



EasyChair Preprint

Nº 1961

Library 2.0 of the Agrarian University of Havana

Amaury Machín Armenteros

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

November 16, 2019

Biblioteca 2.0 de la Universidad Agraria de La Habana

Amaury Machín Armenteros¹

*1-Master en Ciencias, Profesor, Departamento de Informática,
Facultad de Ciencias Técnicas, Universidad Agraria de la Habana,
Cuba. Carretera Tapaste, Km 22 ½, San José de las Lajas,
Mayabeque. Correo: amaurym@unah.edu.cu*

El desarrollo de aplicaciones y la provisión de servicios basados en Código Abierto.

Resumen

En las universidades cubanas existe la necesidad de contar con espacios donde los estudiantes y profesores puedan obtener información referente a temas de interés y no necesariamente sean de manera presencial. Como respuesta a esto se crean las bibliotecas virtuales y los repositorios de recursos digitales.

El presente artículo se centra en el diseño e implementación de una biblioteca 2.0 para la biblioteca de la Universidad Agraria de la Habana que integre las plataformas bVirtual y eBiblio mediante el uso de tecnologías y servicios de código abierto. Para el desarrollo y diseño de la biblioteca 2.0, después de un gran análisis sobre las tecnologías existentes en el mundo, se decidió usar una aplicación basada en servicios web, donde se empleó para el trabajo en el back-end Spring Framework, un framework JAVA y para el trabajo en el front-end AngularJS un framework JavaScript desarrollado por Google y con una gran comunidad de desarrollo. Además, se decidió utilizar Spring Boot que es una herramienta que nace con la finalidad de simplificar aún más el desarrollo de aplicaciones basadas en Spring Core Framework.

La biblioteca 2.0 incorpora una red social y un sistema de recomendación usando el algoritmo K-Vecinos más Cercanos, permitiendo que los usuarios compartan y personalicen su información que posibilitaría el aprovechamiento de la inteligencia colectiva.

Palabras clave: Biblioteca 2.0, Red Social, Sistema de Recomendación, K-Vecinos más Cercanos, Spring Boot.

Materiales y Métodos

Los sistemas de recomendación (SR) son herramientas que permiten asistir a los usuarios en sus procesos de búsqueda de información. Entre los ámbitos más extendidos se encuentran las recomendaciones de productos en tiendas online, películas, vídeos, música, libros o recomendaciones de perfiles a los que seguir en las

RS. Los recomendadores habitualmente son de dos tipos: los que hacen un filtrado colaborativo y los que hacen un filtrado basado en contenido. En este contexto, un filtro es el algoritmo matemático que “decide” cuál es la recomendación óptima basada en los datos que se le entregue.

Los filtrados colaborativos generalmente basan su lógica en las características del usuario. Los datos que se tienen del usuario se convierten en el centro de un filtrado colaborativo. El sistema analiza las decisiones anteriores, las preferencias, las calificaciones que ha dado de otros recursos, la valoración de los mismos, etc. y busca otros usuarios que se parecen a él y que han tomado decisiones parecidas. Los recursos que han tenido éxito con usuarios similares, seguramente también le interesarán al nuevo usuario (González, 2019).

Los filtrados basados en contenido tienen el producto como base de la predicción, en lugar de tener al usuario. Es decir, utiliza las características del artículo para hacer las recomendaciones (González, 2019).

Los SR son herramientas que necesitan de una fuente de información para poder trabajar, como las RS son manejadoras de un gran número de información, estos datos existentes se pueden procesar y así generar recomendaciones que de otra forma no se procesaría debido a su extensión. En la actualidad son muchas las plataformas que usan estas dos tecnologías, como por ejemplo Facebook, LinkedIn, Youtube, entre otras (Cleomar, 2011).

Como se ha visto anteriormente son muchas las ventajas de un SR en una RS, es por ende que con la integración de estas dos herramientas en las bibliotecas virtuales se crea un sistema pensado única y exclusivamente para mejorar la experiencia del usuario y que este encuentre un ambiente en el que se sienta cómodo. Actualmente se cuenta con varias plataformas que nos ofrecen todos estos servicios dos de las más reconocidas son **Oh!Libros** y **Goodreads**.

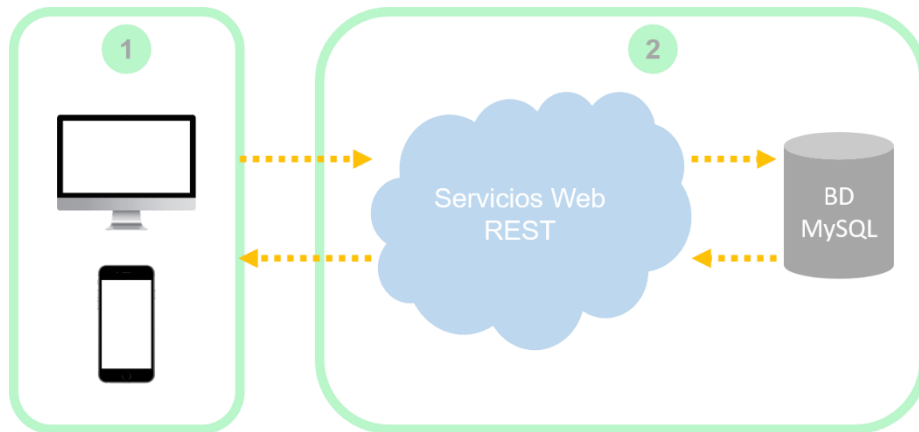
K-vecinos más cercanos

Una de las ramas de la Inteligencia Artificial que se ocupa del diseño y la aplicación de algoritmos de aprendizaje es Machine Learning (Mena, 1999). Dicha rama mediante el reconocimiento de patrones clasificar o identificar objetos en clases o grupos, principalmente de acuerdo a sus propiedades o características (Vázquez, 2008). Dentro de los algoritmos de aprendizaje de Machine Learning más usados para resolver problemas de clasificación se encuentra el **K-Vecinos más Cercanos** o el **K-Nearest-Neighbor (KNN)**. KNN es un algoritmo basado en instancia de tipo supervisado ideal para

introducirse en el mundo del aprendizaje automático. Sirve esencialmente para clasificar valores buscando los puntos de datos “más similares” (Jannach y otros, 2011) (Bagnato, 2018; HARRINGTON, 2012).

El algoritmo trabaja con un conjunto existente de datos llamado conjunto de entrenamiento. Cuando nos dan un nuevo dato lo clasifica en el grupo que corresponda, según tenga K vecinos más cerca de un grupo o de otro. Es decir, calcula la distancia del elemento nuevo a cada uno de los existentes, y ordena dichas distancias de menor a mayor para ir seleccionando el grupo al que pertenecer. Este grupo será, por tanto, el de mayor frecuencia con menores distancias. El K en KNN se refiere al número de elementos (ítems) que generalmente es menor que 20. En contraste con otros algoritmos de aprendizaje supervisado, **KNN** no genera un modelo fruto del aprendizaje con datos de entrenamiento, sino que el aprendizaje sucede en el mismo momento en el que se prueban los datos de test. A este tipo de algoritmos se les llama *Lazy Learning Methods* (Segaran, 2007) (Jannach y otros, 2011) (Ricci y otros, 2011) (Harrington, 2012).

A continuación, se presenta el escenario de actuación al cual se está sometido.



Fuente: Elaboración propia

Figura 1 Esquema básico de funcionamiento del proyecto

Haciendo referencia a la numeración indicada en la ilustración, se puede comentar que el proyecto se encuentra dividido en dos partes fundamentales conocidas como lado cliente **front-end** (1) y lado servidor **back-end** (2) usando **Servicios Web REST** como medio de comunicación.

Servicios web

La World Wide Web Consortium (W3C) define un servicio web como un sistema de software designado para dar soporte a la interacción de máquina a máquina interoperativa a través de una red (Center,

2014). Un Servicio Web o (Web Service) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones permitiendo presentar información dinámica al usuario. Uno de estos protocolos es la Transferencia de Representación de Estado (REST).

REST es un estilo de arquitectura software que se apoya totalmente en el estándar HTTP. Es una alternativa en auge a otros protocolos estándar de intercambio de datos como SOAP (Simple Object Access Protocol), que disponen de una gran capacidad, pero también mucha complejidad (BBVAOpen4U, 2014).

El principio básico del diseño de REST establece una correlación individual entre las operaciones de crear, leer, actualizar y borrar (CRUD) y los métodos HTTP. Según esta correlación:

- Para crear un recurso en el servidor hay que utilizar un POST.
- Para recuperar un recurso hay que utilizar un GET.
- Para cambiar el estado de un recurso, o para actualizarlo, hay que utilizar un PUT.
- Para eliminar o borrar un recurso hay que utilizar un DELETE.

Partiendo de la base de que toda comunicación en internet se mueve a través de recursos, y que estarán representados en un determinado formato, este principio posibilita el uso del servidor web desde diferentes clientes, escritos en distintos lenguajes de programación, y funcionando en diferentes plataformas. Dentro de los formatos que usa REST para el intercambio de datos se encuentra JSON.

JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript. Puede representar cuatro tipos primitivos (cadenas, números, booleanos, valores nulos) y dos tipos estructurados (objetos y arreglos) (Sanchez, 2014).

Los Servicios Web permiten crear aplicaciones escalables, seguras, que sean visible y simple y que se puedan separar los recursos y la representación.

Front-end

El **front-end** son todas aquellas tecnologías que corren del lado del cliente, es decir que corren del lado del navegador web, generalizándose más que nada en tres lenguajes, **HTML**, **CSS** y **JavaScript**, los desarrolladores encargados del front-end, se dedican solo a estas tres tecnologías (Diaz, 2014).

HTML5 (Hyper Text Markup Language) es un lenguaje usado para estructurar y presentar el contenido para la web, provee básicamente tres características: estructura, estilo y funcionalidad. Es uno de los aspectos fundamentales para el funcionamiento de los sitios, pero no es el primero. Se trata de un sistema para formatear el layout de las páginas web (Gauchat, 2012). Con HTML5, los navegadores como Firefox, Chrome, Explorer, Safari y más pueden saber cómo mostrar una determinada página web, saber dónde están los elementos, dónde poner las imágenes y dónde ubicar el texto.

CSS (hojas de estilo en cascada) es un lenguaje que define la apariencia de un documento escrito en un lenguaje de marcado (por ejemplo, HTML). Así, a los elementos de la página web creados con HTML se les dará la apariencia que se desee utilizando CSS: colores, espacios entre elementos, tipos de letra, separando de esta forma la estructura de la presentación.

Esta separación entre la estructura y la presentación es muy importante, ya que permite que sólo cambiando los CSS se modifique completamente el aspecto de una página web. Esto posibilita, entre otras cosas, que los usuarios puedan usar hojas de estilo personalizadas (como hojas de estilo de alto contraste o de accesibilidad) (Bustos, 2018).

JavaScript es un lenguaje de programación que permite realizar actividades complejas en una página web. Fue creado por Netscape, es una tecnología que ha sobrevivido por más de 10 años, es la tercera capa del pastel de los estándares en las tecnologías para la web, dos de las cuales son (HTML y CSS). Permite crear contenido nuevo y dinámico, controlar archivos de multimedia, crear imágenes animadas y muchas otras cosas más. Su sintaxis es similar a la usada en Java y C, al ser un lenguaje del lado del cliente este es interpretado por el navegador, no se necesita tener instalado ningún Framework (Pérez Valdés, 2007). Sin embargo, en la actualidad existen varios framework que son contruidos por desarrolladores expertos que permiten crear sitios web seguros, que validan estándares y cargan rápido. Uno de los más usados es AngularJS.

AngularJS es un framework MVC de JavaScript para el desarrollo web front-end que permite crear aplicaciones SPA Single-Page Applications. Entra dentro de la familia de frameworks como BackboneJS o EmberJS. Es un proyecto de código abierto mantenido por Google. Su objetivo es aumentar las aplicaciones basadas en navegador, en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles (Basalo & Alvarez, 2014). Dentro las ventajas de AngularJS se encuentra que es totalmente extensible y

funciona bien con otras bibliotecas una de ellas es el framework de estilos **AngularJS Material** (<https://material.angularjs.org/latest/>), el cual es tanto un marco de componentes de UI como una implementación de referencia de la especificación de diseño de materiales de Google. Este proyecto proporciona un conjunto de componentes de UI reutilizables, probados y accesibles basados en Material Design (Material, 2018).

El uso de estas tecnologías front-end permiten al desarrollador crear un entorno extraordinariamente expresivo, legible y rápido de desarrollar.

Back-end

El programador back-end es aquel que se encuentra del lado del servidor, es decir, esta persona se encarga de lenguajes como PHP, Python, .Net, Java, etc, se encarga de interactuar con bases de datos, verificar manejo de sesiones de usuarios, montar la página en un servidor, y desde éste "servir" todas las vistas que el front-end crea, es decir, se encarga más que nada de la manipulación de los datos (Diaz, 2014).

Java™ Platform Standard Edition (Java SE), fue seleccionada no por alguna característica especial en su lenguaje, porque los lenguajes por si solos no son muy útiles, sino por la considerable cantidad de librerías escritas ya sea por los desarrolladores de la plataforma (Sun Microsystem) o por terceras partes.

JAVA es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo. Java es la plataforma más extendida en el entorno corporativo. Se trata de una tecnología muy madura y popular que cuenta con innumerables herramientas de todo tipo y es bastante sencillo encontrar programadores (Montoro, 2013). Actualmente grandes plataformas como Netflix la cual consume el 15% del tráfico de internet en todo el mundo usan JAVA (Udemy, 2014). De acuerdo con el Índice de Lenguajes TIOBE("TIOBE," 2018), JAVA está ubicado como primer lugar dentro de los lenguaje de programación.

El uso de frameworks es extremadamente común en la actualidad, y está dado por una buena razón: los frameworks ayudan construir aplicaciones de forma rápida y fácil. El framework elegido para brindar los Servicios Web REST y manejar el acceso a los datos es **Spring Framework**.

Spring es un framework del lenguaje de programación JAVA de código abierto, escrito inicialmente por Rod Johnson, fue lanzado

por primera vez en el mes de junio del año 2003 bajo la licencia Apache 2.0. Convirtiéndose desde entonces en el framework más popular para JAVA Empresarial, ya que permite desarrollar aplicaciones de manera más rápida, eficaz y corta, saltándonos tareas repetitivas y ahorrándonos líneas de código. Son muchos los beneficios que nos proporciona en los que se destacan su organización de forma modular. Utiliza algunas de las tecnologías existentes, el framework web de Spring es un framework MVC web bien diseñado y cuenta con suficiente documentación en la web y con una gran comunidad (Muradas, 2018). Dentro de sus módulos se encuentra **Spring MVC** basado en HTTP y servlets, que provee herramientas para la extensión y personalización de aplicaciones web y servicios web REST (Alvarez, 2013).

Spring Framework ha evolucionado con el paso del tiempo y se han creado herramientas como **Spring Boot** que nace con la finalidad de simplificar aún más el desarrollo de aplicaciones basadas en este. Proporciona una forma más sencilla y rápida de configurar y ejecutar aplicaciones web. En el núcleo de Spring Core, necesita configurar todas las cosas por sí mismo, por lo tanto, puede tener muchos archivos de configuración, como los descriptores XML. Ese es uno de los principales problemas que Spring Boot resuelve, configura automáticamente todas las funciones que desea usar y puede iniciar la aplicación con un solo clic.(Raffai, 2018). Otro de los módulos usado de este framework es **JDBCTemplate** que nos permite la gestión a nuestras bases de datos. Además, se maneja la seguridad con **Spring Security** que proporciona servicios de seguridad para aplicaciones de software empresariales, a través del cual se determina si un cliente tiene autoridad, o autorización, para acceder a ciertos recursos protegidos. Spring Security hace uso de **JSON Web Token** para crear *tokens* de acceso que nos permiten brindar seguridad a las comunicaciones entre cliente y servidor (Paszniuk, 2016). JWT es un standard RFC 7519 que define un método compacto, autónomo y seguro para transmitir información entre dos sistemas. Se usa un objeto JSON de forma segura y mucho más ligera que XML por ser compacto y autónomo (Fonseca, 2017).

Compacto: es de tamaño pequeño, se puede usar en un HTTP header, como parámetro **GET** o **POST**.

Autónomo: el payload contiene toda la información necesaria para el caso de uso. En autenticación por ejemplo puede tener toda la información del usuario sin tener que consultar la base de datos.

El uso de estas tecnologías back-end permiten al desarrollador crear aplicaciones basadas en Spring Framework que son rápidas de construir y seguras.

Las herramientas para desarrollar el sistema fueron seleccionadas en dependencia de lo que se desee implementar. Por tanto, el criterio de selección está dado en escoger la herramienta que brinde mejor soporte para la tecnología, framework o estrategia a utilizar, por las habilidades de los desarrolladores en esta y/o por el tiempo necesario de capacitación.

Como Entorno de Desarrollo Integrado (IDE) usado para el desarrollo del proyecto se ha elegido **IntelliJ IDEA**. Es desarrollado por JetBrains, permite escritura de código sin complicaciones, crea un entorno adecuado en el que todos los miembros del equipo pueden trabajar juntos de manera eficiente (Amoedo, 2017).

Para procesar, describir, administrar y recuperar los datos almacenados en una base de datos se ha usado el Sistema Gestor de Base de Datos (SGBD, en inglés DBMS: DataBase Management System) **MySQL**. Este SGBD es un sistema de gestión de base de datos relacional multihilo y multiusuario, lo que le permite ser utilizado por varias personas al mismo tiempo, e incluso, realizar varias consultas a la vez, lo que lo hace sumamente versátil (Rouse, 2015). MySQL forma parte de la herramienta **XAMPP** la cual es un servidor independiente de plataforma y de software libre.

Una vez construidos nuestros Servicios Web REST el siguiente paso a seguir es probar cada uno de estos, hoy en día existen varios clientes REST que lo permiten. Uno de los más usados es **Postman**, herramienta dirigida a desarrolladores web que permite realizar peticiones HTTP a cualquier API. Postman es muy útil a la hora de programar y hacer pruebas, puesto que nos ofrece la posibilidad de comprobar el correcto funcionamiento de nuestros desarrollos (Postman, 2019).

Conclusiones

La Biblioteca 2.0 se ha desarrollado en función de resolver la problemática que se presentaba. Al término de este trabajo se puede concluir, de manera general, que se han cumplido todos los objetivos trazados al inicio del mismo:

- La incorporación de una red social a la biblioteca virtual permitió contar con medios propios de comunicación y constituir una imagen virtual de la institución para obtener una mayor visibilidad.
- La incorporación de un sistema de recomendación a la biblioteca virtual permitió brindar dinámicamente contenido personalizado a los usuarios, en base a sus preferencias o intereses.
- La disponibilidad del código fuente hace posible que usuarios, programadores y empresas se involucren en el desarrollo de la aplicación. De esta forma, el proceso de detección y corrección de

errores se lleva a cabo de forma eficiente, así como la implementación de nuevas características.

- Con el *software* de código abierto no existe un gasto de dinero en la compra de licencias, sino una inversión en la capacitación del personal.

Referencias

- Alvarez, C. (Producer). (2014). GENBETA. Retrieved from <https://www.genbeta.com/desarrollo/que-es-maven>
- Alvarez Caules, C. (Producer). (2013). ArquitecturaJava. Retrieved from <https://www.arquitecturajava.com/spring-mvc-configuracion/>
- Amoedo, D. (Producer). (2017). Uibunlog. Retrieved from <https://ubunlog.com/intellij-idea-ide-java/>
- Bagnato, J. I. (Producer). (2018). APRENDE MACHINE LEARNING. Retrieved from <http://www.aprendemachinelearning.com/clasificar-con-k-nearest-neighbor-ejemplo-en-python/>
- Basalo, A., & Alvarez, M. A. (Producer). (2014). DesarrolloWeb.com. Retrieved from <https://desarrolloweb.com/articulos/que-es-angularjs-descripcion-framework-javascript-conceptos.html>
- BBVAOpen4U (Producer). (2014). BBVAOpen4U API REST: qué es y cuáles son sus ventajas en el desarrollo de proyectos. Retrieved from <https://bbvaopen4u.com/es/actualidad/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos>
- Bustos, G. (Producer). (2018). Hostinger. Retrieved from <https://www.hostinger.com.ar/tutoriales/que-es-javascript-introduccion-basica/#gref>
- Center, I. K. (Producer). (2014). IBM Knowledge Center ¿Qué es un servicio web? Retrieved from https://www.ibm.com/support/knowledgecenter/es/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ac55710.htm
- Cleomar Valois B. Jr, M. A. d. O. (2011). Recommender systems in social networks. Retrieved 8 de marzo, 2019, from <http://dx.doi.org/10.1590/S1807-17752011000300009>
- Díaz, I. A. (Producer). (2014). SERPROGRAMADORES. Retrieved from <https://serprogramador.es/que-es-frontend-y-backend-en-la-programacion-web/>
- Fonseca, J. (Producer). (2017). Laraveles. Que es en realidad JWT. Retrieved from <https://laraveles.com/que-es-en-realidad-jwt/>
- González, A. (2019). Sistemas de Recomendación de contenidos con Machine Learning. Retrieved 8 de marzo, 2019

- Goodreads (Producer). (2019). Goodreads Retrieved from <https://www.goodreads.com/>
- HARRINGTON, P. (2012). *Machine Learning in Action*. United States of America: Manning.
- Material, T. W. A. (Producer). (2018). AngularJS Material. What is AngularJS Material? Retrieved from <https://material.angularjs.org/latest/>
- Mena, J. (1999). *Data Mining Your Website*: Digital Press.
- Montoro, S. (Producer). (2013). La Pastilla Roja. Cómo seleccionar una plataforma de desarrollo para un proyecto web. Retrieved from <https://lapastillaroja.net/2013/10/como-seleccionar-plataforma-tecnologica/>
- Muradas, Y. (Producer). (2018). OpenWebinars. Retrieved from <https://openwebinars.net/blog/conoce-que-es-spring-framework-y-por-que-usarlo/>
- Oh!Libro (Producer). (2019). Oh!Libro. Retrieved from <https://www.ohlibro.com/>
- Paszniuk, R. (Producer). (2016). Programacion.com.py. ¿Qué es Json Web Token (JWT)? Retrieved from <https://www.programacion.com.py/varios/que-es-json-web-token-jwt>
- Pérez Valdés, D. (Producer). (2007). MAESTROS DEL WEB. Retrieved from <http://www.maestrosdelweb.com/que-es-javascript/>
- Postman (Producer). (2019). Postman. Retrieved from <https://www.getpostman.com/>
- Raffai, Z. (Producer). (2018). ZoltanRaffai. Retrieved from <https://www.zoltanraffai.com/blog/what-is-spring-boot/>
- Rouse, M. (Producer). (2015). TechTarget. Retrieved from <https://searchdatacenter.techtarget.com/es/definicion/MySQL>
- Sanchez, J. (Producer). (2014). Frontend-Labs. Retrieved from <https://frontendlabs.io/1490--json-que-es-json-parse-json-stringify>
- Segaran, T. (2007). *Programming Collective Intelligence*. United States of America: O'Reilly Media, Inc.
- TIOBE. (2018). Retrieved from <https://www.tiobe.com/tiobe-index/>
- Udemy, F. (Producer). (2014). Internet YA - Soluciones Web. Retrieved from <https://www.internetya.co/php-vs-asp-net-costos-escalabilidad-y-rendimiento/>
- Vázquez, F. (2008). *Caracterización e interpretación de descripciones conceptuales en dominios poco estructurados*. Mexico.