



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE CIENCIAS

**DESARROLLO DE UN SERVICIO WEB PARA
CONSULTA Y EXTRACCIÓN DE CONSTANCIAS
PARA CURSOS, DIPLOMADOS Y ESPECIALIDADES
PARA EL SICECD**

**REPORTE DE
SERVICIO SOCIAL**

**QUE PARA OBTENER EL TÍTULO DE:
LICENCIADO EN CIENCIAS DE LA
COMPUTACIÓN**

P R E S E N T A :

JORGE ERICK RIVERA LÓPEZ



**TUTORA:
M. EN C. MARÍA GUADALUPE ELENA
IBARGÜENGOITIA GONZÁLEZ
2020**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

1. Datos del alumno
Rivera
López
Jorge Erick
55 36 48 79 30
Universidad Nacional Autónoma de México
Facultad de Ciencias
Ciencias de la Computación
312121648
2. Datos del tutor
M en C
María Guadalupe Elena
Ibargüengoitia
González
3. Datos del sinodal 1
Dra
Hanna Jadwiga
Oktaba
4. Datos del sinodal 2
Dr en Letras
Alfredo Juan
Arnaud
Bobadilla
5. Datos del sinodal 3
Mat
Salvador
López
Mendoza
6. Datos del sinodal 4
Dr
Francisco
Valdés
Souto
7. Datos del trabajo escrito
Desarrollo de un servicio web para consulta y extracción de constancias para cursos,
diplomados y especialidades para el SICECD
33 p
2020

ÍNDICE

Introducción

Capítulo 1 Objetivo general del servicio social

1.1 Necesidad del SICECD

1.2 Objetivo de Consultar/Extraer constancias

Capítulo 2 Tecnologías para el desarrollo

Capítulo 3 Desarrollo del caso de uso Consultar/Extraer constancias

3.1 Análisis de requerimientos

3.2 Especificación de requerimientos de software

3.3 Diseño

3.4 Construcción

3.5 Pruebas

Conclusiones

Bibliografía

Introducción

El trabajo de desarrollo de un servicio web para la extracción y consulta de constancias pertenece al proyecto de servicio social nombrado “Apoyo a la investigación. Desarrollo de un Sistema de Control Escolar para Cursos y Diplomados (SICECD)”, el cual tuvo como objetivo facilitar la elaboración de la oferta educativa y la administración de cursos sostenidos en plataformas de educación a distancia preexistentes.

Según los objetivos perseguidos, el SICECD debería contar con distintas funcionalidades que permitieran llevar un registro de todos los participantes, grupos y cursos, además de las inscripciones. También podría obtener los datos necesarios para que el usuario final pudiera realizar una mejor oferta educativa a los participantes, por lo cual estaría constantemente monitoreando los datos requeridos en las plataformas educativas para así extraerlos y dejarlos disponibles en el SICECD para su posterior administración.

En el proyecto estuvo involucrada la Facultad de Ciencias de la Universidad Nacional Autónoma de México, la cual desempeñó el rol de desarrollador a través de varios servicios sociales y trabajos de titulación.

El objetivo general del servicio social pretendió que los alumnos tuvieran una primera experiencia en un trabajo profesional de desarrollo de software, aplicando los conocimientos adquiridos en la carrera en Ciencias de la Computación con especial énfasis en el diseño e implementación de un sistema, como se ve en la licenciatura en materias tales como Modelado y Programación, Ingeniería de Software, entre otras.

El objetivo particular del trabajo fue exponer el desarrollo del caso de uso Consultar/Extraer constancias, el cual resuelve una parte del problema que se planteará más adelante.

El reporte de servicio social se divide en 4 capítulos. En el *Capítulo 1 Objetivo general del servicio social* se detalla a fondo el objetivo general del servicio social antes mencionado. También se justifica la importancia de la creación del SICECD y también el objetivo particular de uno de los casos de uso del proyecto del servicio social. En el *Capítulo 2 Tecnologías para el desarrollo* se describen conceptos sobre tecnologías usadas en la implementación de la solución. Por último, en el *Capítulo 3 Desarrollo del caso de uso Consultar/Extraer constancias*, se muestran distintas fases de la vida del proyecto en particular de uno de los casos de uso.

Capítulo 1 Objetivo general del servicio social

El servicio social, nombrado *Apoyo a la investigación. Desarrollo de un Sistema de Control Escolar para Cursos y Diplomados (SICECD)*, tuvo por objetivo que los alumnos tuvieran una experiencia cercana a la vida profesional en una empresa de software, ya que se planteó el trabajo con un equipo en el que cada estudiante tuviera un rol dentro de éste, ya sea como desarrollador o algún otro, además de reuniones con el cliente, por lo cual tendría un panorama de lo que es un proyecto en la vida real. También se pretendió que cada alumno participante aplicara los conocimientos adquiridos en la licenciatura de Ciencias de la Computación y mostrara sus habilidades para que pudiera fortalecerlas. De esta manera, se adquiriría una experiencia sumamente completa que permitiría que el estudiante desarrollara un software que resolviera lo requerido y, que una vez titulado, se integrara de forma óptima al mundo laboral.

1.1 Necesidad del SICECD

El sistema general, al que se le ha puesto por nombre Sistema de Control Escolar para Cursos y Diplomados (en adelante se le llamará SICECD), contendrá la solución propuesta al problema.

Durante el segundo semestre del año 2019, en el curso de Ingeniería de Software II se buscó a una persona que fungiera como un cliente que requiriera un software, el cual sería trabajado durante todo el curso por los alumnos mediante equipos de trabajo. El cliente resultó ser un profesor con doctorado en letras que dirige un área entera dedicada a brindar asesorías, cursos, diplomados y especialidades a docentes de todo el país, por lo que necesitaba de un software que pudiera administrar estos servicios, datos de los docentes y alumnos e inclusive contener una plataforma de aprendizaje a distancia.

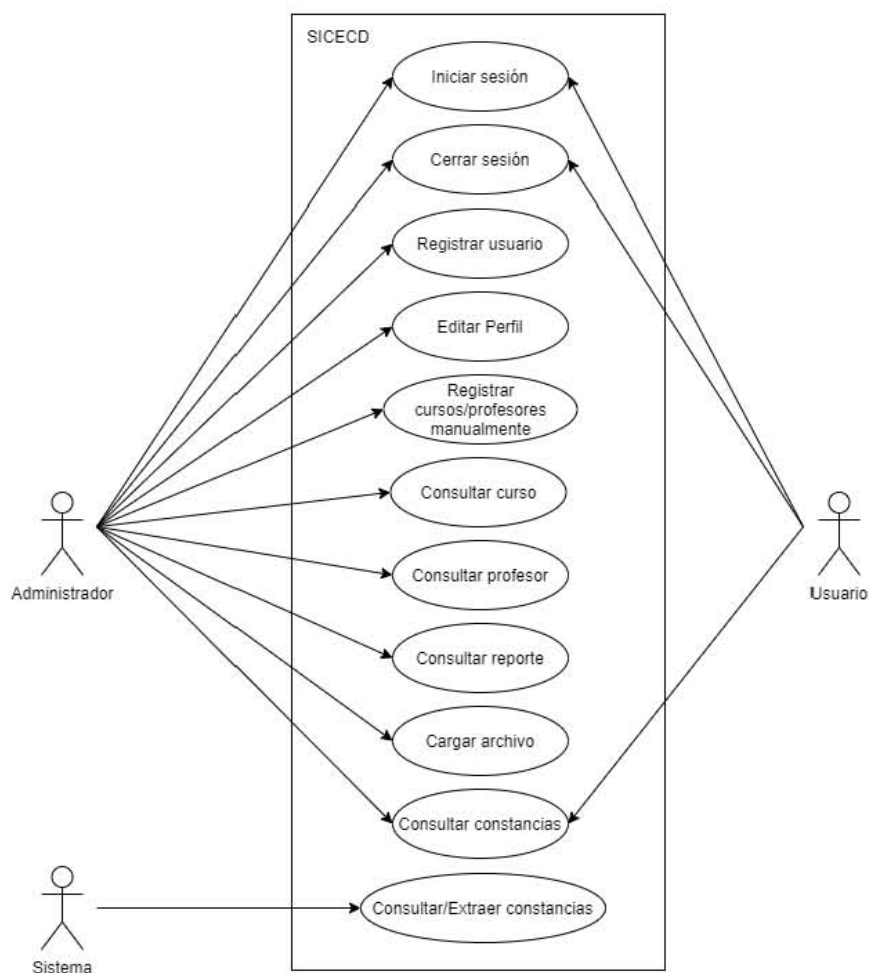
Lo anterior fue el preámbulo al planteamiento de la verdadera necesidad de dicho profesor, pues él ya contaba con el software necesario para cubrirla, a saber, Moodle. Esta plataforma, desarrollada en el lenguaje de programación PHP, ofrece educación a distancia: los alumnos pueden tomar clases; los docentes o asesores, tener un control de calificaciones, subir tareas, calificarlas, cargar material de estudio y establecer comunicación personal con los estudiantes. Cabe mencionar que los alumnos que toman los cursos son, asimismo, docentes en sus respectivas instituciones de origen. Sin embargo, la información del alumnado puede estar desintegrada y aunque Moodle ofrece un control sobre las inscripciones, cursos y grupos, al profesor le resulta más cómodo separar la parte administrativa de las asesorías, de la acción de los cursos. Con esto, podría generar un mayor provecho en cuanto a la oferta de estos últimos y proponer a ciertas personas diplomados o especialidades enfocados en los que anteriormente ya han completado.

Por otra parte, se generan y generarán constancias por cada curso, diplomado o especialidad. Si bien, por parte de Moodle sostienen que ya las generan, se necesita que éstas existan también en el área administrativa que será controlada en gran parte por el SICECD.

En resumen, el profesor requirió llevar un registro, control y seguimiento de los participantes de cada curso al que están inscritos para poder disponer de un historial por participante, con objeto de generar reportes útiles para las distintas áreas de la institución educativa que usará el software finalmente.

Diagrama general de casos de uso

En este diagrama se pueden apreciar los distintos casos de uso que comprendió el SICECD, y los actores que usarán el sistema, que podemos identificar como usuario Administrador, usuario Usuario y el Sistema mismo.



Este proyecto de Servicio Social pretendió resolver este problema, por lo que se propuso crear una aplicación web desarrollada con el lenguaje de programación Java usando Spring Framework. Luego, se aplicó el patrón de diseño Modelo Vista Controlador con un motor de base de datos PostgreSQL.

El equipo de desarrollo del proyecto estuvo integrado por alumnos de la Facultad de Ciencias, a los que se les asignó un caso de uso distinto.

1.2 Objetivo de la Consultar/Extraer constancias

En el marco general del SICECD, me fue asignado el caso de uso identificado como *Consultar/Extraer constancias*. Como se ha mencionado anteriormente, Moodle es una parte importante dentro del proyecto en general. Éste tiene un módulo para generar constancias para cada curso. Se ha requerido que estas constancias existan en el sistema administrativo del SICECD desarrollado con Spring framework.

No es tarea de este nuevo sistema el crear constancias, por lo que se ha requerido que:

- Se obtenga de alguna forma las constancias generadas en Moodle.
- Estos archivos tengan seguridad.
- Se mantengan las constancias actualizadas ya que cada cierto tiempo cambia su diseño.
- Adicionalmente, se ha requerido que la solución del problema esté adaptada para trabajar con distintas versiones existentes y por existir de Moodle.

Glosario de términos

A continuación, se muestra un glosario de términos necesarios para entender el resto del reporte.

Término	Descripción
Base de datos	Lugar que guarda datos acerca del SICECD.
Clave de grupo	Secuencia de caracteres que identifican unívocamente a un grupo de docentes que toman un curso a través de alguna versión de Moodle.
Clave de curso	Secuencia de caracteres que identifican unívocamente a un curso sostenido en alguna versión de Moodle.
Constancia	Documento que constata la aprobación de un Curso, Diplomado o Especialidad en alguna versión de Moodle.
Constancia nueva	Constancia no existente en el SICECD.
Lista de direcciones	Lista que contiene direcciones electrónicas de todos los servidores de las versiones de Moodle relacionados con el SICECD.
Profesor	Docente que está inscrito en algún curso de alguna versión de Moodle.

Sistema	Software que ha de resolver el problema de administración escolar (véase también SICECD).
Versión de Moodle	Es un software encargado de servir cursos de distintos temas en línea que contiene el servicio web de consulta/extracción de constancias y que es una copia legal de Moodle.
SICECD	Sistema de Control Escolar para Cursos y Diplomados.
Servicio web	Software que se encargará de enviar constancias solicitadas por el SICECD.
Tupla	Secuencia de elementos.

Haciendo un análisis de los requerimientos, es claro que la interacción con un usuario humano es, técnicamente, indirecta ya que no existe interfaz alguna que pueda utilizar el usuario final para accionar la funcionalidad que resuelva el problema, por lo que se han identificado dos casos de uso desprendidos del caso de uso original en los que sería de gran utilidad que la solución actúe en el sistema.

Hay que notar que el título del reporte denota un solo servicio web, por lo que los servicios propuestos más adelante, lógicamente, trabajan como parte de uno solo, así como los consumidores o clientes de dichos servicios.

Los casos identificados son los siguientes:

- La funcionalidad deberá trabajar cada cierto tiempo en el cual tendrá que extraer todas las constancias que no existan en el SICECD desde las versiones de Moodle o, en su caso, actualizar las constancias que lo necesiten.
- Cuando un usuario requiera una sola constancia no existente en el SICECD la funcionalidad permitirá la consulta y extracción de ésta.

Analizando lo anterior se ha propuesto crear:

- Un Servicio web que procese la solicitud de múltiples constancias, el cual estará integrado a cada versión de Moodle.
- Un Servicio web que procese la solicitud de una sola constancia, el cual estará integrado a cada versión de Moodle.
- Un Cliente que solicite periódicamente constancias nuevas o la actualización de las constancias ya presentes en el SICECD al primer servicio web ya mencionado.
- Un Cliente que solicite una constancia nueva al segundo servicio web ya mencionado, por lo que funcionará bajo demanda.

Por lo anterior, los servicios web serán desarrollados con el lenguaje de programación PHP porque deberán utilizar ciertas funciones existentes en Moodle, así como su base de datos.

Capítulo 2 Tecnologías para el desarrollo

En los siguientes párrafos se describirán, las herramientas más importantes que ayudaron a la implementación de la solución del caso de uso.

Servidor Apache Tomcat

El software Apache Tomcat es una implementación de código abierto de las tecnologías Java Servlet, JavaServer Pages, Java Expression Language y Java WebSocket. Pretende ser una colaboración de los mejores desarrolladores de todo el mundo además de que impulsa numerosas aplicaciones web críticas a gran escala en una amplia gama de industrias y organizaciones.

Este software se desarrolla en un entorno abierto y participativo y se publica bajo la licencia de Apache versión 2.¹

Java

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems.² Pertenece al paradigma de la Orientación a Objetos. Se dice que es seguro, pues no permite al programador hacer uso de apuntadores, como sí sucede en C o C++, además, realiza de forma automatizada la asignación y liberación de memoria dinámica utilizando un recolector de basura. Aunado a esto, permite la portabilidad de los programas ya que utiliza una máquina virtual llamada Java Virtual Machine (JVM) con lo cual un programa, una vez compilado, no tiene que volverse a compilar para poder ser usado en otro lugar.³

PHP

PHP es el acrónimo recursivo de *Hypertext Preprocessor*, este lenguaje de programación es de código abierto, principalmente se utiliza para el desarrollo web.

Las páginas de PHP tienen código incrustado en HTML, de tal forma que cuando el cliente accede a una página PHP recibe el resultado de haber ejecutado el código, así que el cliente no puede saber realmente que es lo que se realiza para dar el resultado final.

PHP también ofrece la posibilidad de ser utilizado para desarrollar aplicaciones de escritorio y de ser ejecutado sin necesidad de tener una aplicación web o de escritorio a través de la línea de comandos.⁴

PostgreSQL

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que usa y amplía el lenguaje SQL combinando muchas características que almacenan y escalan de manera segura las cargas de trabajo de datos más complicadas. Sus orígenes se remontan a 1986 como parte del proyecto POSTGRES en la Universidad de California en Berkeley y tiene más de 30 años de desarrollo activo en la plataforma central.

PostgreSQL se ha ganado una sólida reputación por su arquitectura comprobada, confiabilidad, integridad de datos, conjunto de características robustas, extensibilidad y la dedicación de la comunidad de código abierto detrás del software para ofrecer soluciones innovadoras y de alto rendimiento. PostgreSQL se ejecuta en todos los principales sistemas operativos, cumple con ACID desde 2001 y tiene complementos potentes como el extensor de base de datos geoespaciales PostGIS.⁵

Spring framework y Spring Boot

El primero es un framework de código abierto para la creación de aplicaciones empresariales. Funciona con Java, aunque también tiene soporte para trabajar con otros lenguajes. Con éste, se pueden realizar diferentes tipos de aplicaciones, tales como aplicaciones de escritorio, de línea de comandos, web, que acceden a bases de datos vía SQL, con esquemas de seguridad básica y aplicaciones que tienden a ser escalables.⁶

Por otra parte, Spring Boot también es un framework diseñado para poner en funcionamiento las aplicaciones hechas con Spring framework lo más rápido posible, con una configuración inicial mínima.⁷

Servicio web

Supongamos que existe la siguiente página web llamada www.tienda.com.mx, la cual se dedica a mostrar y vender artículos de distinta índole, como ropa, calzado, artículos deportivos, computadoras, celulares, libros etc. El usuario que accese en la página podrá comprar los productos que decida, además, encontrará un servicio de entrega a través de una empresa de paquetería que tiene, a su vez, un sitio propio nombrado www.paqueteria.com.mx. Cuando el usuario ha pagado sus productos, se le entrega un número de seguimiento de sus paquetes, mismos que pueden ser seguidos directamente desde la página de la paquetería o desde la tienda virtual. Internamente lo que hace www.tienda.com.mx es realizar una consulta a www.paqueteria.com.mx, por lo cual nos damos cuenta de que ambas aplicaciones asociadas a dichas páginas tienen un lenguaje en común para intercambiarse información.

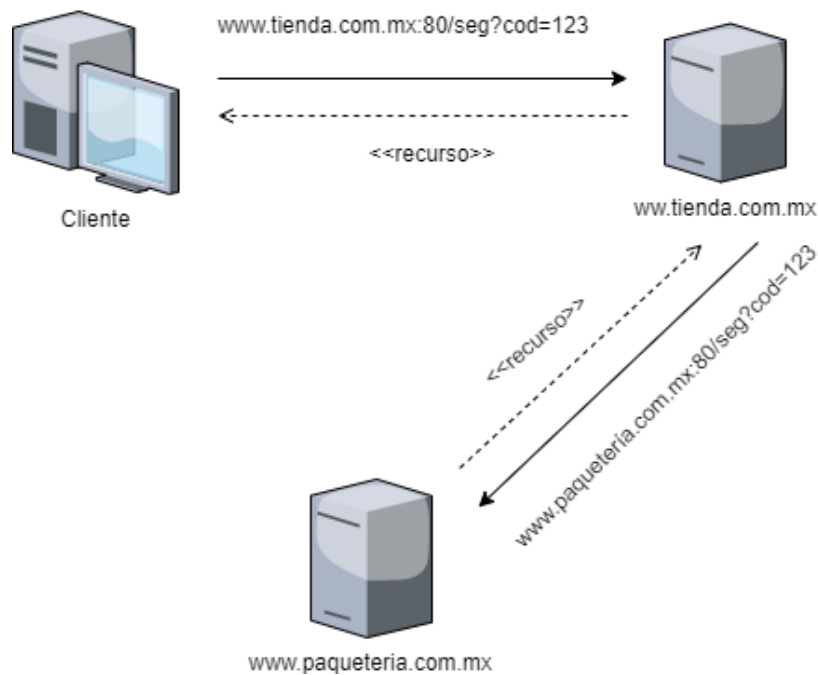


Figura 1. Interacción en la arquitectura REST

Las aplicaciones de red, hoy en día, se encuentran en muchas ocasiones con escenarios iguales o análogos, en los que una aplicación requiere cruzar su propia frontera para realizar una tarea. Se está hablando de que es necesario un protocolo que permita el rápido desarrollo de nuevas aplicaciones de red que puedan interactuar entre sí. Lo nuevo en el problema es la escala, no en cuanto el tamaño de la red, sino en la cantidad de diferentes tipos de aplicaciones de red.

A las arquitecturas que han sido propuestas como solución al problema se les llama *Servicio Web*, que toman su nombre del hecho de que una aplicación puede ofrecer acceso remoto a otras de tipo cliente-servidor para formar una red de aplicaciones.

Las arquitecturas propuestas para el manejo de los *Servicios web*, son REST y SOAP las cuales serán descritas a mayor profundidad, pero, para el propósito de este trabajo, se explicará con mayor detalle REST.

Lo que intenta la arquitectura SOAP (Simple Object Access Protocol) es hacer el problema lo suficientemente realizable, al menos en teoría, para tener protocolos que puedan ser personalizados para cada aplicación de red.

Los servicios web SOAP están basados en la suposición de que la mejor manera de integrar las aplicaciones a través de la red es utilizando protocolos que son personalizados por cada aplicación.

Arquitectura REST

A diferencia de SOAP, los servicios REST consideran que es mejor replicar el modelo con el que trabaja la arquitectura de la World Wide Web. Este modelo, articulado por Roy Fielding, es conocido como *REpresentational State Transfer* (REST). Una ventaja notable es que no hay necesidad de hacer una nueva arquitectura REST puesto que la arquitectura Web existente es suficiente, aunque son necesarias unas cuantas extensiones. En la arquitectura Web cada Servicio Web, como recurso, está identificado por una URI que puede ser accedida por HTTP.

REST usa un pequeño conjunto de métodos de HTTP, entre ellos, GET y POST. Éstos transforman la complejidad del uso de un protocolo al uso de la carga útil, que es la representación abstracta del estado de un recurso.

La interoperabilidad entre el cliente y el Servicio Web depende del acuerdo en la representación del estado de los recursos. Las formas más comunes son XML y JavaScript Object Notation (JSON).

Lo mismo ocurre con SOAP, la diferencia con REST radica en que el protocolo a utilizar entre el cliente y servicio web depende del acuerdo mutuo, mientras en REST, el protocolo es siempre HTTP.

La Web ha mostrado que la estandarización y la estabilidad se escalan muy bien. Asimismo, REST está provisto de seguridad con Secure Socket Layer/Transport Layer Security (SSL/TLS).⁸

JavaScript Object Notation (JSON)

JavaScript Object Notation o por su traducción al castellano Notación de Objetos de JavaScript, es un formato ligero de intercambio de datos, está basado en un subconjunto del Lenguaje de Programación JavaScript. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

JSON está constituido por dos estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.

- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

Estas son estructuras universales. Virtualmente, todos los lenguajes de programación las soportan de una forma u otra. Por tanto, es razonable que un formato de intercambio de datos que es independiente del lenguaje de programación se base en estas estructuras.

A continuación, se muestran los elementos que puede contener un JSON:

Objeto

Es un conjunto desordenado de pares nombre/valor. Un objeto comienza con llave de apertura { y termina con llave de cierre }. Cada nombre es seguido por dos puntos y los pares nombre/valor están separados por coma.

Arreglo

Es una colección de valores. Un arreglo comienza con corchete izquierdo [y termina con corchete derecho]. Los valores se separan por coma.

Valor

Un valor puede ser una cadena de caracteres con comillas dobles "", un número, *true* o *false*, *null*, un objeto, o un arreglo. Estas estructuras pueden anidarse.

Cadena

Una cadena de caracteres es una colección de cero o más caracteres Unicode encerrados entre comillas dobles, usando barras divisorias invertidas como escape. Un carácter está representado por una cadena de caracteres de un único carácter. Una cadena de caracteres es parecida a una cadena de caracteres C o Java.

Los espacios en blanco pueden insertarse entre cualquier par de símbolos.

Estos son los elementos que conforman un JSON en el estándar definido en el RFC 8259 publicado por la Internet Engineering Task Force (IETF).

```
{
  "nombre": "Juan",
  "edad": 22,
  "estatura": 1.80,
  "hermanos": ["Pedro","Pablo"],
  "numeros" : { [1,2,3]}
}
```

Figura 2. Ejemplo de JSON

Como se puede apreciar cada elemento tiene una posible representación en muchos de los lenguajes de programación. Por ejemplo, para Objeto pueden existir las estructuras de datos arreglos, listas o vectores, y de forma análoga para los tipo Arreglo, los valores o cadenas tendrán también una representación a través de valores numéricos que todo lenguaje de programación debería soportar.^{9,10}

Moodle

Este programa de código abierto permite administrar cursos, tareas y calificaciones; realizar consultas, intercambio de mensajes privados entre alumno-profesor, foros, talleres, añadir recursos y elaborar una Wiki. Es multilinguaje gracias a la aportación de la comunidad de desarrollo contando hasta el momento con traducción a 120 lenguajes. Está implementado con el lenguaje de programación PHP, y es compatible con MariaDB, MySQL, PostgreSQL, SQL Server, y Oracle como manejador de bases de datos.

Este software es gratuito y cualquiera puede extender y modificar el código fuente con fines comerciales o no. También permite la personalización de la interfaz y otros elementos, además que su uso es escalable en cantidad de usuarios: desde unos cuantos, hasta millones de usuarios.¹¹

Capítulo 3 Desarrollo del caso de uso Consultar/Extraer constancias

En este capítulo se mostrará el análisis, diseño, construcción y pruebas de la solución del caso de uso Consultar/Extraer de constancias.

3.1 Análisis de requerimientos

Recordemos que el sistema general a construir debe resolver los problemas de administración que acompañan algunas plataformas educativas, así como darle una mejor perspectiva al usuario final para generar una mejor oferta educativa.

Dentro de las funcionalidades del sistema, se requiere que las constancias de cursos, diplomados y especialidades expedidas originalmente en las plataformas educativas estén disponibles en el nuevo sistema. Así, deberá existir una funcionalidad que resuelva dicho problema. Su desarrollo, se me ha asignado.

A continuación, se ha identificado de forma más detallada lo que debería contener:

- La funcionalidad deberá trabajar cada cierto tiempo en el cual tendrá que extraer todas las constancias de las distintas versiones de Moodle que no existan en el SICECD, o, en su caso, actualizar las constancias que lo requieran.
- Cuando un usuario requiera una constancia no existente en el SICECD, la funcionalidad permitirá la consulta y extracción de ésta desde la versión de Moodle que la tenga.

3.2 Especificación de requerimientos de software

Al caso de uso se le ha puesto por nombre *Consultar/Extraer constancias*.

Los actores identificados son los siguientes:

- Usuario: Este actor tiene la capacidad de consultar su información contenida en el sistema, de ninguna forma podrá eliminar o modificar la información.
- Administrador: Este actor extiende las capacidades del usuario anterior, añadiendo la capacidad de modificar o eliminar la información del sistema.
- Sistema: SICECD

Este caso de uso como se observó en el *Capítulo 2 Planteamiento del problema*, tendría dos casos de uso incluidos, el primero se llamaría *Consultar/Extraer constancias masivamente* y el segundo *Consultar/Extraer constancia bajo demanda*.

Enseguida se muestra el diagrama general para el caso de uso Consultar/Extraer constancias.

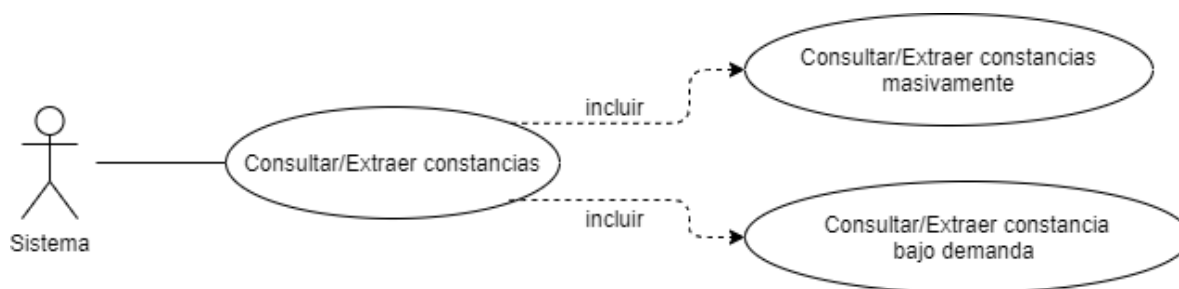


Figura 3. Diagrama general de caso de uso

Consultar/Extraer constancias masivamente

Detalle de caso de uso

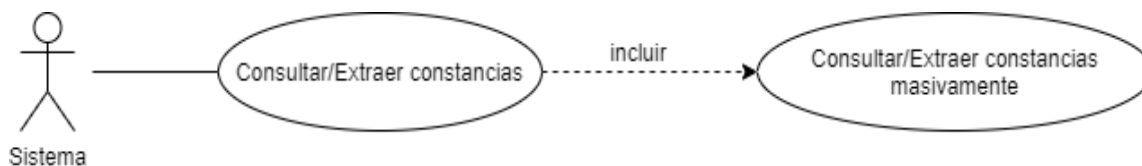


Figura 4. Diagrama particular de caso de uso

Actores:

Sistema

Descripción:

El SICECD emite un mensaje a los servicios web de consulta/extracción de constancias de las distintas versiones de Moodle pidiendo ciertas constancias provocando que cada servicio web devuelva las solicitadas (que existan dentro del servidor) y se guarden en el SICECD con seguridad de acceso para dejarlas disponibles para uso de éste mismo.

Precondiciones:

Se tiene la información necesaria acerca de los profesores, grupos, inscripciones, cursos y direcciones de las distintas versiones de Moodle, es decir, hay al menos una relación entre alumno, grupo, curso, inscripción e información de creación de constancias.

Flujo normal de eventos

Actor(es)	SICECD	Versión de Moodle	
Paso	Acción	Paso	Acción
1	Obtener un conjunto de tuplas. Cada una de éstas, incluye valores de correo electrónico del profesor, su clave de grupo, clave de curso al cual está inscrito, y tiempo de creación de constancia. Luego, enviar dicha relación.	2	Obtener el conjunto de tuplas y buscar las constancias dadas a las tuplas (A2) en su base de datos. Devolver un conjunto de tuplas, donde cada una incluye valores de correo electrónico del profesor, su clave de grupo, clave de curso al cual está inscrito y tiempo de creación de constancia; además de un mensaje indicando si se obtuvieron constancias o no. Luego, enviar dicha relación.
3	Obtener la respuesta positiva de la versión de Moodle (A1) y guardar la información obtenida en la Base de datos, además de agregar seguridad de acceso a los archivos.		
4	Probar con otra versión de Moodle en la lista de direcciones y volver al paso 1.		

Tabla 1. Flujo normal de eventos A

Flujo alternativo de eventos

ID	Nombre	Acción
A1	Se obtiene, como respuesta del Servicio web, un conjunto de tuplas con un mensaje no exitoso.	No realizar ninguna acción.
A2	Algún Servicio web obtiene un conjunto de tuplas con	No incluir nada en el nuevo conjunto de respuesta.

	datos inválidos.	
--	------------------	--

Tabla 2. Flujo alternativo de eventos A

Flujo excepcional de eventos

ID	Nombre	Acción
E1	Pérdida de conexión con la base de datos.	Escribir en una bitácora de eventos lo sucedido.

Tabla 3. Flujo excepcional de eventos A

Poscondiciones:

Las constancias estarán dentro del servidor del SICECD con seguridad de acceso y su información asociada estará disponible en la base de datos de éste.

Consultar/Extraer constancias bajo demanda

Detalle de caso de uso

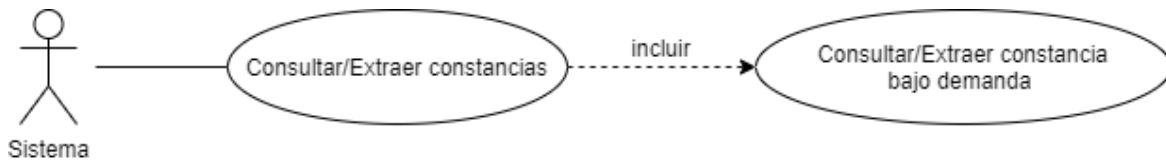


Figura 5. Diagrama particular de caso de uso

Actores:

Sistema

Descripción:

Un actor usuario o actor administrador solicita una constancia y ésta no se encuentra en la base de datos, por lo cual el Sistema emite un mensaje a los servicios web de las distintas versiones de Moodle solicitándola, provocando que algún servicio web responda con la constancia requerida.

Precondiciones:

Se tiene la información necesaria acerca del profesor, grupo, inscripción, curso y de las direcciones de las versiones de Moodle en el SICECD.

Flujo normal de eventos

Actor(es)	SICECD	Versión de Moodle	
Paso	Acción	Paso	Acción
1	Obtener la tupla con valores de correo electrónico del profesor, su clave de grupo, clave de curso al cual está inscrito y tiempo de creación de constancia. Luego, enviar dicha relación.	2	Obtener la relación y buscar la constancia dada la relación (A4) en su base de datos de constancias. Enviar una relación con valores de correo electrónico, clave de grupo del profesor, clave de curso al cual está inscrito y tiempo de creación de constancia; además de un mensaje indicando si se obtuvo la constancia o no. Luego, enviar dicha relación.
3	Obtener la respuesta positiva de la versión de Moodle (A3) y guardar la información obtenida en la Base de datos, además de agregar seguridad de acceso a los archivos.		
4	Probar con otra versión de Moodle en la lista de direcciones y volver al paso 1.		

Tabla 4. Flujo normal de eventos B

Flujo alternativo de eventos

ID	Nombre	Acción
A3	Se obtiene en la respuesta de la versión de Moodle un mensaje no exitoso.	No realizar ninguna acción.
A4	Algún Servicio web obtiene un conjunto de tuplas con datos no válidos.	No incluir nada en el nuevo conjunto de respuesta.

Tabla 5. Flujo alternativo de eventos B

Flujo excepcional de eventos

ID	Nombre	Acción
E2	Pérdida de conexión con la base de datos.	Escribir en una bitácora de eventos lo sucedido.

Tabla 6. Flujo excepcional de eventos B

Poscondiciones:

Las constancias estarán dentro del servidor del SICECD con seguridad de acceso y su información asociada estará disponible en la base de datos del SICECD.

3.3 Diseño

Los siguientes diagramas corresponden a los consumidores o clientes de los servicios web que pertenecen al SICECD y los servicios web que pertenecen a las distintas versiones de Moodle.

Diagrama de paquetes de los clientes de los servicios web

Se aprecia que sólo existirán los componentes Modelo y Controlador (que interactuarán entre sí) para los clientes de los servicios web.

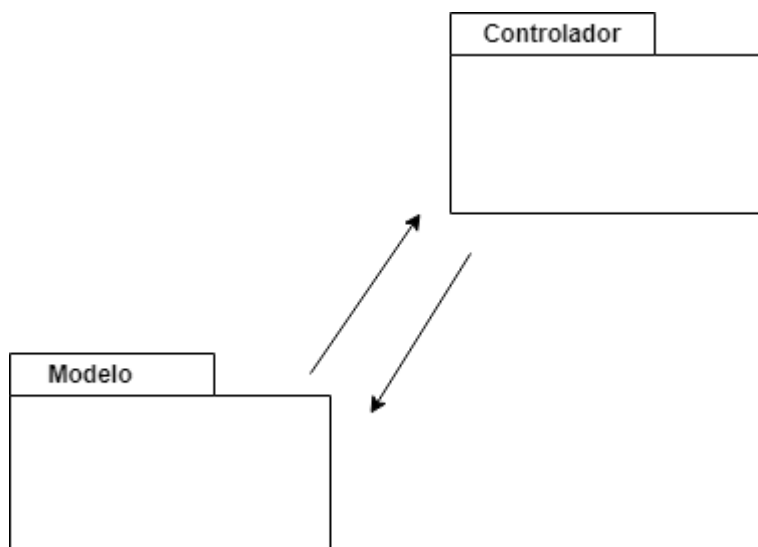


Figura 6. Diagrama de paquetes

Diagrama de distribución de los clientes de los servicios web

A continuación, se aprecia cómo los componentes Controlador y Modelo se hallan funcionando dentro de lo que será el servidor del SICECD.

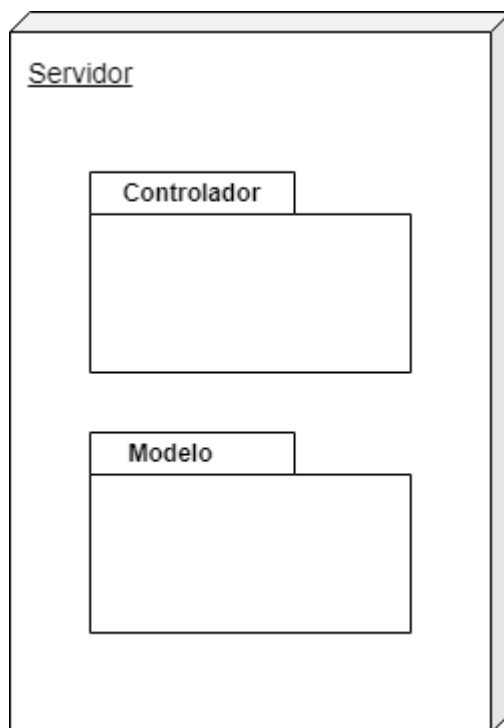


Figura 7. Diagrama de distribución

Definición del ambiente de implementación de los clientes de los servicios web

La siguiente tabla muestran las herramientas que se usarán para el desarrollo de la solución del problema.

Concepto	Herramienta	Versión
Framework	Spring Framework	5.0
Diagramador UML	draw.io	N/A
Manejador de bases de datos	PostgreSQL	10
Servidor de aplicaciones	Apache Tomcat	8.5.39
IDE	Spring Tools Suite	4.0
Control de versiones	Git	2.11

Tabla 7. Ambiente de implementación

Diagrama de clases para los clientes de los servicios web

En la imagen de abajo se muestran las clases que funcionarán para el Modelo y las que pertenecen al Controlador.

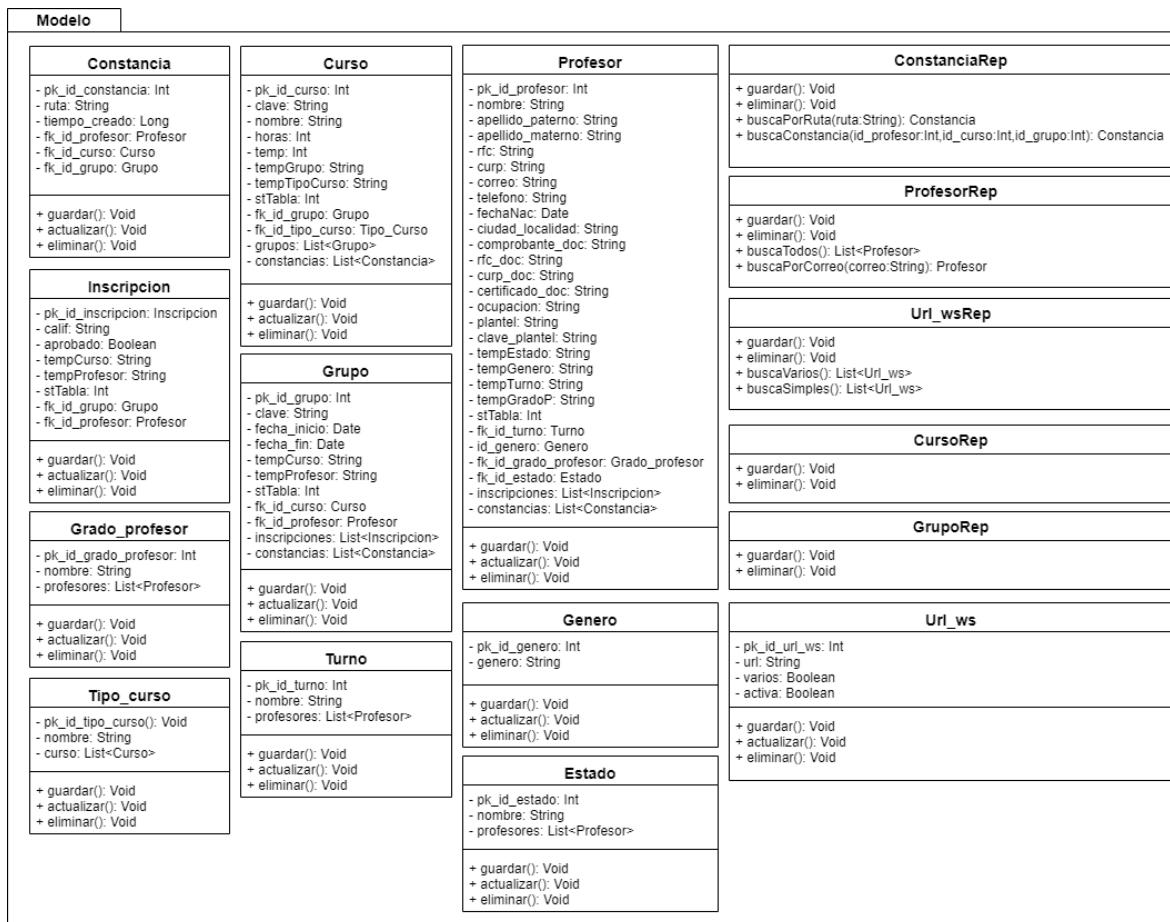


Figura 8. Diagrama de clases para el Modelo

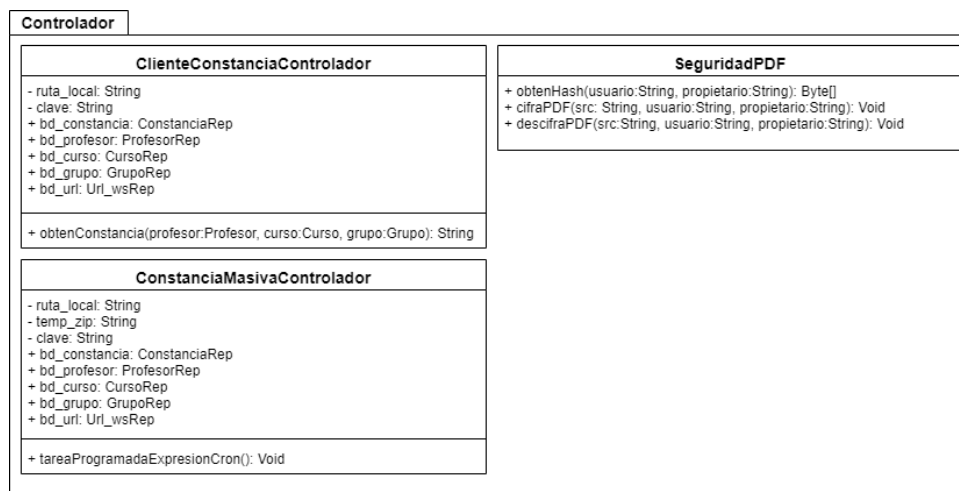


Figura 9. Diagrama de clases para el Controlador

Diagrama de secuencia de cliente del servicio web de consulta/extracción masiva de constancias

El siguiente esquema muestra la secuencia de acciones del cliente, desde la obtención de una relación de las distintas constancias que se quieren actualizar o traer por primera vez hasta el llamado a la función que envía la relación y procesa la respuesta del servicio web.

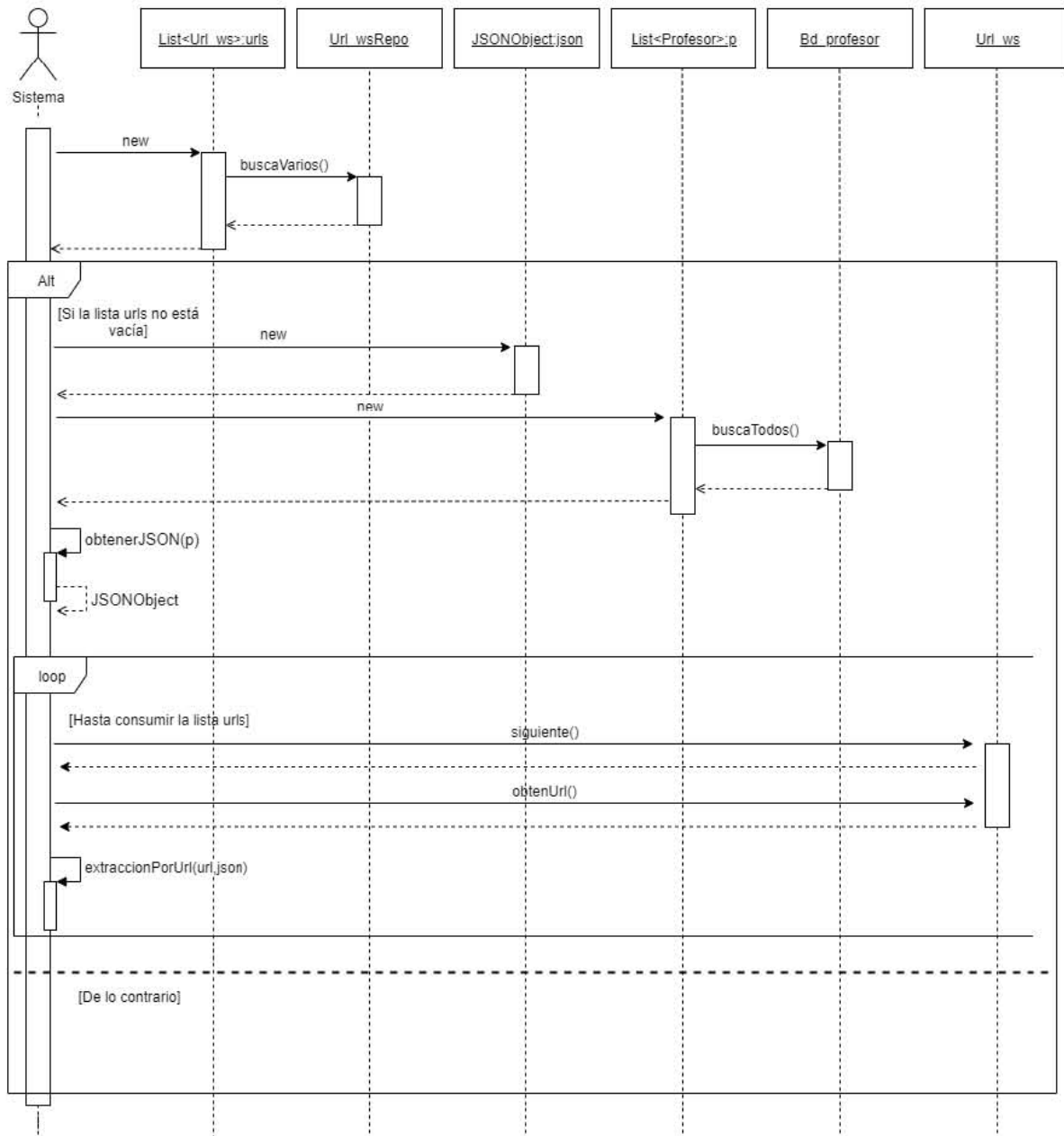


Figura 10. Diagrama de secuencia de cliente A

Diagrama de secuencia de cliente del servicio web de consulta/extracción bajo demanda de constancia

Asimismo, se muestra la secuencia de acciones realizadas por el cliente, desde la obtención de los datos necesarios para obtener una constancia hasta el procesamiento de la respuesta del servicio web.

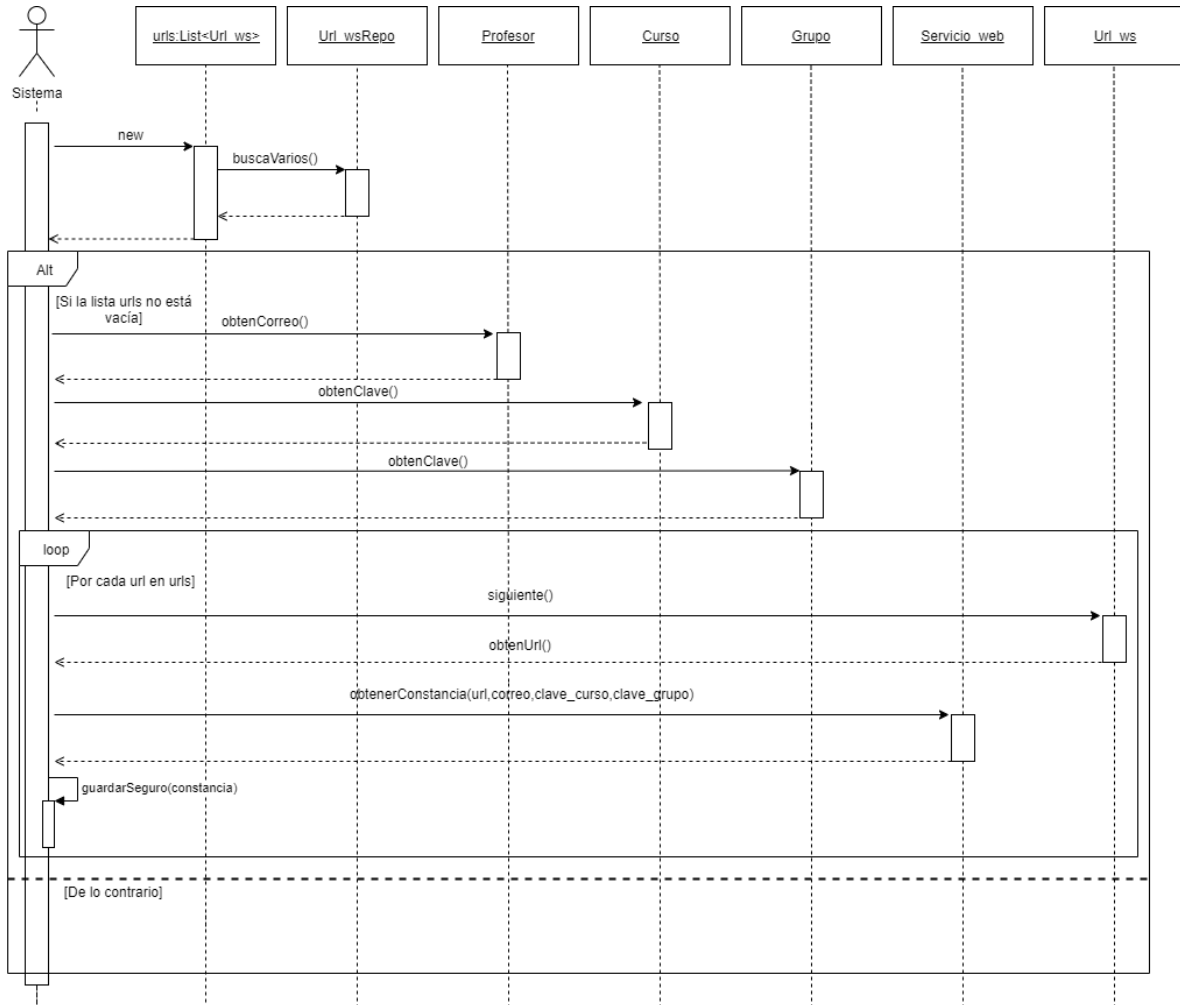


Figura 11. Diagrama de secuencia de cliente B

Diagrama de secuencia de servicio web de consulta/extracción masiva de constancias

Se muestra la secuencia de acciones realizadas por el servicio web, desde que recibe una petición de algún cliente hasta el envío de la respuesta al cliente.

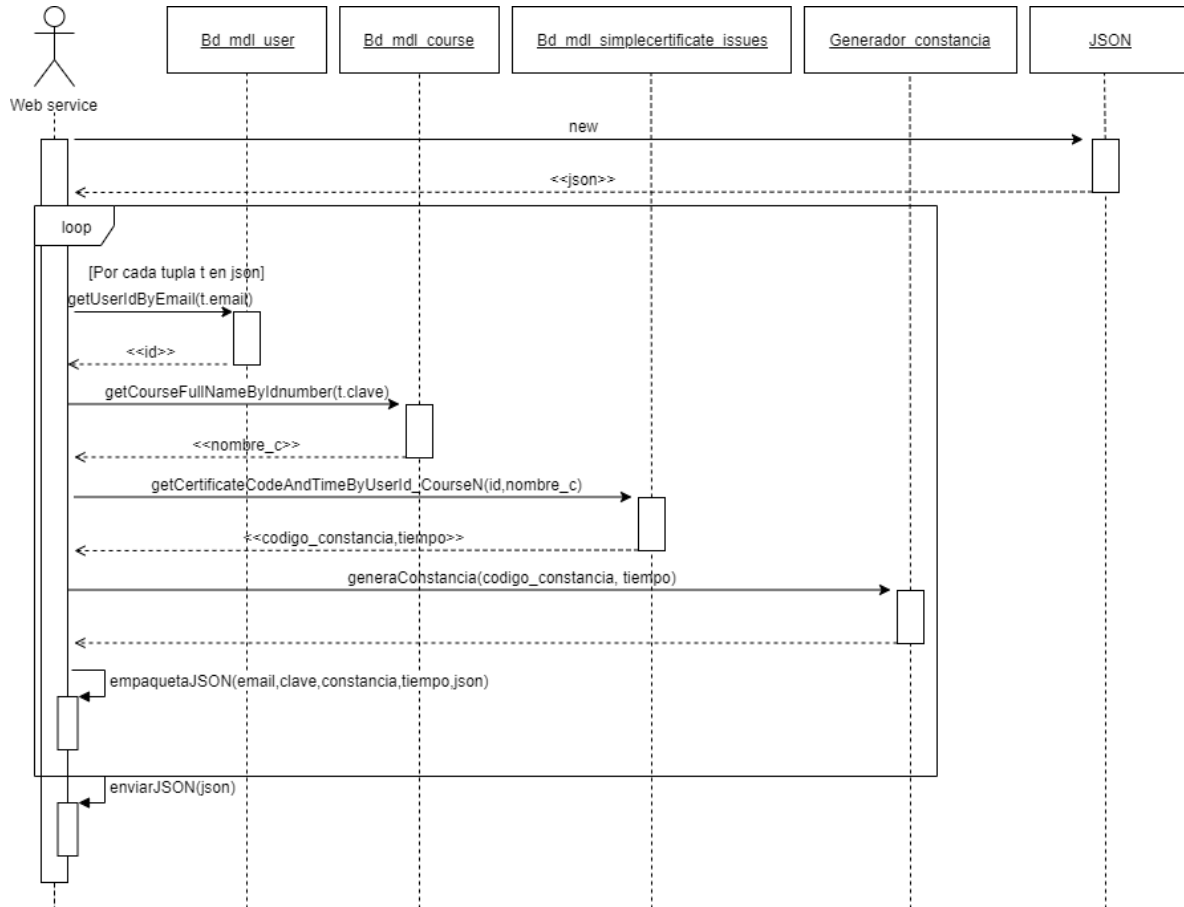


Figura 12. Diagrama de secuencia de servicio web A

Diagrama de secuencia de servicio web de consulta/extracción bajo demanda de constancia

Ahora, se muestra la secuencia de acciones realizadas por el servicio web, desde que recibe una petición de algún cliente hasta el envío de la respuesta al cliente.

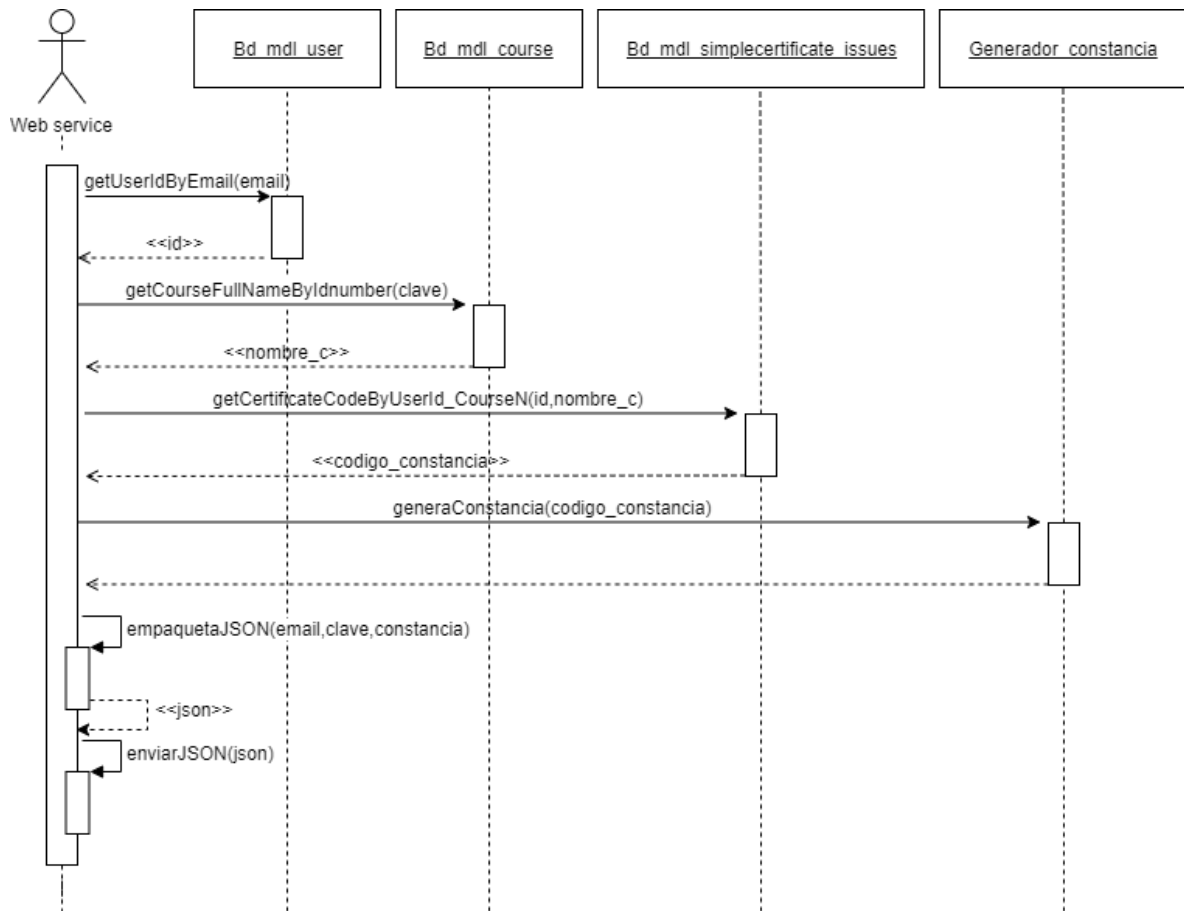


Figura 13. Diagrama de secuencia de servicio web B

Diagrama Entidad Relación para ambos casos de uso

En síntesis, se muestran las distintas relaciones y atributos de las entidades necesarias para resolver el caso de uso en general.

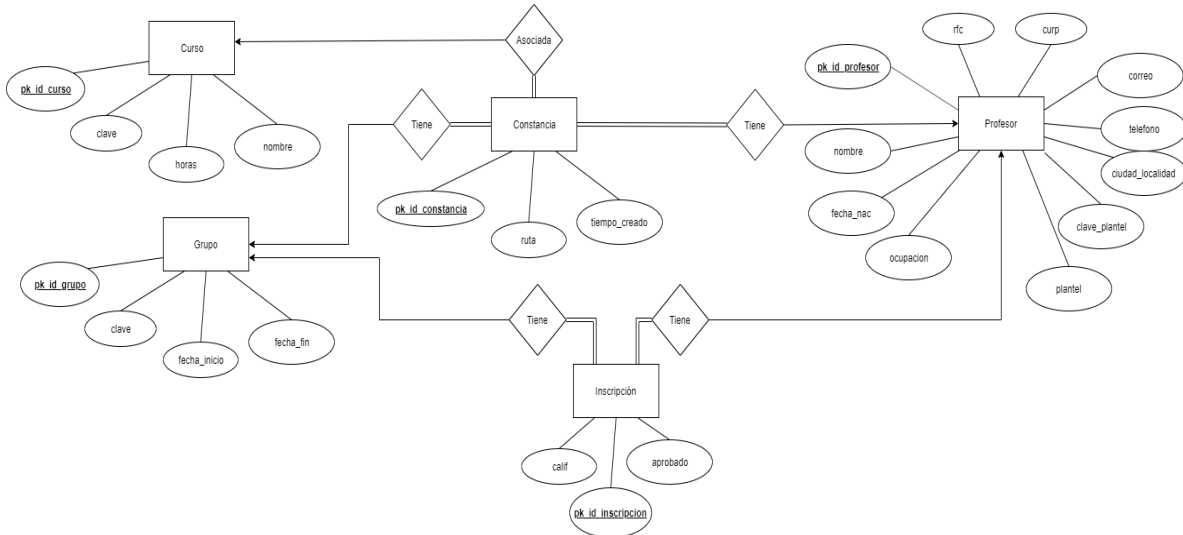


Figura 14. Diagrama E-R general

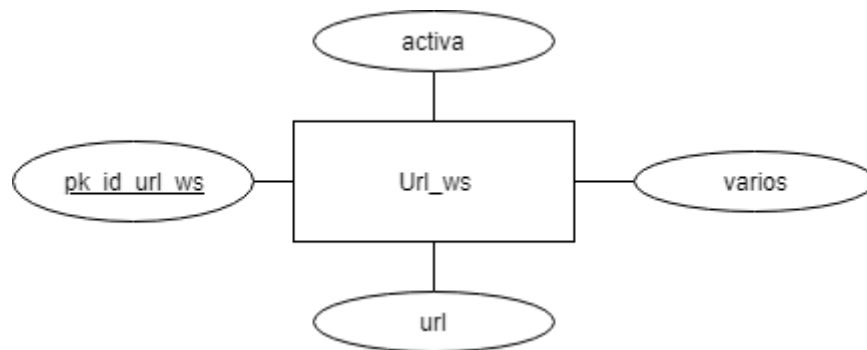


Figura 15. Diagrama E-R para entidad Url_ws

3.4 Construcción

A continuación, se muestra una tabla con las herramientas a utilizar para la implementación de la solución, así como las clases asociadas a la solución de los casos de uso.

Especificación del ambiente de desarrollo

Concepto	Herramienta	Versión
Framework	Spring Framework	5.0
Diagramador UML	draw.io	N/A
Manejador de bases de datos	PostgreSQL	10
Servidor de aplicaciones	Apache Tomcat	8.5.39
IDE	Spring Tools Suite	4.0
Control de versiones	Git	2.11

Tabla 8. Ambiente de desarrollo

Lista de casos de uso construidos

Adelante se muestran los casos de usos desprendidos del caso de uso Consultar/Extraer constancias con los documentos con el código fuente generados, así como sus respectivas versiones.

- Consultar/Extraer constancias masivamente

Constancia.java (versión 1.3)

ConstanciaMasivoControlador.java (versión 1.1)

wconstancias.php (versión 1.6)

SeguridadPDF.java (versión 1.1)

- Consultar/Extraer constancia bajo demanda

Constancia.java (versión 1.3)

ClienteConstanciaControlador.java (versión 1.5)

wconstancia.php (versión 1.4)

SeguridadPDF.java (versión 1.1)

3.5 Pruebas

Enseguida se muestran las tablas de casos de prueba por caso de uso.

Consultar/Extraer constancias masivamente

- Prueba 1

Entradas	Resultado esperado
<p>Teniendo datos válidos en todas las tablas, con la tabla <i>Url_ws vacía</i>, ejecutar la siguiente consulta en la base de datos:</p> <pre>INSERT INTO public.url_ws (activa, url, varios) VALUES (true, 'http://unamedu.serveftp.com:8585/mod/simplecertificate/wsconstancias.php', true).</pre> <p>Ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i>, que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad.</p> <p>Correr la aplicación.</p>	<p>En la carpeta definida en el archivo <i>application-cert.properties</i> en el campo <i>ws.ruta_local</i> buscar carpetas que contengan archivos pdf's los cuales no son visualizables en una aplicación para leer esta clase de archivos si no se tiene la contraseña del archivo.</p>

Tabla 9. Prueba 1-A

- Prueba 2

Entradas	Resultado esperado
<p>Teniendo datos válidos en todas las tablas, excepto en la tabla profesor en la columna <i>correo</i>, o en la tabla Grupo en la columna <i>clave</i>, o en la tabla Curso en la columna <i>clave</i> y con la tabla <i>Url_ws vacía</i>, ejecutar la siguiente consulta en la base de datos:</p> <pre>INSERT INTO public.url_ws (activa, url, varios) VALUES (true, 'http://unamedu.serveftp.com:8585/mod/simplecertificate/wsconstancias.php', true).</pre> <p>Ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i>, que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad.</p>	<p>En la carpeta definida en el archivo <i>application-cert.properties</i> en el campo <i>ws.ruta_local</i> no debería haber archivos y carpetas nuevas.</p>

Correr la aplicación.	
-----------------------	--

Tabla 10. Prueba 2-A

- Prueba 3

Entradas	Resultado esperado
<p>Teniendo vacías todas las tablas, con la tabla <i>Url_ws</i> vacía ejecutar la siguiente consulta en la base de datos:</p> <pre>INSERT INTO public.url_ws (activa, url, varios) VALUES (true, 'http://unamedu.serveftp.com:8585/mod/simplecertificate/wsconstancias.php', true).</pre> <p>Ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i>, que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad.</p> <p>Correr la aplicación.</p>	<p>En la carpeta definida en el archivo <i>application-cert.properties</i> en el campo <i>ws.ruta_local</i> no debería haber archivos o carpetas nuevas, ni se debería generar alguna excepción .</p>

Tabla 11. Prueba 3-A

- Prueba 4

Entradas	Resultado esperado
<p>Teniendo vacías todas las tablas, ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i>, que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad.</p> <p>Correr la aplicación.</p>	<p>No sucede nada, ni generarse excepción alguna.</p>

Tabla 12. Prueba 4-A

- Prueba 5

Entradas	Resultado esperado
<p>Con tabla <i>Url_ws</i> vacía ejecutar la siguiente consulta en la base de datos:</p> <pre>INSERT INTO public.url_ws (activa, url, varios) VALUES (false, 'http://unamedu.serveftp.com:8585/mod/simplecertificate/wsconstancias.php', true).</pre> <p>Ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i>, que es una expresión cron del</p>	<p>No debería suceder nada, ni generarse excepción alguna.</p>

estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad. Correr la aplicación.	
--	--

Tabla 13. Prueba 5-A

- Prueba 6

Entradas	Resultado esperado
Teniendo todas las tablas vacías, ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i> , que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad. Correr la aplicación.	No debería suceder nada, ni generarse excepción alguna.

Tabla 14. Prueba 6-A

- Prueba 7

Entradas	Resultado esperado
Con tabla <i>Url_ws</i> vacía ejecutar la siguiente consulta en la base de datos: <i>INSERT INTO public.url_ws (activa, url, varios)</i> <i>VALUES (true, 'http://algunacosa.com:8585/mod/simplecertificate/wsconstancias.php', true).</i> Ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i> , que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad. Correr la aplicación.	No debería suceder nada, ni generarse excepción alguna.

Tabla 15. Prueba 7-A

Consultar/Extraer constancia bajo demanda

Realizar previamente un formulario que reciba clave de grupo, clave de curso y correo del profesor del lado del SICECD para realizar las pruebas (se presupone una base de datos no vacía en el SICECD).

- Prueba 1

Entradas	Resultado esperado
<p>Con tabla <i>Url_ws</i> vacía ejecutar la siguiente consulta en la base de datos:</p> <pre>INSERT INTO public.url_ws (activa, url, varios) VALUES (true, 'http://unamedu.serveftp.com:8585/ mod/simplecertificate/wsconstancia.php', false).</pre> <p>Identificar en la base de datos a algún profesor de un curso y grupo específico con constancia asociada y eliminar el archivo y entrada en la tabla <i>Url_ws</i>. Luego, introducir en el formulario los datos correspondientes.</p>	<p>En la carpeta definida en el archivo <i>application-cert.properties</i> en el campo <i>ws.ruta_local</i> debería hallarse un archivo nuevo.</p>

Tabla 16. Prueba 1-B

- Prueba 2

Entradas	Resultado esperado
<p>Con tabla <i>Url_ws</i> vacía ejecutar la siguiente consulta en la base de datos:</p> <pre>INSERT INTO public.url_ws (false, url, varios) VALUES (false, 'http://unamedu.serveftp.com:8585/ mod/simplecertificate/wsconstancia.php', false).</pre> <p>Identificar en la base de datos a algún profesor de un curso y grupo específico con constancia asociada y eliminar el archivo y entrada en la tabla <i>Url_ws</i>.</p> <p>Luego, introducir en el formulario los datos correspondientes.</p>	<p>No debería suceder nada, ni generarse excepción alguna.</p>

--	--

Tabla 17. Prueba 2-B

- Prueba 3

Entradas	Resultado esperado
<p>Con tabla <i>Url_ws</i> vacía ejecutar la siguiente consulta en la base de datos: <i>INSERT INTO public.url_ws (activa, url, varios)</i> <i>VALUES (true, 'http://algunacosa.com:8585/</i> <i>mod/simplecertificate/wsconstancia.php', false).</i> Ajustar en el archivo <i>application-cert.properties</i> el valor de <i>ws.scheduleTransfer</i>, que es una expresión cron del estilo (<i>0 a b * * ?</i>), donde “a” es sustituido por un minuto y “b” por una hora en formato de 24 horas, esta expresión es la hora de ejecución de la funcionalidad. Correr la aplicación.</p>	<p>No debería suceder nada, ni generarse excepción alguna.</p>

Tabla 18. Prueba 3-B

Conclusiones

El objetivo general del servicio social pretendió que los alumnos tuvieran una primera experiencia en un trabajo profesional de desarrollo aplicando los conocimientos adquiridos en la licenciatura en Ciencias de la Computación con especial énfasis en el diseño e implementación de un sistema como lo marca el plan de estudios en materias tales como Modelado y Programación, Ingeniería de Software, entre otras.

El objetivo específico del trabajo fue desarrollar el caso de uso Consultar/Extraer constancias perteneciente al proyecto de servicio social del SICECD.

Construí un software para solucionar el problema, el cual consistió en un servicio web que en su interior se subdivide en dos servicios que actúan en estos momentos: cuando se solicita una constancia bajo demanda, y cuando se requiere traer más de una cada cierto tiempo; este servicio web, por simplicidad, actúa bajo la arquitectura REST, lo cual permite trabajar con la infraestructura otorgada por el protocolo HTTP que, por supuesto, utiliza Moodle.

Por otra parte, por fortuna, el caso de uso contempla la programación en un sistema ya existente como Moodle, así, pude mejorar mi habilidad para entender el funcionamiento de código ajeno y escrito con las particularidades de un idioma que no es el español, lo que constituyó, a mi juicio, la mejor experiencia.

Trabajar en un ambiente fuera de la zona de confort de un estudiante me impulsó a mejorar diversos aspectos en mis habilidades de programación al poder interactuar en distintos puntos complementarios al de la sola implementación.

Finalmente, el servicio social me brindó una experiencia edificante, ya que me llevó a tomar decisiones que repercutieron en el desarrollo de los otros compañeros de equipo. Además, la prueba del sistema resultó muy interesante porque en este punto se separó la teoría de la práctica, con lo que constaté que realizar un programa que procesa muchos recursos requiere de una amplia experiencia para que las funcionalidades cubran la mayoría de los casos de prueba en forma óptima.

Bibliografía

- [1] The Apache Software Foundation. (2019). *Apache Tomcat*. 7 de Marzo 2020, de The Apache Software Foundation Sitio web: <http://tomcat.apache.org/>
- [2] The PHP Group. (2019). *¿Qué es PHP?*. 7 de Marzo 2020, de The PHP Group Sitio web: <https://www.php.net/manual/es/intro-whatism.php>
- [3] The PostgreSQL Global Development Group. (2019). *What is PostgreSQL?*. Octubre 2019, de The PostgreSQL Global Development Group Sitio web: <https://www.postgresql.org/about/>
- [4] Oracle. (2019). *¿Qué es la tecnología Java y para qué la necesito?*. 7 de Marzo 2020, de Oracle Sitio web: https://www.java.com/es/download/faq/whatis_java.xml
- [5] López, A. (2013). *Introducción al desarrollo de programación con JAVA (Tercer edición)*. México: Las prensas de ciencias.
- [6] López L. (2018). *¿Qué es Spring framework?*. 7 de Marzo 2020, de OpenWebinars Sitio web: <https://openwebinars.net/blog/que-es-spring-framework/>
- [7] Spring. (2020). *Spring Boot: Overview*. 7 de Marzo 2020, de VMware Sitio web: <https://spring.io/projects/spring-boot>
- [8] Peterson, L. & Davie, B. (2012). *Applications*. En *Computer Networks: a system approach (5th edition)* (pp.718-727). Estados Unidos: Morgan Kaufman.
- [9] Bray, T., Ed. (2017). *The JavaScript Object Notation (JSON) Data Interchange Format*. 7 de Marzo 2020, de Internet Engineering Task Force Sitio web: <https://tools.ietf.org/html/rfc8259>
- [10] Ecma International. (2017). *Standard ECMA-404: The JSON Data Interchange Syntax*. 7 de Marzo 2020, de Ecma International Sitio web: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>
- [11] Comunidad Moodle. (2019). *Acerca de Moodle*. 7 de Marzo 2020, de Moodle Sitio web: https://docs.moodle.org/all/es/Acerca_de_Moodle
- [12] Pressman R. (2010). *Ingeniería del Software: un enfoque práctico (séptima edición)*. México: McGraw-Hill Educación.