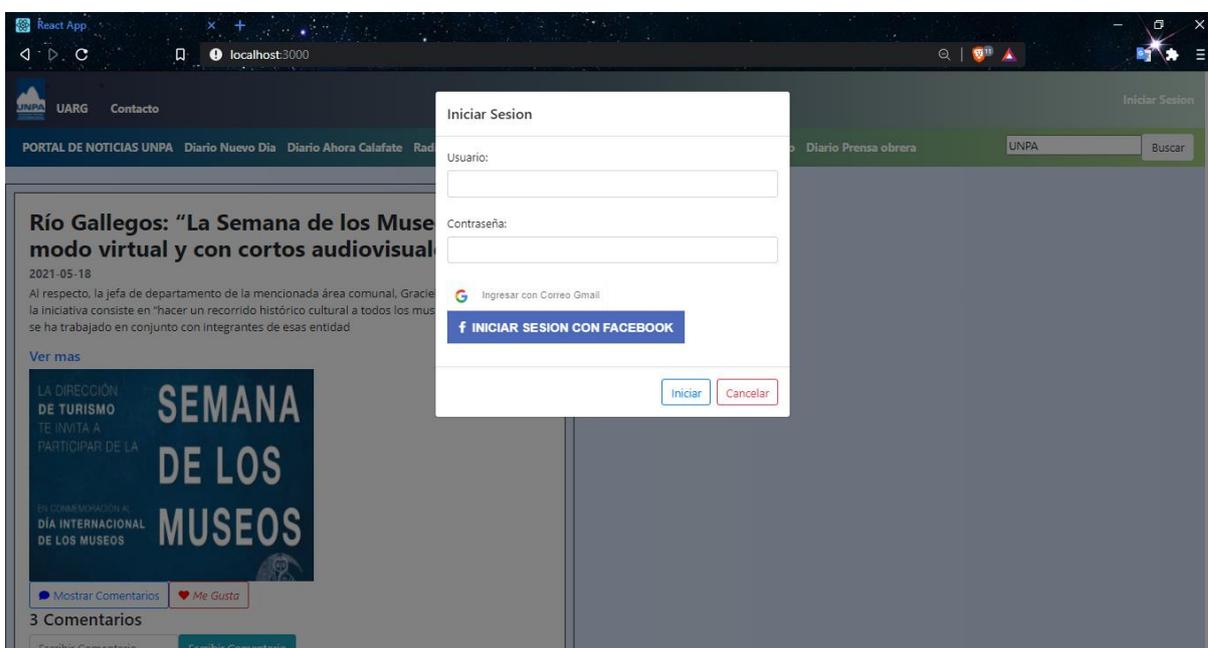


Figura 6 pantalla de inicio de sesión

En la Figura 7, se presenta la funcionalidad de la aplicación para obtener un feed-back de las noticias que aparecen en el portal a partir de los comentarios y me gusta de los usuarios registrados en el portal.

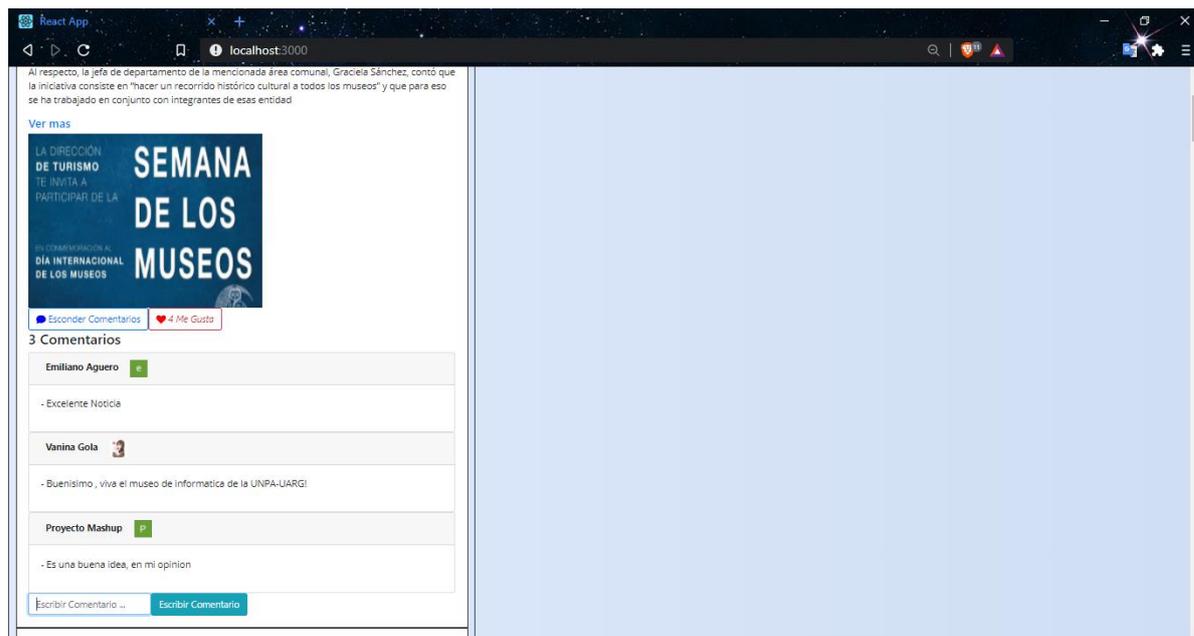


Figura 7 Sección de comentarios y me gusta una noticia

En la Figura 8, se presenta la funcionalidad de la aplicación en filtrar las noticias que se refieren específicamente a la unidad académica UNPA.

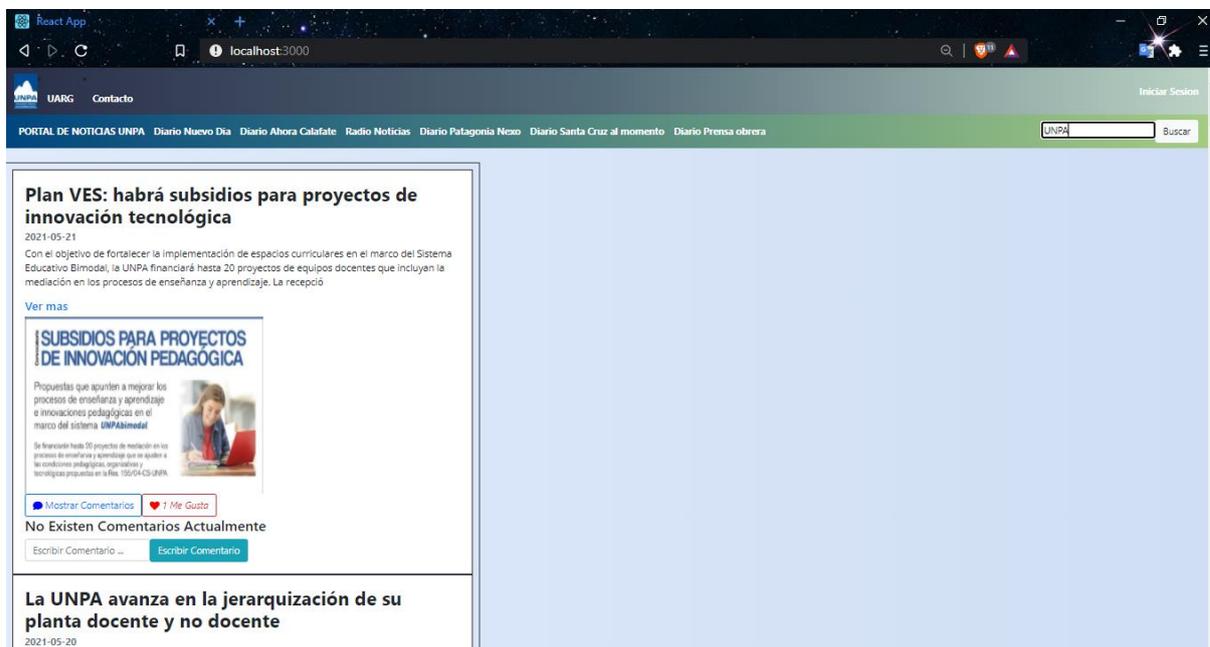


Figura 8 Imagen de búsqueda de noticia

4. GUIA DE RECOMENDACIONES

A continuación, se presentan las recomendaciones a partir de las experiencias del trabajo desarrollado:

- Estudiar y analizar los métodos para obtener y consumir los datos de los sindicadores web o feeds (RSS, ATOM), en función de los lenguajes de desarrollo del mashup.
- Examinar los formatos, estructuras y tipos de datos disponibles en las APIs o Feeds, debido a que cuando utilizan múltiples fuentes diversas requieren ser compatibilizadas para su integración.
- En la fase de implementación utilizar un entorno de desarrollo compatible con el entorno de producción, con el objetivo de evitar mayores demoras y esfuerzos.
- Supervisar la disponibilidad de los canales (RSS) para mantener la aplicación en funcionamiento.
- Adecuar los artefactos de desarrollo a las necesidades del proyecto, debido a que los requerimientos son diferentes de un proyecto a otro.

5. CONCLUSIONES

Este trabajo propuso estudiar y aplicar técnicas y estrategias para la extracción, composición e integración de contenidos proveniente de Periódicos digitales.

En cuanto a las guías de desarrollo, se debió analizar actividades que no estaban contempladas, pero que fueron necesarias para poder llevar a cabo las iteraciones. Estas fueron denominadas en la Guía de Desarrollo Mashup como Identificación de obstáculos y Acciones adicionales y necesarias. La primera actividad permitió llevar registros de los problemas encontrados en cada iteración, teniendo así la posibilidad de analizarlos y solucionarlos antes de avanzar a la siguiente iteración. En cambio, la segunda actividad sirvió para documentar tareas extras que se realizaron tendientes a mejorar el desarrollo.

La herramienta de diseño PIPES, fue útil para entender las operaciones lógicas a aplicar en los Feed (Combinar, Filtrar). También facilitó la representación de manera general el funcionamiento de la parte del Back-End del portal. Sin embargo, algunas limitaciones de PIPES se presentaron en cuanto a la integración en la operación de combinación exacta de los RSS y ATOM. Por este motivo el diagrama Pipes no refleja totalmente la implementación del portal web.

La dificultad principal en la integración de los Feed fue el inconveniente al momento de unificar los datos ya que entre los dos formatos (RSS o ATOM) se presentan grandes diferencias. Para superar este inconveniente se codificó un método que permite la unificación de los mismos para lograr un mejor manejo de los datos en la aplicación Web.

Otra dificultad se presentó en razón que algunos de los portales de noticias tales como “Ahora Calafate” y “Patagonia Nexa”, dejan temporalmente sin acceso a los respectivos feed (No se puede acceder al recurso en el servidor específico). Para resolver esta dificultad se debió evaluar la forma de manejar estas excepciones, la solución fue aplicar manejo de excepciones a través de métodos integrados de la versión de PHP, para que la aplicación funcione de manera normal y no tenga problemas en funcionamiento.

El Mashup de Noticias UNPA integra y compone noticias de 9 diarios digitales, a pesar de que existen más de 87 periódicos digitales en la provincia. Sucede que la mayoría no cuentan con



la tecnología de RSS o Atom, por lo cual tal vez no se obtengan todas las noticias que toman como enfoque la UNPA.

Para la instalación del portal en Linux fue necesario comprender las configuraciones necesarias para el funcionamiento de las herramientas esenciales de la aplicación, tal como el servidor Apache y los permisos necesarios para acceder y modificar los archivos de configuración del servidor. Este trabajo adicional fue necesario ya que no se contaba con la experiencia para poder realizarlo.

Como conclusión final, el desarrollo Mashup no es una tarea sencilla aunque lo parezca ya que sin importar el conocimiento a priori de las herramientas es obligatorio realizar un análisis para encontrar problemas referido a la integración de las fuentes y disponibilidad por parte de los proveedores.

AGRADECIMIENTOS

A Juan Enríquez y Franco Trinidad por ayudarnos con sus conocimientos a desplegar la aplicación web en el servidor de la UNPA.

REFERENCIAS

- CABALLERO VALDIVIA J. J. (2016) Modelo de Procesos para el Desarrollo Front-end. Interfaces. Lima: Universidad de Lima Facultad de Ingeniería de Sistemas ISSN: 1993-4912
- CASADO MANZANERO V. (2009) Diseño de un mundo virtual para la enseñanza de arquitectura de software. Escuela Politécnica Superior Universidad Carlos III de Madrid.
- COMUNIDAD RED HAT. (2007). Qué son las API y para qué sirven. Recuperado el 02/07/2020, de Red Hat, Inc. <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>
- CORDÓN, N. C. (2007). Los mashups dan un nuevo enfoque a la Web. Computer World. Recuperado (11/09/2020) <https://www.computerworld.es/archive/los-mashups-dan-un-nuevo-enfoque-a-la-web>
- FICHTER D. (2009) What is a Mashup?. More <Library Mashup> Exploring New Waves To Delivery Data. Nicole C. Engard.
- HERNÁNDEZ MURGUÍA A. (2007). Mashups: Qué Son y Qué No Son. Recuperado el 10/04/2020, de Software Gurú México. Sitio web: <https://sg.com.mx/content/view/256>
- KOSCHMIDER A., TORRES V., PELECHANO V. (2009) Elucidating the mashup hype: Definition, Challenges, Methodological guides and tools for mashup. 2nd Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web in conjunction with the 18th International World Wide Web. Universidad Politecnica de Valencia.
- MARROQUÍN BORJA. P.E (2016). Desarrollo de aplicaciones web e integración de datos usando mashup. Tesis ingeniería informática y ciencia de la computación de la Universidad Tecnológica Equinoccial de Quito.
- MENÉNDEZ. R (2018). ¿Qué es RSS? Recuperado el 22/06/2020, de Rafael Barzanallana Universidad de Murcia <https://www.um.es/docencia/barzana/PRACTICAS/RSS-Google-Reader.html>



- NIETO PEÑALVER L. (2012) Metodología de diseño para el desarrollo de mashups semánticos. Tesis de Magister en Ingeniería de Software. Universidad Nacional de La Plata 2012
- TEJEDOR. S (2007). Periodismo <<mashup>> Combinación de recursos de la web social con una finalidad periodística. Universidad autónoma de Barcelona.
- TEJEDOR.S., SORIA G., SANZ M., ORTUÑO X. (2011). La web 2.0 en iberoamérica: estrategias para una alfabetización ciberperiodística. Grupo de investigación Gabinete de Comunicación y Educación Universidad Autonomía de Barcelona.
- TINAJERO DIAZ I. E (2016). Composición de sistemas con Mashup. El caso Physical Trello. Tesis de Magister en ingeniería de Software. Universidad autónoma de Zacatecas México
- VARA MESA, J. M. (2000). Desarrollo web en entorno servidor (1.a ed.). Ra-Ma.