



FACULTAD 2

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE
INGENIERO EN CIENCIAS INFORMÁTICAS**

**Implementación del módulo de información estadístico-
bibliométrico para el sistema de Representación de
Bibliografía Científica**

AUTORES: Betsy Dorta Cárdenas

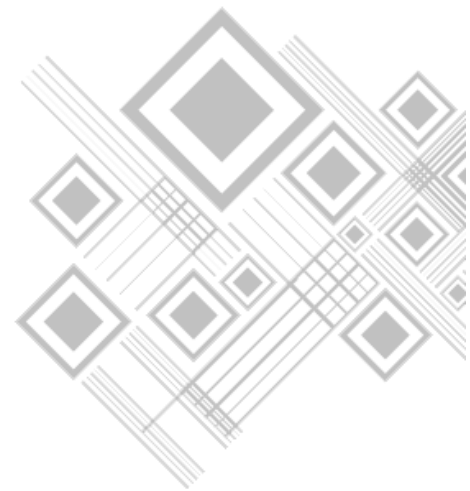
Laura Llanes Zayas-Bazán

TUTORES: Ing. Vladimir Milián Núñez

Ing. Roberto Antonio Infante Milanés

La Habana, 27 de junio de 2017

“Año 59 de la Revolución”





PENSAMIENTO

“Existen derrotas, pero nadie está a salvo de ellas. Por eso es mejor perder algunos combates en la lucha por nuestros sueños que ser derrotados sin siquiera saber por qué se está luchando”



Paulo Coelho



DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmamos la presente a los ____ días del mes de _____ del _____.

Betsy Dorta Cárdenas
Firma del autor

Laura Llanes Zayas-Bazan
Firma del autor

Ing. Vladimir Milián Núñez
Firma del tutor

Ing. Roberto Antonio Infante Milanés
Firma del tutor



DATOS DE CONTACTOS

Datos del Autor:

Betsy Dorta Cárdenas.

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: bdorta@estudiantes.uci.cu

Datos del Autor:

Laura Llanes Zayas – Bazan.

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: llanes@estudiantes.uci.cu

Datos del Tutor:

Ing. Vladimir Milián Núñez: graduado en el año 2008 en Ingeniería en Ciencias Informáticas con Título de Oro en la Universidad de las Ciencias Informáticas. Ha participado en varios eventos como son Informática, Evento Nacional de Seguridad Informática, Jornada Científica del ICIMAF, UCIENCIA, entre otros. Sus áreas de interés son relacionadas con el aprendizaje automático y el análisis de datos.

Datos del Tutor:

Ing. Roberto Antonio Infante Milanés: graduado en el año 2015 en Ingeniería en Ciencias Informáticas en la Universidad de Granma. Profesor Instructor, ha participado en los eventos CIVITEC 2015 y 2016, Informática 2016 y en la Peña Tecnológica del MININT 2016, entre otros.



DEDICATORIA

A mi mamita linda.

A mi super papá.

A mi bella hermana.

A todos los presentes, los que creyeron, no me abandonaron y confiaron en mí. Muchas gracias.

Laura y Betsy



RESUMEN

En la actualidad, la búsqueda de bibliografía científica está determinada por el análisis y evaluación de las publicaciones indexadas en bases de datos. A pesar de esto, la mayoría de los principales motores de búsqueda y fuentes bibliográficas, no cuenta con mecanismos que les permitan a los usuarios tomar decisiones teniendo en cuenta las áreas, autores, instituciones o publicaciones más relevantes.

Durante el curso 2014-2015 se desarrolló en la Universidad de las Ciencias Informáticas el Sistema de Representación de Bibliografía Científica. Este sistema permite la búsqueda de publicaciones científicas, a partir de un criterio especificado por el usuario. Sin embargo, no permite calcular una serie de indicadores estadísticos-bibliométricos que le permiten al usuario conocer que solución es más afín. En el presente trabajo se describe el desarrollo de un módulo que permite trabajar con algunos de los indicadores más utilizados a nivel mundial, tales como: productividad, productividad de los autores o Ley de Lotka, visibilidad o impacto, índice h, índice g e índice i10.

La implementación de la solución propuesta se realizó a partir del uso del lenguaje de programación del lado del servidor Python, la utilización del marco de desarrollo Django y PostgreSQL como sistema gestor de base de datos. La solución fue validada mediante la elaboración de un conjunto de casos de pruebas de caja blanca y pruebas de caja negra, asociadas a la metodología seleccionada para guiar el proceso de desarrollo.

Palabras clave: bibliografía científica, indicadores estadísticos bibliométricos.



ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	5
1.1 Introducción.....	5
1.2 Conceptos fundamentales	5
1.3 Análisis de sistemas existentes que brindan información estadística de las publicaciones científicas indexadas	14
1.3.1 CiteSeerX.....	14
1.3.2 dblp.....	15
1.3.3 Google Académico	15
1.3.4 Microsoft Academic Search	15
1.3.5 Resultados del estudio realizado	16
1.4 Ambiente de desarrollo	17
1.4.1 Metodología de desarrollo	17
1.4.2 Lenguajes.....	18
1.4.3 Herramientas y tecnologías	20
1.5 Conclusiones parciales del capítulo	22
CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA.....	24
2.1 Introducción	24
2.2 Propuesta de solución	24
2.3 Modelo Conceptual.....	25
2.4 Requisitos del sistema.....	26
2.4.1 Requisitos funcionales del sistema	26



2.4.2	Requisitos no funcionales	27
2.5	Definición de los casos de uso.....	29
2.5.1	Actores del sistema	29
2.5.2	Diagrama de casos de uso del sistema	29
2.5.3	Descripción de los casos de uso del sistema.....	30
2.6	Etapa de diseño.....	32
2.6.1	Diseño Arquitectónico.....	32
2.6.2	Patrones de diseño.....	35
2.7	Modelo físico de los datos	38
2.8	Diagrama de paquetes.....	39
2.9	Diagrama de clases del diseño utilizando estereotipos web.....	40
2.10	Conclusiones parciales del capítulo	44
CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBA.....		46
3.1	Introducción	46
3.2	Etapa de implementación	46
3.2.1	Diagrama de componentes.....	46
3.3	Estándares de codificación empleados.....	47
3.4	Diagrama de despliegue	47
3.4.1	Descripción de los protocolos de comunicación.....	48
3.5	Pruebas de Software	49
3.5.1	Estrategia de prueba seguida	49
3.5.2	Pruebas de caja blanca	49
3.5.3	Pruebas de caja negra.....	50



3.5.4	Pruebas de sistema	53
3.5.5	Pruebas de despliegue	54
3.6	Resultados de las pruebas	54
3.7	Conclusiones parciales del capítulo	55
CONCLUSIONES		56
RECOMENDACIONES		57
GLOSARIO		58
REFERENCIAS BIBLIOGRÁFICAS		59
ANEXOS.....		61
Anexo I. Preguntas formuladas al cliente durante la entrevista no estructurada		61
Anexo II. Descripción de los casos de uso del sistema.....		62
Anexo III. Diagrama de clases del diseño utilizando estereotipos web		69



ÍNDICE DE FIGURAS

Figura 1. Resultados gráficos de la Ley de Lotka.....	12
Figura 2. Representación logarítmica lineal de la Ley de Lotka.....	13
Figura 3. Propuesta de solución.....	25
Figura 4. Modelo Conceptual	26
Figura 5. Diagrama de caso de uso del sistema RBC.	30
Figura 6. Arquitectura Cliente-Servidor.	33
Figura 7. Funcionamiento del MTV en Django.	35
Figura 8 Ejemplo de la clase Document.....	36
Figura 9 Ejemplo de la clase Org_unit.	36
Figura 10 Ejemplo de la clase Publication.....	37
Figura 11 Formulario UserCreationForm para validar un nuevo usuario.	37
Figura 12. Modelo físico de los datos.....	39
Figura 13 Diagrama de paquetes del sistema RBC.....	40
Figura 14. Diagrama de clases del diseño utilizando estereotipos web, Mostrar indicador de productividad.	41
Figura 15. Diagrama de clases del diseño utilizando estereotipos web, Calcular la productividad de los autores (Ley de Lotka).	42
Figura 16. Diagrama de clases del diseño utilizando estereotipos web, Calcular visibilidad o impacto.	43
Figura 17. Diagrama de clases del diseño utilizando estereotipos web, Mostrar índice h.....	44
Figura 18. Diagrama de componente	46
Figura 19. Diagrama de despliegue	48
Figura 20 Código de test.....	50
Figura 21. Resultado de la Prueba unitaria.	50



Figura 22. Gráfico correspondiente al caso de prueba del RF3 Calcular Visibilidad o Impacto.	53
Figura 23. Prueba de carga	54
Figura 24. Diagrama de clases del diseño utilizando estereotipos web, Mostrar índice g.....	69
Figura 25. Diagrama de clases del diseño utilizando estereotipos web, Mostrar índice i10.....	70



ÍNDICE DE TABLA

Tabla 1. Cálculo del índice h.....	11
Tabla 2. Cálculo del índice g.....	11
Tabla 3. Ley de Lotka.	13
Tabla 4. Comparación de Sistemas existentes.	16
Tabla 5. Requisitos funcionales del sistema RBC.	27
Tabla 6. Definición de los actores.	29
Tabla 7. Descripción del CU: Calcular la productividad de los autores (Ley de Lotka).	30
Tabla 8. Caso de prueba de partición equivalente del RF2 Calcular la productividad de los autores (Ley de Lotka).....	51
Tabla 9. Pruebas de Despliegue	54
Tabla 10. Resultado de las pruebas.....	54
Tabla 11. Descripción del CU: Mostrar índice de productividad.	62
Tabla 12. Descripción del CU: Calcular visibilidad o impacto.	63
Tabla 13. Descripción del CU: Mostrar índice h.	65
Tabla 14. Descripción del CU: Mostrar índice g.	66
Tabla 15. Descripción del CU: Mostrar índice i10.....	68
Tabla 16. Caso de prueba de partición equivalente del RF1 Mostrar índice de productividad.	71
Tabla 17. Caso de prueba de partición equivalente del RF4,5,6 Mostrar índices h, Mostrar índice g, Mostrar índice i10.	72



INTRODUCCIÓN

A nivel mundial, cada vez más la comunicación científica se extiende a gran escala y han sido creadas plataformas específicas para la interacción e intercambio de información. Para facilitar el análisis y evaluación de las investigaciones científicas, se desarrolla un estudio de las publicaciones indexadas en bases de datos. Este análisis permite obtener una medida de la relevancia de los resultados científicos, el conocimiento, impacto y beneficios generados.

Por ello, es necesario contar con herramientas que ayuden a los investigadores a determinar cuál o cuáles son las publicaciones y/o autores de mayor impacto en un área determinada de la investigación; cuál es el autor, grupo o institución que más publicaciones realiza sobre un tema determinado; el número de artículos realizados por un autor, grupo, institución o país; el número de artículos publicados por las revistas sobre un tema determinado en un período de tiempo determinado; la cantidad de artículos publicados por las principales áreas de investigación de una disciplina; el número de relaciones establecidas entre diferentes autores, grupos, instituciones o países para la publicación de artículos, entre otros.

Este conjunto de información puede ser de gran ayuda para cualquier investigador, aunque fundamentalmente para los investigadores más jóvenes, ya que son una valiosa fuente para decidir en qué área de la ciencia investigar, qué autores y publicaciones estudiar, así como qué autores, grupos o instituciones pueden ser posibles colaboradores.

Los artículos científicos son síntesis de informes, que orientan a los lectores interesados en consultar el trabajo original. Se encargan de difundir los resultados de una investigación realizada sobre un área determinada del conocimiento. Deben ser claros y precisos para lograr comunicar su objetivo, también deben estar referenciados adecuadamente para poder justificar y verificar los datos contenidos en el trabajo. Algunos sistemas que se encargan de la publicación de estos artículos a nivel mundial son Google Académico, que es el más conocido; el sistema dblp, CiteSeerX y Microsoft Academic Search.

En la Universidad de las Ciencias Informáticas (UCI), a partir del curso 2014-2015, se ha estado desarrollando el sistema web de Representación de Bibliografía Científica (RBC). Esta herramienta le brinda al usuario la posibilidad de introducir un criterio de búsqueda por palabras claves y obtener



publicaciones científicas en el área de las Ciencias Básicas y la Cibernética, que coincidan con el criterio introducido. Asimismo, permite obtener resultados más específicos al posibilitar realizar búsquedas mediante un documento base, es decir, se sube un documento al sistema y este devuelve las publicaciones que existen en la base de datos similares al documento introducido como base para la búsqueda. Además de lo planteado, RBC tiene como objetivo, ser de ayuda en la toma de decisiones en el ámbito científico, por lo que es necesario que brinde o permita obtener otro cúmulo de información sobre los artículos, autores, áreas de la ciencia o grupos.

En la actualidad, el sistema RBC permite para los autores, ver la cantidad de artículos publicados por año, la cantidad de citas recibidas y la relación de coautoría; para el caso de los artículos permite ver la cantidad de citas recibidas por año, y los artículos donde son citados. No obstante, el sistema no permite obtener una serie de información como son la ley Lotka, índices de productividad, índices de posicionamiento (g, h, i10), indicadores estos de suma importancia a la hora de decidir qué autor o autores estudiar primero, sobre qué área de la ciencia apostar, en qué revistas publicar, entre otros.

Por lo anterior se define como **problema a resolver**: la incapacidad del sistema RBC para mostrar información estadística de las bibliografías científicas, afectando negativamente la toma de decisiones a la hora de elegir las más relevantes.

El **objeto de estudio**, se enmarca en los sistemas informáticos, para centrar la investigación se tiene como **campo de acción** los indicadores estadísticos-bibliométricos.

Para solucionar el problema planteado se define como **objetivo general** del presente trabajo de diploma: desarrollar un módulo para el sistema RBC que permita la obtención de información estadística sobre indicadores estadísticos-bibliométricos, que contribuya a la toma de decisiones en el ámbito científico.

Con vista al cumplimiento del objetivo general se proponen las siguientes **tareas de la investigación**:

- Análisis de bibliografías para conocer los principales conceptos que serán tratados durante el desarrollo del documento.
- Selección de las tecnologías, metodologías y herramientas más factibles para el desarrollo del sistema.



- Análisis de los sistemas informáticos que utilizan indicadores estadísticos-bibliométricos, existentes a nivel internacional, estableciendo similitudes con la investigación en curso.
- Desarrollo del módulo de información estadístico-bibliométrico para el sistema RBC, en correspondencia con la metodología seleccionada.
- Ejecución de pruebas de software a las funcionalidades implementadas a la aplicación, para comprobar el correcto funcionamiento de la misma.

Para la realización de la actual investigación se emplearon varios métodos científicos de investigación, los cuales fueron:

Los métodos de investigación empíricos: permiten al investigador hacer una serie de investigaciones referentes a su problemática, retomando experiencia de otros autores, lo que permite tener una base para la posterior investigación.

- **Entrevista:** se empleó para conocer las necesidades del cliente y definir los requisitos y características de la solución propuesta.
- **Modelación:** utilizado para modelar la arquitectura, crear los artefactos, diagramas y modelos a utilizar en el desarrollo del módulo de información estadístico-bibliométrico para el sistema RBC.

Los métodos de investigación teóricos: permiten revelar las relaciones esenciales del objeto de investigación, son fundamentales para la comprensión de los hechos.

- **Histórico-Lógico:** este método ayuda a dar cumplimiento al estudio del estado del arte. Se usa con el objetivo de estudiar los principales conceptos sobre la Informetría, más específicamente la bibliometría; y así adquirir mayor conocimiento sobre el tema.
- **Analítico-Sintético:** permite realizar un análisis del problema, primero en partes pequeñas, analizando la bibliometría y la cienciometría de forma separada; y después como un todo integrado.

Para mejorar la comprensión el documento se encuentra estructurado en 3 capítulos.



Capítulo 1. Fundamentación teórica: se realiza un estudio del estado del arte, tratando así los conceptos de informetría, bibliometría, cienciometría y sistema de recomendación bibliográfica. Además, del análisis de sistemas existentes con un funcionamiento similar al de RBC. Se describe la metodología de desarrollo, las herramientas, lenguajes y las tecnologías que se van a utilizar para la propuesta de solución del problema en cuestión.

Capítulo 2. Características del Sistema: se describen las características de la solución que se propone como resultado de la presente investigación. Se aborda todo lo referente al análisis y diseño del módulo de información estadístico-bibliométrico para el sistema RBC. Además, se elaboran de las descripciones de los requisitos, diagramas de casos de uso del sistema y los prototipos de interfaz. Se define la arquitectura utilizada y los patrones de diseño, los diagramas correspondientes, como son los de clases empleando estereotipos web, el de paquete y los de colaboración; y el rediseño de la base de datos.

Capítulo 3. Implementación y Prueba: describe todo lo referente a la implementación del sistema. Se representan los diagramas de componente y de despliegue. Además, se definen los estándares de codificación empleados para garantizar la fácil comprensión del sistema y las pruebas realizadas para comprobar su correcto funcionamiento, cumpliendo con las características requeridas por el cliente.



CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se presentan los elementos teóricos que sirven de base a la investigación del problema planteado. Se realiza un estudio del estado del arte, mediante un análisis de sistemas de software existentes que brindan información estadística bibliométrica de las publicaciones científicas indexadas. Además, se realiza una breve descripción de la metodología, lenguajes, herramientas y tecnologías a utilizar para dar solución al problema planteado.

1.2 Conceptos fundamentales

En este acápite se analizan las principales definiciones asociadas al dominio del problema y que son necesarios para el desarrollo de la investigación. De acuerdo con esto, se ofrece un enfoque de los aspectos fundamentales relacionados con cada uno de ellos, partiendo del concepto de sistema de recomendación bibliográfica.

- **Sistema de recomendación bibliográfica:** son herramientas que “...ayudan al usuario a escoger elementos de una gran cantidad de opciones. El volumen de la información es la razón principal del surgimiento de estos sistemas”. (Cañar, y otros, 2009)

Los sistemas de recomendación son “...toda técnica de deducción de la información ofrecida por un usuario a unos temas concretos y conocidos por el sistema. Posteriormente el sistema compara la información deducida del usuario con otra información deducida de la misma forma de otras fuentes y ofrece al usuario una ponderación de qué fuente es la más afín al usuario”. (Torres, 2014)

Resumiendo lo antes expuesto, se puede decir que un sistema de recomendación no es más que una herramienta que facilita la búsqueda de información a partir de las particularidades expresadas por los usuarios.

- **Informetría:** diferentes autores han dado su definición de informetría. Algunos de los más citados son los siguientes:



“... la informetría es un sub-campo emergente de la Ciencia de la Información, y está basada en la combinación de técnicas avanzadas de recuperación de la información con estudios cuantitativos de los flujos de información”. (Wormell, 1998)

Rajan ha apuntado los objetivos de la misma: “La Informetría formaliza y consolida muchos estudios de medición referidos a la productividad en Información; intenta enfocar el concepto de la organizada complejidad de la Sociedad de la Información (sic)...”. (Rajan, 1985)

“...La informetría comprende la medición de los aspectos de la información, el almacenamiento y su recuperación, por lo que contiene la teoría matemática y la modelización...”. (Spinak, 1996)

Tomando como referente las definiciones citadas, se concluye que la informetría está dirigida al estudio de la información, tomando como base los aspectos cuantitativos de la misma, que incluye otros de carácter comunicativo. Se sustenta en el desarrollo de investigaciones bibliométricas y cienciométricas.

El área de la informetría se desarrolla en un campo muy extenso, los bibliómetras y cienciométristas centralizaron sus estudios en las mediciones y los modelos matemáticos de un conjunto de áreas bien específicas que incluyen:

- La evolución de las literaturas temáticas y el desarrollo de las bases de datos en bibliotecas.
 - Particularidades de las publicaciones fuentes, específicamente la distribución de artículos por pautas en las diferentes revistas.
 - Las características de los distintos autores, enmarcando la productividad por la cantidad de documentos y el grado de colaboración existente en ellos.
 - El estudio de las citaciones, en particular la distribución de las instituciones, países, autores, etc.
 - El uso de la información en bibliotecas, bases de datos, revistas.
- **Cienciometría:** las primeras definiciones consideraban la cienciometría como “la medición del proceso informático”, donde el término informático significaba, a diferencia de hoy día con una connotación de computadoras, “la disciplina científica que estudia la estructura y las propiedades de la información científica y las leyes del proceso de comunicación”. (Mikhilov, 1984)



Haciendo referencia a la cienciometría, Derek J. de Solla Price expresó que el principal objetivo de la misma es conseguir un análisis matemático de la Ciencia. Aunque reconocía que se necesitaba también la aportación de historiadores, sociólogos y psicólogos para aquellos tipos de análisis que no se podían expresar en términos métricos. Además, plantea que la actividad científica *“es peculiarmente medible y regular en sus conductas”*. (Price, 1961)

La cienciometría es un campo interdisciplinar ya que involucra a un gran número de disciplinas dada la amplitud de sus objetivos. Es una disciplina científica dedicada a los aspectos cuantitativos de la ciencia y la investigación científica, que sitúa su mayor énfasis en la evaluación científica y en los procedimientos e indicadores desarrollados para conseguirla. Aplica técnicas bibliométricas a la ciencia, sin embargo, no está limitada únicamente por este tipo de técnicas, pues también tanea el desarrollo y las políticas científicas. Algunos de los temas que investiga son:

- La dependencia existente entre la ciencia y la tecnología.
- La obsolescencia de las muestras científicas.
- El progreso cuantitativo de la ciencia.
- La creatividad y productividad de los científicos.
- El perfeccionamiento de las disciplinas y subdisciplinas, etc.

La cienciometría utiliza técnicas matemáticas y estudios estadísticos para analizar las peculiaridades de la investigación científica. Esta área se puede suponer como una herramienta de la sociología de la ciencia.

- **Bibliometría:** la bibliometría es *“el conjunto de conocimientos metodológicos para la aplicación de técnicas cuantitativas, destinadas al estudio de los procesos de producción, comunicación y uso de la información científica, con el objetivo de contribuir al análisis y evaluación de la Ciencia y la investigación”*. (Sainero, 2017)

Por otra parte, en 1969 se definió a la Bibliometría como *“la aplicación de los métodos matemáticos y estadísticos a los libros y otros medios de comunicación”*. (Pritchard, 1969)

B. C. Brookes en la Segunda Conferencia Internacional sobre Bibliometría, Cienciometría e Informetría de 1989, celebrada en Ontario, Canadá, reconoce a la Bibliometría *“como una*



disciplina limitada a la actividad bibliotecaria, aunque esta se debe enriquecer mediante las relaciones interdisciplinarias con estadística para refinar sus técnicas”. (Brookes, 1990)

La bibliometría estudia la distribución de los sectores tecnológicos y científicos apoyándose en las fuentes bibliográficas y evidentes para reconocer a los actores, sus relaciones y sus tendencias. Su alcance es multidisciplinario y estudia uno de los aspectos más relevantes y objetivos que es la comunicación impresa. Para puntualizar exactamente las ideas y los términos de la bibliometría, digamos que comprende:

- Estudio cuantitativo de la elaboración de documentos como se manifiesta en las bibliografías.
- La aplicación de análisis estadísticos para asimilar las características la creación de documentos y su uso.
- La aplicación de métodos estadísticos y matemáticos al estudio del uso que se forja de los libros y otros puntales dentro y entre los sistemas de bibliotecas.
- Estudio cuantitativo de las unidades físicas propagadas o de las unidades bibliográficas.

La bibliometría aporta los elementos cuantitativos precisos para la organización y dirección eficaz de las bibliotecas, así como para la estimación del efecto de su actividad en la sociedad. Aporta la información cuantitativa necesaria para el progreso de las actividades bibliotecarias y bibliográficas, lo que permite el análisis cualitativo integral de dicha actividad y la medición cuantitativa de las fuentes documentales o colecciones de las mismas.

- **Indicadores bibliométricos:** “...los estudios bibliométricos tienen por objeto el tratamiento y análisis cuantitativo de las publicaciones científicas. Forman parte de los «estudios sociales de la ciencia» y entre sus principales aplicaciones se encuentra el área de la política científica...” (Zulueta, 1999). Los indicadores bibliométricos permiten clasificar, manejar y analizar grandes volúmenes de publicaciones científicas. Hay una gran variedad de estos indicadores, a continuación, se describen los fundamentales.
 - **Indicadores personales:** estos indicadores se ocupan de características de los autores tales como la profesión, el sexo, la edad, el país, institución..., obteniendo la distribución porcentual en los trabajos de una publicación, una base de datos u otro conjunto de documentaciones.



- **Indicadores de producción:** los indicadores de producción se alcanzan por medio del recuento de las publicaciones científicas. Puede considerarse una medida de la actividad científica.

La productividad científica, se puede decir que es, la cantidad de publicaciones producidas por un autor, una institución, un departamento o grupo de investigación, una disciplina o un país durante un período de tiempo determinado. Lo que permite establecer rankings atendiendo al nivel de producción de la institución u otros elementos conseguidos por agregación que se comparen.

- **Indicadores de dispersión:** mediante ellos se conoce que publicaciones componen el núcleo de la disciplina. Para ello comúnmente se recurre a considerarlo aquellas que reservan el 50% de las citas.
- **Indicador de productividad:** se especifica como el logaritmo decimal del número de publicaciones.

A partir de los datos de productividad de los autores estudiados, pueden obtenerse indicadores como el autor que más publicaciones realiza sobre un tema determinado, el número de artículos realizados por un autor, grupo, institución o país, las publicaciones y/o autores de mayor impacto en un área determinada de la investigación, el número de artículos publicados por una revista en un período de tiempo determinado, cantidad de artículos comprendidos por las principales áreas de investigación de una disciplina, el número de relaciones establecidas entre diferentes autores, grupos, instituciones o países para la publicación de artículos.

- **Indicadores de visibilidad o impacto:** evalúan la influencia de los autores y de los trabajos publicados, son las magnitudes más destacadas de la bibliometría por el efecto inmediato que tienen sobre la profesión de los investigadores.

El más simple de estos indicadores es el del **número total de citas** recibidas, aunque en ocasiones se calcula su logaritmo, el cual es conocido como **índice de Platz**. El **promedio de citas** es otro de estos indicadores, el cual obtiene cada contribución de un autor. La importancia del número de citas recibidas depende del contexto en que se aplica.



El **factor de impacto** (FI) es la razón entre las citas recibidas y los artículos publicados en una revista, en el caso de una publicación, como una revista, el impacto real de esta depende también del número de artículos que origina durante el tiempo analizado. El FI se calcula para una revista en un año determinado.

Ejemplo FI 2003: para ello se utiliza la ecuación $\frac{A}{B}$ (1), donde:

A : número de veces en que los artículos publicadas en el período 2001-2002 han sido citados por las publicaciones a las que se les da seguimiento a lo largo del año 2003.

B : número de artículos publicados en esta revista en el período 2001-2002.

○ **Índices de posicionamiento:**

1. **índice h** o de Hirsch. Un investigador tiene un índice h , si todas sus publicaciones h reciben al menos h citas cada una, y el resto tiene h citas como máximo. La tabla 1 ilustra dos ejemplos de cálculo del *índice h* a partir de los 5 trabajos publicados por dos investigadores y ordenados de mayor a menor por el número de citas.

El investigador 1 tiene 3 trabajos con 3 o más citas, por lo que su índice h es 3, para llegar a 4 necesitaría tener 4 trabajos con al menos 4 citas. Por su parte, el investigador 2 tiene un índice h igual a 4 ya que tiene 4 trabajos con 4 citas o más, no tiene 5 trabajos con 5 citas o más.

Mediante el ejemplo también se puede observar uno de los desperfectos del *índice h*, el cual tiende a penalizar a los investigadores que priman la calidad frente a la cantidad y no publican un número de trabajos elevado, pero sus contribuciones terminan teniendo mucho impacto. En el caso del ejemplo, ambos investigadores han publicado la misma cantidad de trabajos, pero, aunque el investigador 1 recibe globalmente muchas más citas, al acumularlas solamente en dos trabajos, alcanza un *índice h* igual a 2, mientras que el investigador 2 con un impacto global mucho menor atendiendo al número total de citas, tiene un índice mayor ($h = 4$).



Tabla 1. Cálculo del índice h.

Investigador 1		Investigador 2	
Trabajos publicados ordenados por número de citas recibidas	Número de citas	Trabajos publicados ordenados por número de citas recibidas	Número de citas
1	610	1	8
2	290	2	7
3	2	3	5
4	1	4	4
5	1	5	3
Total trabajos = 5	Total trabajos = 904	Total trabajos = 5	Total trabajos = 27

2. **Índice g:** Un autor tiene un índice g cuando, considerando los g artículos más citados de dicho autor, la cantidad de citas acumuladas por estos g artículos es superior a g al cuadrado.

Tabla 2. Cálculo del índice g.

P	TC	P^2	$\sum TC$
1	4	1	4
2	3	4	7
3	2	9 ≤	9
4	2	16 >	11

En este caso, el índice g es 3, pues es la publicación que corresponde cuando se realiza una comparación entre P^2 y $\sum TC$, donde el último valor de $P^2 \leq \sum TC$.

3. **Índice i10:** El índice i10 corresponde a la cantidad de publicaciones de un autor que han sido citadas al menos 10 veces.
- **Indicadores de colaboración:** calculan las relaciones existentes entre los creadores científicos que han finalizado con la publicación conjunta de resultados. Puede contabilizarse la proporción de trabajos con dos, tres o más autores e identificarse aquellos sujetos o instituciones que más han publicado conjuntamente.



- **Indicadores de obsolescencia:** miden la degeneración de las publicaciones, entre los más utilizados se encuentra el **semiperíodo de Burton y Kebler** que se define como la mediana de las referencias ordenadas por su antigüedad. Otra medida es el **índice de Price** o porcentaje de referencias que tienen menos de 5 años de antigüedad.
- **Leyes bibliométricas:**
 - **Ley de la productividad de los autores:** expresa una relación cuantitativa entre los autores y las contribuciones realizadas en un campo dado a lo largo de un período de tiempo. La **Ley de Lotka** se expresa mediante la ecuación:
$$A(n) = \frac{K}{n^2} \quad (2)$$

K : es la máxima cantidad de publicaciones por un único autor.
 n : cantidad de publicaciones para las que se quiere conocer el número de autores.

Un ejemplo de esta Ley se muestra en la tabla 2, junto a una representación gráfica ilustrada en las figuras 1 y 2.

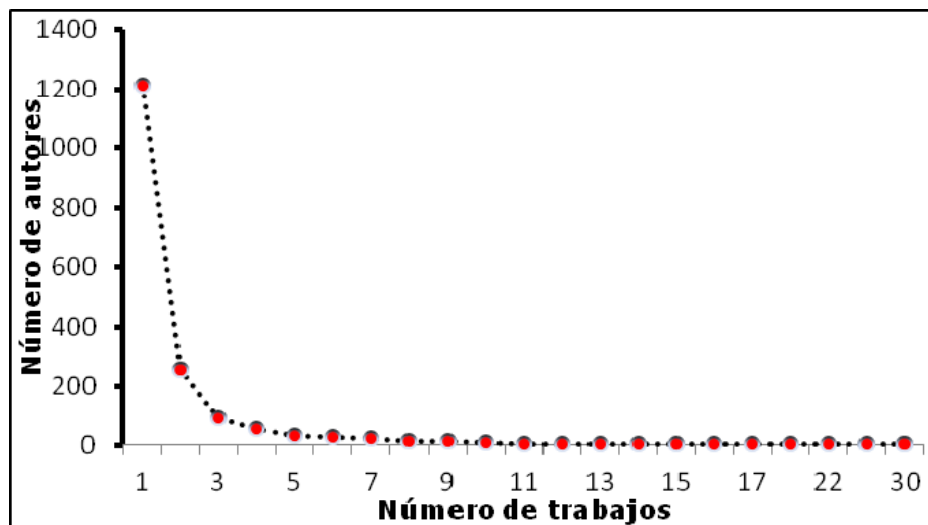


Figura 1. Resultados gráficos de la Ley de Lotka.

Tomado de: (Ardanuy, 2012)

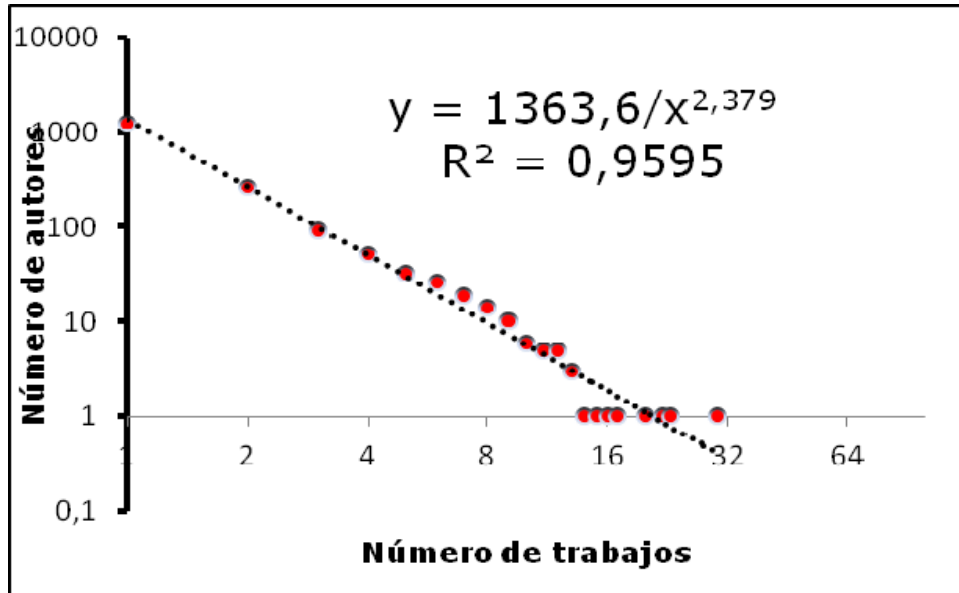


Figura 2. Representación logarítmica lineal de la Ley de Lotka.

Tomado de: (Ardanuy, 2012)

La tabla 2 representa los resultados gráficamente, no obstante, se observa que, debido a su rápida decadencia, no es demasiado útil por lo que se suele acudir a una representación logarítmica (figura 2) en que el gráfico se vuelve lineal aproximadamente.

Tabla 3. Ley de Lotka.

Número de autores	Número de contribuciones
1214	1
258	2
92	3
52	4
32	5
26	6
19	7
14	8
10	9



6	10
5	11
5	12
3	13
1	14
1	15
1	16
1	17
1	20
1	22
1	23
1	30
Total de autores 1744	Total de contribuciones publicadas 3236

1.3 Análisis de sistemas existentes que brindan información estadística de las publicaciones científicas indexadas

Con el objetivo de adquirir conocimiento para el presente trabajo se seleccionaron los siguientes sistemas como objeto de estudio, los cuales representan soluciones afines para la implementación del nuevo módulo que se le añadió al sistema RBC, se utilizó el cálculo de indicadores estadísticos bibliométricos en publicaciones científicas indexadas. Para ello es necesario conocer si los sistemas siguientes utilizan el mismo lenguaje de programación, si son de código abierto, si permiten el cálculo de algún indicador bibliométrico y sobre todo si se puede integrar con el sistema RBC.

1.3.1 CiteSeerX

El sistema CiteSeerx¹ muestra las publicaciones que contiene la base de datos; el total de publicaciones por autor; con el número de citas de cada una, además del año de publicación de cada artículo. Contiene un buscador que permite filtrar las búsquedas y proporciona la dirección donde se encuentran los documentos indexados, así como brinda la posibilidad de descargarlos directamente si está disponible.

¹ Accesible a través de la página <http://citeseerx.ist.psu.edu/index>



El sistema indexa más de 10 millones de artículos científicos. Está programado en lenguaje java y su código fuente se encuentra disponible en GitHub.

1.3.2 dblp

El sistema dblp² está centrado fundamentalmente en indexar publicaciones relacionadas con el campo de la ciencia de la computación. El mismo permite buscar y visualizar las diversas publicaciones que contiene la base de datos, cada una con el número de veces que ha sido citada. Permite obtener un perfil de los autores donde muestra también la red de co-autoría con el número de publicaciones en conjunto. Estas publicaciones se encuentran distribuidas por intervalos de años. También presenta un buscador para filtrar la búsqueda y proporciona la dirección original del documento, además de descargar el registro bibliográfico a diferentes formatos. No se tiene acceso al código fuente.

1.3.3 Google Académico

El sistema Google Académico³ es una biblioteca de propósito general que a diferencia de otros es capaz de indexar los repositorios institucionales que estén públicos en internet. Permite visualizar de las publicaciones el año correspondiente a cada una y el número de citas, de versiones y permite descargar el registro bibliográfico a diferentes formatos. Se puede obtener un perfil de los autores donde muestra además del listado de publicaciones, el número de citas y dos indicadores bibliométricos: el índice h y el índice i10. Muestra también una gráfica con la cantidad de publicaciones por año de cada autor. Contiene un buscador como los demás sistemas y la opción de descargar los documentos que estén disponibles.

1.3.4 Microsoft Academic Search

Microsoft Academic Search⁴, cubre más de 38 millones de publicaciones y más de 19 millones de autores. Es una herramienta que utiliza .net y ofrece la posibilidad de recuperar por autor, conferencia, revista, organización, año, especialidad y una fecha concreta de publicación o un intervalo de fechas. Desde la página inicial de búsquedas, ofrece también la posibilidad de obtener los datos desde

² Accesible a través de la página <http://dblp.uni-trier.de/>

³ Accesible a través de la página <https://scholar.google.com/>

⁴ Accesible a través de la página <http://academic.research.microsoft.com/>



diferentes perspectivas, por recuentos de citas, palabras clave, autores principales y más. Muestra las relaciones de un autor con su comunidad científica, las redes de colaboración de un autor concreto, las coautorías, etc. Presenta algunos indicadores sobre todo de productividad e índices g y h.

1.3.5 Resultados del estudio realizado

Los sistemas anteriores permiten a los usuarios la búsqueda de documentación científica. Todos presentan un buscador con opciones de filtrados para agilizar el proceso y una serie de información en cada una de sus publicaciones. Pero solo dos de estos sistemas: Google Académico, Microsoft Academic Search, muestra el cálculo de dos indicadores bibliométricos: índice h e índice i10, que permiten al usuario tener mayor información sobre los autores.

Luego de analizar las características y ventajas de los sistemas estudiados, se decidió establecer un conjunto de indicadores, con el objetivo de arribar a las conclusiones del estudio de las soluciones anteriores. En la siguiente tabla se muestra el cumplimiento o no de dichos sistemas con los indicadores establecidos.

Tabla 4. Comparación de Sistemas existentes.

Sistema	De código abierto	Python lenguaje programación utilizado	como de	Cálculo indicadores bibliométricos	de	Integración como módulo para el sistema RBC
CiteSeerX	X	-		-		-
DBLP	-	-		-		-
Google Académico	-	-		X		-
Microsoft Academic Search	-	-		X		-

Después del análisis realizado, se arriba a la conclusión de que los sistemas estudiados sirven de base para la elaboración del nuevo módulo, pues representan una fuente de valor para la investigación y aportan conocimientos básicos sobre las funcionalidades a tener en cuenta. No obstante, solo representan una solución parcial a la problemática de la presente investigación y no cumplen con los



indicadores establecidos, por lo que se mantiene la necesidad de implementar un nuevo módulo para el sistema RBC.

1.4 Ambiente de desarrollo

Para el desarrollo del módulo que se propone en el presente trabajo, se seleccionó primeramente la metodología que guiará el proceso de desarrollo y seguidamente, los lenguajes, y un conjunto de herramientas y tecnologías.

1.4.1 Metodología de desarrollo

Para una selección adecuada de las metodologías, lenguajes y herramientas a utilizar en el desarrollo de la investigación se realizó un estudio, el cual se refleja a continuación:

Una metodología de desarrollo es *“un conjunto de actividades organizadas que tienen por objetivo la realización de un trabajo. Para cada actividad se define, además de las entradas y salidas, la forma en la que debe llevarse a cabo”*. (Moine, 2013)

Por las características del desempeño del trabajo, teniendo en cuenta que el equipo de desarrollo está compuesto por 2 integrantes, existe un intercambio directo con el cliente y que los requisitos pueden ser cambiantes, se decide optar por una metodología ágil, dentro de ellas se escoge AUP-UCI.

Para la implementación del nuevo módulo se cuenta con poco tiempo de desarrollo, y una de las ventajas de AUP radica en la estimación de tiempo. También simplifica la documentación de algunos procesos de desarrollo. Finalmente, como factor determinante, se tuvo en consideración que AUP es la metodología usada actualmente en la UCI, con algunas modificaciones.

A continuación, se describen los principales conceptos y características de la metodología AUP-UCI.

1.4.1.1 AUP-UCI

Según Galindo, la metodología AUP *“...heredera de otros paradigmas como la programación extrema (por sus siglas en inglés XP) y (por sus siglas en inglés RUP). Consta de principios y prácticas influyentes*



en la construcción del software en armonía con la documentación esencial de entregables específicos para el entendimiento de la solución.” (Galindo, 2012)

AUP contiene 4 fases, esas son: inicio, elaboración, construcción y transición; por su parte, para la modificación, en AUP-UCI se mantiene la fase de inicio, se unifican las otras tres denominándose ejecución y se le agrega la fase de cierre.

Inicio: en esta etapa se realiza la planeación, analizando las estimaciones de costos y tiempo para determinar si es factible o no el desarrollo del proyecto.

Ejecución (Elaboración, Transición, Construcción): en esta etapa se desarrolla el software, considerando los requisitos y la arquitectura, es implementado y entregado al cliente.

Cierre: se analizan los resultados obtenidos y se desarrolla formalmente la culminación del proyecto.

En cuanto a las disciplinas, mantiene 7 disciplinas y consta de 4 escenarios, después de haberse realizado el estudio del negocio, el que más se adapta es el escenario 2.

Escenario 2: tiene como objetivo fundamental la presentación de información, lo que es fundamental en el presente caso, para conocer los indicadores bibliométricos con los que se trabaja y plantea que no es necesario incluir en el sistema, las responsabilidades de las personas que realizan las actividades.

Este escenario propone que para el modelado del negocio se utilice un Modelo Conceptual y para el sistema, un Diagrama de Caso de Uso del Sistema.

1.4.2 Lenguajes

1.4.2.1 Lenguaje de programación

Para el desarrollo de la aplicación se emplea el lenguaje de programación Python debido a que es el utilizado en el desarrollo de las versiones anteriores del sistema RBC.

- **Python 2.7.11:** sus principales características son (Duque, 1990):



- Lenguaje interpretado o de script: se ejecuta utilizando un programa intermedio llamado intérprete, en lugar de compilar el código a lenguaje máquina que pueda comprender y ejecutar directamente una computadora.
- Tipado dinámico: no es necesario declarar el tipo de dato que va a contener una determinada variable, sino que su tipo se determinará en tiempo de ejecución según el tipo del valor al que se asigne y el tipo de esta variable puede cambiar si se le asigna un valor de otro tipo.
- Fuertemente tipado: no se permite tratar a una variable como si fuera de un tipo distinto al que tiene, es necesario convertir de forma explícita dicha variable al nuevo tipo previamente. Por ejemplo, si tenemos una variable que contiene un texto (variable de tipo cadena o string) no podremos tratarla como un número (sumar la cadena "9" y el número 8). En otros lenguajes el tipo de la variable cambiaría para adaptarse al comportamiento esperado, aunque esto es más propenso a errores.
- Multiplataforma: el intérprete de Python está disponible en multitud de plataformas (UNIX, Solaris, Linux, DOS, Windows, OS/2, Mac OS, etc.) por lo que si no utilizamos librerías específicas de cada plataforma nuestro programa podrá correr en todos estos sistemas sin grandes cambios.
- Orientado a objetos: es un paradigma de programación en el que los conceptos del mundo real relevantes para nuestro problema se trasladan a clases y objetos en nuestro programa. La ejecución del programa consiste en una serie de interacciones entre los objetos.

1.4.2.2 HTML5

El Lenguaje de Marcado de Hipertexto (por sus siglas en inglés HTML), es un conjunto de etiquetas o comandos, complementados en la mayoría de los casos por extensiones que permiten dar formato a un archivo, con el objetivo básico de crear un documento que pueda ser visualizado en forma de página web y que, además, pueda, por medio de dichas etiquetas, tener la estructura o forma deseada por quien la diseñó.

"HTML es un lenguaje de marcado utilizado para definir la estructura y contenido de una página o documento web." (Manz, 2015)



1.4.2.3 JavaScript

Es el lenguaje de script para páginas web: *“es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase”* (Mozilla Developer Network, 2017). Es orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. En la implementación del nuevo módulo se utilizó en la funcionalidad de calcular visibilidad o impacto (Ley de Lotka), validando que la entrada de datos solo fuera numérica.

1.4.2.4 CSS3

El lenguaje de hojas de estilo en cascada (Cascading Style Sheets, CSS por sus siglas en inglés): *“la idea de CSS es la de utilizar el concepto de separación de presentación y contenido, intentando que los documentos HTML incluyan sólo información y datos, relativos al significado de la información a transmitir (el contenido), y todos los aspectos relacionados con el estilo (diseño, colores, formas, etc....) se encuentren en un documento CSS independiente (la presentación).”* (Manz, 2017)

1.4.2.5 Lenguaje de modelado

El lenguaje de modelado unificado (por sus siglas en inglés UML) *“...es una de las herramientas más utilizadas en el mundo actual del desarrollo de software, esto se debe a que permite a los desarrolladores crear diseños que engloben sus propósitos de manera sencilla y fácil de comprender para otras personas. El UML está compuesto por diversos elementos gráficos que se combinan para formar diagramas. Proporciona características que permiten organizar y extender los diagramas. Es necesario resaltar que UML indica qué es lo que supuestamente hará el sistema, pero no cómo lo hará”.* (Schmuller, 2000)

Mediante los gráficos, este lenguaje permite especificar, incluyendo aspectos conceptuales tales como las funciones del sistema y los procesos de negocio, un software.

1.4.3 Herramientas y tecnologías

El componente de indicadores estadísticos bibliométricos propuesto por la presente investigación se integró con el sistema RBC, por lo que se consideró utilizar las mismas herramientas con las que se implementó el sistema RBC. A continuación, se realiza una breve descripción de las mismas.



1.4.3.1 Framework

Un framework es un marco de trabajo, definiéndolo de forma sencilla, no es más que un esqueleto en la implementación de una aplicación, contiene artefactos o módulos concretos de un software que permite unir los diferentes componentes de un proyecto.

“...Un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta y tiene como objetivo acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones”. (Gutiérrez., 2014)

Para el desarrollo del sistema se utiliza Django en su versión 1.6 ya que el mismo está escrito en Python que es el lenguaje de programación utilizado.

Django es para aplicaciones web gratuitas y de código abierto, que facilita y agiliza el desarrollo de una aplicación, utiliza como patrón arquitectónico Modelo Vista Plantilla, quien cumple con el paradigma Modelo Vista Controlador; donde el modelo sigue siendo el modelo, la vista sería el controlador y la plantilla la vista.

1.4.3.2 Entorno de desarrollo integrado

Un entorno de desarrollo integrado (por sus siglas en inglés IDE), es un programa informático que brinda un conjunto de componentes que hacen más fácil la programación. Conforman un ambiente favorable para los desarrolladores. Como entorno de desarrollo integrado se utiliza PyCharm en su versión 4.0.4., el cual permite la integración con el framework Django y soporta intérpretes de Python 2.7.

PyCharm4.0.4 *“... es un IDE desarrollado por la compañía JetBrains6, está basado en IntelliJ IDEA, el IDE de la misma compañía, pero enfocado hacia Java y la base de Android Studio. PyCharm tiene cientos de funciones que lo puede ver como una herramienta pesada, pero que valen la pena ya que ayuda con el desarrollo del día a día.”* (Blanco, 2014)

1.4.3.3 Herramienta CASE



Se selecciona Visual Paradigm for UML como herramienta de ingeniería de software asistida por computadora (por sus siglas en inglés CASE) para el modelado de la propuesta de solución. *“...es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todos los tipos de diagramas (diagramas de casos de uso, diagramas de requerimiento y diseño de bases de datos relacionales), código inverso, generar código desde diagramas y generar documentación. También proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML. Presenta licencia gratuita y comercial.”* (Blanco., 2014)

Esta herramienta permite representar todo tipo de diagramas en el ciclo de vida del desarrollo de software. Además, se integra con Python y PostgreSQL.

1.4.3.4 Sistema gestor de base de datos

Un sistema gestor de base de datos es una colección de datos relacionados entre sí, estructurados y organizados, además de un conjunto de programas que acceden y gestionan dichos datos.

Se utilizó PostgreSQL en su versión 9.5, es un sistema de gestión de bases de datos objeto-relacional y con su código fuente disponible libremente.

- **Interfaz gráfica para la manipulación de la base de datos:** se utiliza PgAdmin III en su versión 1.22.1 como herramienta de código abierto con el propósito general de diseñar, mantener y administrar la base de datos de PostgreSQL.

1.5 Conclusiones parciales del capítulo

Después de realizar un análisis del marco teórico de la investigación, se puede arribar a las siguientes conclusiones:

- Los sistemas estudiados que brindan información estadística de las publicaciones científicas indexadas, constituyeron una fuente de valor para la investigación y sirvieron de sostén para la elaboración de la misma, aportando conocimiento sobre las funcionalidades básicas a tener en cuenta; sin embargo, no se escogieron como solución ya que no resuelven el problema planteado.



- Para guiar el proceso de desarrollo de software se seleccionó la metodología AUP-UCI, ya que el equipo es pequeño, se cuenta con poco tiempo de desarrollo y actualmente es la metodología que guía el proceso de desarrollo de software en la UCI.
- Se seleccionaron las herramientas y las tecnologías a utilizar durante el desarrollo de la solución.



CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

El presente capítulo tiene un enfoque práctico y funcional, cuyo objetivo es describir los elementos de la propuesta de solución. Se describen los artefactos requeridos en la planificación definidos por la metodología AUP-UCI. Se desarrolla el modelo de negocio, además del modelado del sistema, describiendo los requisitos funcionales y no funcionales que se deben cumplir para satisfacer las necesidades del cliente. Se define la arquitectura utilizada y los patrones de diseño. Además, se muestran los diagramas correspondientes, como son los de clases empleando estereotipos web, el de paquete y los de colaboración; y el rediseño de la base de datos.

2.2 Propuesta de solución

“El sistema RBC, permite realizar búsquedas de publicaciones mediante un artículo, término o autor determinado” (Jimenez, 2015). Como se mencionó con anterioridad, este sistema no permite el cálculo de una serie de indicadores estadísticos-bibliométricos que son necesarios conocer para contribuir a la toma de decisiones. Teniendo en cuenta esto, el nuevo módulo desarrollado incorpora funcionalidades que permiten realizar estos cálculos.

El usuario realiza al sistema el pedido de información de los indicadores implementados; el sistema realiza la consulta a la base de datos para obtener la información necesaria para el cálculo de los indicadores y muestra el resultado al usuario.

Los indicadores calculados son:

- Productividad de los autores o ley de Lotka.
- El índice h.
- El índice i10.
- El índice g.
- El indicador bibliométrico de visibilidad o impacto, específicamente el factor de impacto.
- El indicador de productividad; dentro de el se determinan:
 - Cuál o cuáles son las publicaciones y/o autores de mayor impacto en un área determinada de la investigación.
 - Cuál es el autor que más publicaciones realiza sobre el tema.



- El número de artículos realizados por un autor, grupo, institución o país.
- El número de artículos publicados por las revistas determinado en un período de tiempo determinado.
- Cantidad de artículos comprendidos por las principales áreas de investigación de una disciplina.
- El número de relaciones establecidas entre diferentes autores, grupos, instituciones o países para la publicación de artículos.

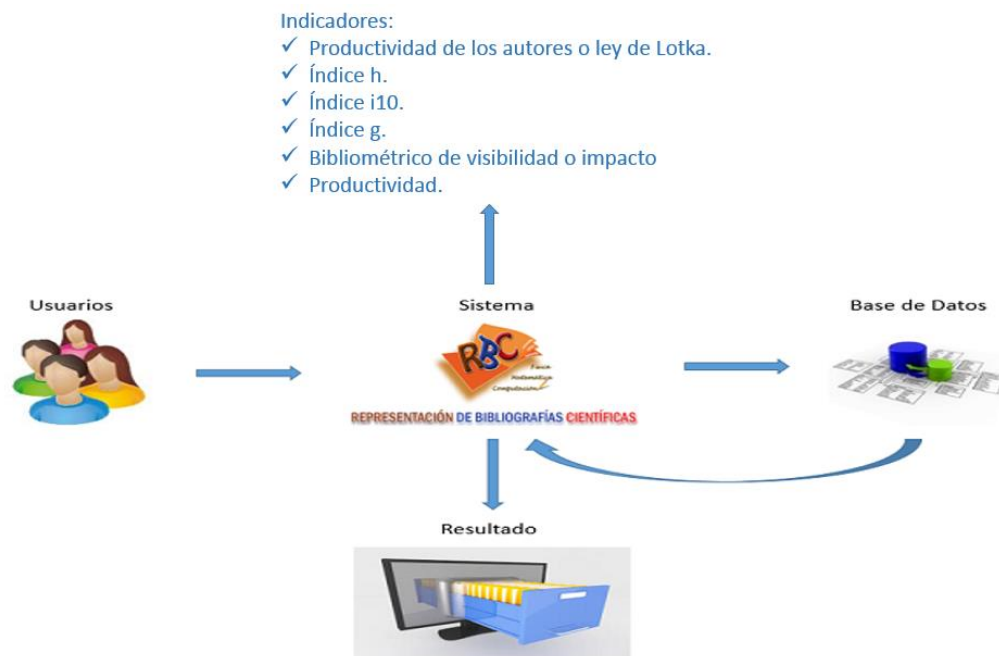


Figura 3. Propuesta de solución.

2.3 Modelo Conceptual

Un modelo conceptual tiene como objetivo “*identificar y explicar los conceptos significativos en un dominio de problema, identificando los atributos y las asociaciones existentes entre ellos*” (Pressman, 2010). En el presente modelo se definen los conceptos relacionados con el sistema RBC.

- **Usuario:** persona que por medio de un ordenador puede acceder a las funcionalidades de la herramienta; es quien solicita la búsqueda de artículos a través de un término, fragmento o un actor y puede acceder a los diversos indicadores bibliométricos referentes a esa búsqueda.



- **Administrador:** es un usuario del sistema que tiene la responsabilidad de implementar, configurar, mantener, monitorizar, documentar y asegurar el correcto funcionamiento del sistema. Esta tarea se extiende a la planificación de crecimiento del sistema, como también a la importante tarea de respaldo de la información contenida en ellos.
- **Publicaciones científicas:** documentos científicos que se encuentran en la base de datos, y que serán mostrados al usuario como resultado de su búsqueda.
- **Indicadores bibliométricos:** ofrecen nuevas informaciones al usuario y permiten averiguar el impacto causado por un trabajo científico, además de manejar, clasificar y analizar grandes volúmenes de publicaciones científicas.

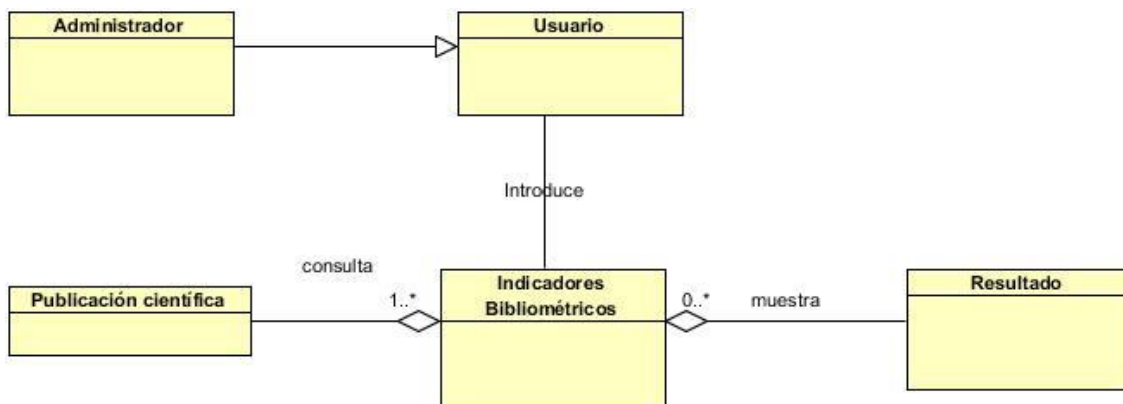


Figura 4. Modelo Conceptual

2.4 Requisitos del sistema

Los requisitos son la definición de las necesidades del sistema, todas las características que debe tener antes de ser culminado. Su identificación es una tarea compleja para el equipo de desarrollo, pues debe satisfacer las necesidades de los usuarios finales y del cliente; dándole así, cumplimiento a los objetivos propuestos.

2.4.1 Requisitos funcionales del sistema

Los requisitos funcionales definen las funciones del sistema de software, y una función es descrita como un conjunto de entradas, comportamientos y salidas. Para determinarlos se desarrollan tres actividades; primero se realiza la captura de requisitos, luego su definición y por último la validación.



Tabla 5. Requisitos funcionales del sistema RBC.

Número	Nombre	Descripción
RF1	Mostrar indicador de productividad	<p>Determina diversos datos de las publicaciones dependiendo de la búsqueda que el usuario realice:</p> <ul style="list-style-type: none"> ○ Cuál es el autor, grupo o institución que más publicaciones realiza sobre el tema. ○ El número de artículos realizados por un autor, grupo, institución o país. ○ El número de artículos publicados por las revistas sobre un tema determinado en un período de tiempo determinado. ○ Cantidad de artículos comprendidos por las principales áreas de investigación de una disciplina. ○ El número de relaciones establecidas entre diferentes autores, grupos, instituciones o países para la publicación de artículos.
RF2	Calcular la productividad de los autores (Ley de Lotka)	Determina que partiendo de un número de autores con un solo trabajo sobre un tema determinado, es posible predecir el número de autores con n trabajos mediante la ecuación (2) del acápite 1.2.
RF3	Calcular visibilidad o impacto	Permite calcular el Factor de impacto de una revista determinada. Mide la relación entre las citas recibidas y los artículos publicados. Se calcula mediante la ecuación (1) del acápite 1.2.
RF4	Mostrar índice h	<p>Permite determinar la distribución de las citas que han recibido los trabajos científicos de un investigador.</p> <p>Para ello el número de citas tiene que ser menor o igual al número de publicaciones.</p>
RF5	Mostrar índice g	Un autor tiene un índice g cuando, considerando los g artículos más citados de dicho autor, la cantidad de citas acumuladas por estos g artículos es superior a g al cuadrado.
RF6	Mostrar índice i10	Recorre las publicaciones que se han citado al menos 10 veces.

2.4.2 Requisitos no funcionales



Los requerimientos no funcionales se conocen como un conjunto de características de calidad, que es necesario tener en cuenta al diseñar e implementar el software. Son requisitos de portabilidad, de reutilización, de entorno de desarrollo, de usabilidad, de disponibilidad que hacen al producto atractivo a la vista del cliente, usable y confiable. Estos requerimientos se agrupan en varias categorías:

Orientados al usuario:

- **Rendimiento y escalabilidad:** el sistema tiene un tiempo de respuesta rápido, comprobado mediante la prueba de carga y estrés.
- **Apariencia e interfaz:** los usuarios encuentran sencillo el uso de la interfaz, además de atractiva a la vista. Todos los mensajes aparecen en idioma español. El nuevo módulo no modificó la base de la interfaz.

Hardware

- Para las PC clientes:
 - Requerimientos 512 MB de RAM recomendada o superior.
 - Tarjeta de red para establecer la conexión.
- Para el servidor:
 - Es recomendado utilizar un computador con procesador Dual Core o superior con 4 GB de memoria RAM y 500 Gb de disco duro.
 - Tarjeta de red para establecer la conexión.

Software

- Para las PC clientes:
 - Sistema Operativo Windows 7 o superior, o GNU/Linux con interfaz gráfica y soporte para conectarse a la red.
 - Para la utilización del sistema se requerirá el uso de un navegador web, preferentemente Mozilla Firefox en su versión 30.0 o superior.
- Para el servidor:
 - Se requiere el sistema gestor de bases de datos PostgreSQL 9.5, el marco de trabajo Django 1.6, las bibliotecas de Python necesarias como *nltk*, *gensim*, *sklearn*, entre otras.



- Sistema Operativo Windows 7 o superior y GNU/Linux con interfaz gráfica y soporte para conectarse a la red.

2.5 Definición de los casos de uso

A partir del análisis de los requisitos funcionales del sistema se definieron los siguientes casos de uso:

CU1: Mostrar índice de productividad.

CU3: Calcular visibilidad o impacto.

CU2: Calcular la productividad de los autores (Ley de Lotka).

CU4: Mostrar índice h.

CU5: Mostrar índice g.

CU6: Mostrar índice i10.

2.5.1 Actores del sistema

Un actor es una entidad externa al sistema que guarda una relación con este y que le demanda una funcionalidad. Se incluyen los operadores humanos, pero también todos los sistemas externos.

Tabla 6. Definición de los actores.

Actor	Descripción
Usuario	A partir de que el usuario realice la búsqueda que necesita para obtener el o los documentos requeridos, que se encuentra en la base de datos. Podrá conocer una serie de indicadores bibliométricos, ya sea el índice h, el índice i10, el índice g, la productividad de los autores o Ley de Lotka, el indicador de visibilidad e impacto, específicamente el factor de impacto, que se determina solo para las revistas, y una serie de información que corresponde a el indicador de productividad; lo que le permitirá seleccionar mejor cuál es el documento más afín para él.

2.5.2 Diagrama de casos de uso del sistema

El diagrama de casos de uso del sistema muestra los pasos que el actor sigue para realizar una operación. Permite ilustrar los requerimientos del sistema al mostrar cómo reacciona a eventos que se producen en su ámbito o en él mismo.

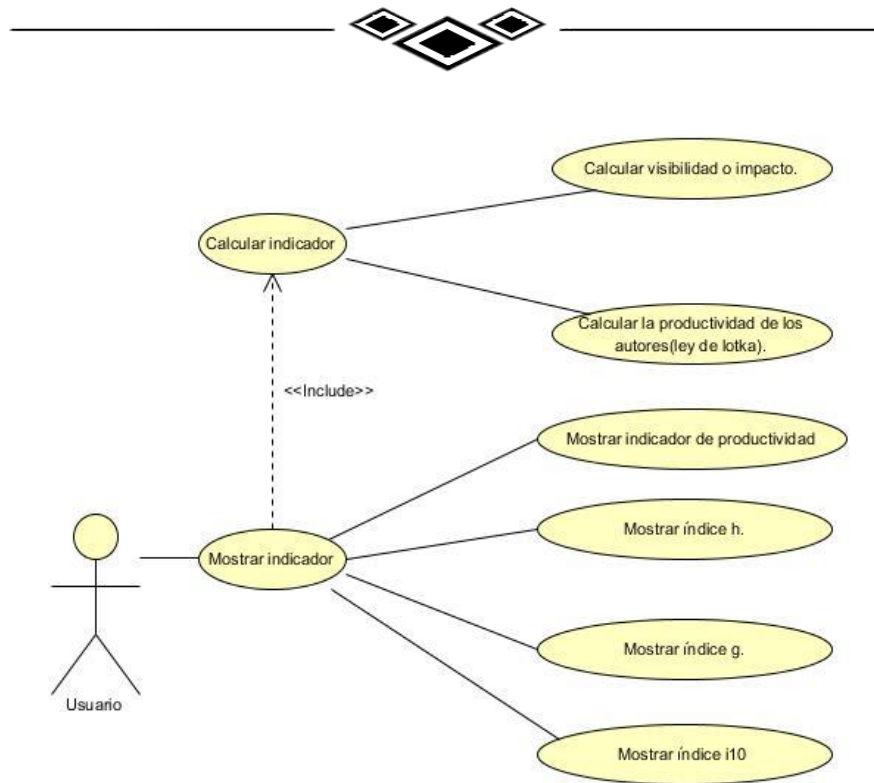


Figura 5. Diagrama de caso de uso del sistema RBC.

2.5.3 Descripción de los casos de uso del sistema.

A continuación, se muestra la descripción del caso de uso (CU) Calcular productividad de los autores (Ley de Lotka). El resto de los CU pueden ser consultados en el Anexo 1.

CU 2: Calcular la productividad de los autores (Ley de Lotka).

Tabla 7. Descripción del CU: Calcular la productividad de los autores (Ley de Lotka).

Nombre	Calcular la productividad de los autores (Ley de Lotka).
Objetivo	Determinar la productividad de los autores (An) o Ley de Lotka.
Actor(es)	Usuario
Resumen	Para determinar $A(n)$, se necesita conocer la cantidad de autores con un solo trabajo (K) y un número de artículos(n) sobre el cual se quiere conocer la cantidad de autores que han publicado esa cantidad de artículos, mediante la ecuación (1) del acápite 1.2.
Complejidad	Alta



Prioridad	Media
Precondiciones	Realizar búsqueda.
Postcondiciones	
Flujo de eventos	
Flujo básico Calcular la productividad de los autores (Ley de Lotka).	
Actor	Sistema
1.	Redirecciona a la página con todas las publicaciones correspondientes a la búsqueda.
2.	Muestra la opción de conocer la productividad de los autores.
3.Escribir el valor de n.	
4.	Muestra A(n) correspondiente al valor de n.
Flujo alternativo Calcular productividad de los autores (Ley de Lotka).	
3.1. Escribir el valor de n con cualquier carácter que no sea un número entero.	
	Muestra un error, "debe entrar un número entero"
Prototipo de interfaz gráfica para el CU: Calcular la productividad de los autores (Ley de Lotka).	



Administración Información

as Filter

Resultados de la búsqueda para: "as":

[libro] **Asymptotic Behavior of the Number of Lost Messages (Citaciones: 2)**

The goal of the paper is to study asymptotic behavior of the number of lost messages. long messages are assumed to be divided into a random number of packets which are transmitted independently of one another, an error in transmission of a packet results in the loss of the entire message. messages arrive to the m/gi/1 finite buffer model and can be lost in two cases as either at least one of its packets is corrupted or the buffer is overflowed, with the parameters of the system typical for models of information transmission in real networks, we obtain theorems on asymptotic behavior of the number of lost messages.

Li-fung Chang, Vyacheslav M. Abramov, Yeneisy Barroso, Pablo Miro
<http://academic.research.microsoft.com/Publication/4407175/asymptotic-behavior-of-the-number-of-lost-messages>

[ponencia] **High-Bandwidth Encryption with Low-Bandwidth (Citaciones: 1)**

This paper describes a simple protocol, the remotely keyed encryption protocol (rkep), that enables a secure, but bandwidth-limited, cryptographic smartcard to function as a high bandwidth secret-key encryption and decryption engine for an insecure, but fast, host processor: the host processor assumes most of the computational and bandwidth burden of each cryptographic operation without ever learning the secret key stored on the card. by varying the parameters of the protocol, arbitrary...

Matt Blaze
<http://www.cis.upenn.edu/~blaze/F15F23PaperFall04/Readings/Blaze.pdf>

Ordenar por el título

Ordenar la búsqueda:

Ley de Lotka:

"Partiendo del número de autores con un solo trabajo, es posible predecir el número de autores con n trabajos."

n = 1 = 10 autores

↑

2.6 Etapa de diseño

Permite aplicar distintas técnicas y principios con el propósito de definir el sistema, con los suficientes detalles como para permitir su correcta implementación. Modela una entidad que es construida más adelante. "Es una simplificación de la realidad, es necesario en la construcción del software para comunicar la estructura de un sistema complejo, especificar el comportamiento deseado del sistema, comprender mejor lo que se está construyendo, descubrir oportunidades de simplificación y reutilización" (Booch, y otros, 1999). Para ello se realizaron el diagrama de paquetes, los diagramas de clases del diseño utilizando estereotipos web y los diagramas de colaboración.

2.6.1 Diseño Arquitectónico

La arquitectura del software es el diseño de más alto nivel de la estructura de un sistema, indica también su funcionamiento e interacción entre las partes. La arquitectura se selecciona a partir de las



restricciones y los requerimientos que debe cumplir el sistema y consiste en un conjunto de patrones y abstracciones que proporcionan un marco definido y claro para interactuar con el código fuente del software.

Para el desarrollo del nuevo módulo se empleó la arquitectura cliente-servidor, la cual no es más que un modelo de aplicación donde las tareas se reparten entre el servidor que sería el proveedor y los clientes que serían los demandantes. Para ello, el cliente realiza una solicitud al servidor mediante el protocolo HTTP, quien le devuelve una respuesta, puede ejecutarse sobre una sola computadora o distribuido a través de una red de computadoras.

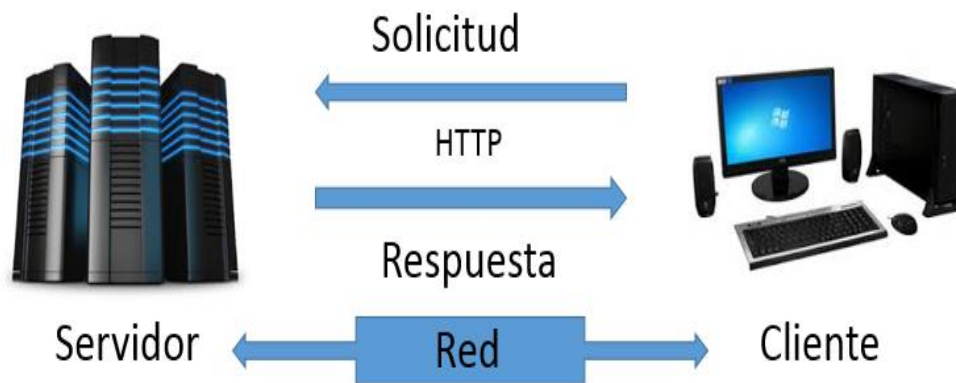


Figura 6. Arquitectura Cliente-Servidor.

Para facilitar la creación del nuevo módulo se utilizó el framework de desarrollo Django, el cual está escrito en Python y utiliza una modificación de la arquitectura Modelo-Vista-Controlador (MVC) llamada Modelo-Plantilla-Vista, conocido por sus siglas provenientes del inglés como Model-Template-View (MTV).

En Django la capa modelo es la misma, la capa vista sería la controladora y la plantilla (templates) sería la vista.

- **Capa Modelo:** “El modelo define los datos almacenados, se encuentra en forma de clases de Python, cada tipo de dato que debe ser almacenado se encuentra en una variable con ciertos parámetros, posee métodos también. Todo esto permite indicar y controlar el comportamiento de



los datos” (Revista de Información, Tecnología y Sociedad, 2012). Es decir, esta es la capa de acceso a la base de datos y contiene toda la información sobre los datos: cómo acceder a estos, cómo validarlos, y su relación y comportamiento.

- **Capa Plantilla:** *“La plantilla es básicamente una página HTML con algunas etiquetas extras propias de Django, en sí no solamente crea contenido en HTML (también XML, CSS, JavaScript, CSV, etc.)”* (Revista de Información, Tecnología y Sociedad, 2012). Es decir, esta es la capa de presentación. Esta capa recibe los datos y los organiza para la presentación: como algunas cosas son mostradas sobre una página web u otro tipo de documento.
- **Capa Vista:** *“La vista se presenta en forma de funciones en Python, su propósito es determinar qué datos serán visualizados, entre otras cosas más. El ORM de Django permite escribir código Python en lugar de SQL para hacer las consultas que necesita la vista. La vista también se encarga de tareas conocidas como el envío de correo electrónico, la autenticación con servicios externos y la validación de datos a través de formularios. Lo más importante a entender con respecto a la vista es que no tiene nada que ver con el estilo de presentación de los datos, sólo se encarga de los datos, la presentación es tarea de la plantilla.”* (Revista de Información, Tecnología y Sociedad, 2012). Es decir, esta es la capa de la lógica de negocios, contiene la lógica que accede al modelo y la delega a la plantilla apropiada: se puede asociar como un puente entre el modelo y las plantillas.

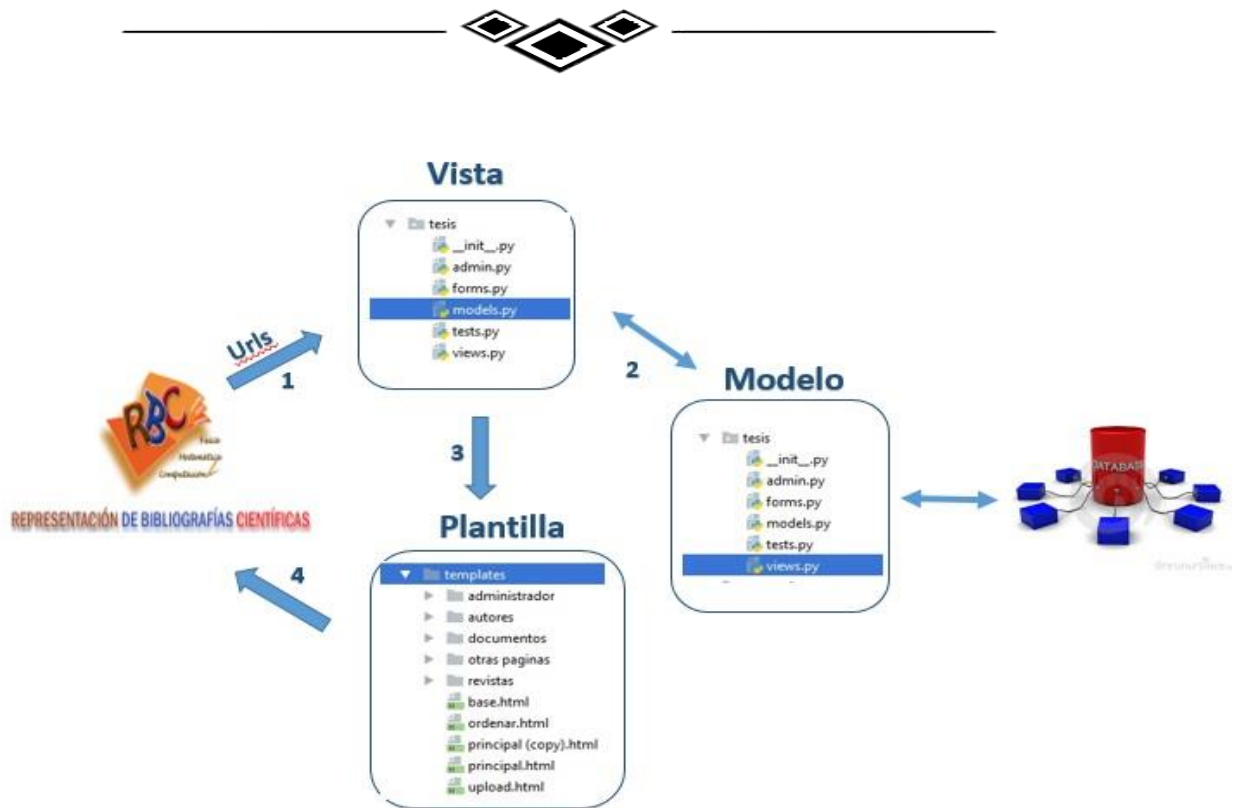


Figura 7. Funcionamiento del MTV en Django.

2.6.2 Patrones de diseño

Los Patrones **GRASP** (siglas provenientes del inglés Generales de Software para Asignación de Responsabilidades). Los patrones GRASP “describen los principios fundamentales de la asignación de responsabilidades a objetos expresados en forma de patrones” (Larman, 1999). Existen 9 patrones, pero los utilizados para la implementación del nuevo módulo son: experto, bajo acoplamiento, alta cohesión y creador, según el framework de desarrollo que se utiliza, Django.

- **Experto:** “Es el encargado de asignar la responsabilidad de la creación de un objeto o la implementación de un método a una clase que contenga toda la información necesaria para cumplir con dicha responsabilidad” (Pressman, 2010). Se puede decir que se distribuye el comportamiento entre las clases que contienen la información requerida. Son más fáciles de entender y mantener. Las clases Document, Reference y Person cuentan con la información necesaria para cumplir cada una de las responsabilidades que le corresponden.



```
class Document(models.Model):
    title=models.CharField(max_length=200)
    abstract=models.TextField(max_length=300)
    type=models.CharField(max_length=50,choices=TYPES_CHOICES)
    author_count=models.IntegerField()
    links=models.URLField()
    keyword = models.ManyToManyField(Keywords,through='Document_Keywords',max_length=100)
    person = models.ManyToManyField(Person,through='Author_ship',max_length=100)
    investigation_area = models.ForeignKey(InvestigationArea)
    def __unicode__(self):
        return self.title
```

Figura 8 Ejemplo de la clase Document.

- **Bajo acoplamiento:** “El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con qué las conoce y con qué recurre a ellas. En tal sentido, el término bajo acoplamiento significa que una clase no depende de muchas clases” (Pressman, 2010). La idea es que una clase dependa lo menos posible de otra, así, en caso de que una clase sea modificada se tenga la mínima repercusión posible en el resto.

La vista implementa métodos que son llamados por los Localizadores Uniformes de Recursos (por sus siglas en inglés URLs) pero los cambios que se realicen en la implementación de estos métodos no afecta en nada a las URLs.

A las clases Org_unit, Organization y Person se les asignan responsabilidades de forma tal que solo se comuniquen con las clases que se encargan de facilitar el proceso de pertenencia a cada una de ellas.

```
class Org_unit(models.Model):
    name=models.CharField(max_length=150)
    city=models.CharField(max_length=300)
    post_code=models.IntegerField()
    street=models.CharField(max_length=100)
    country=models.CharField(max_length=100)
    def __unicode__(self):
        return self.name
```

Figura 9 Ejemplo de la clase Org_unit.



- **Alta cohesión:** “Es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realizan un trabajo enorme” (Pressman, 2010). La información de las clases debe ser coherente.

A las clases Publication y Reference se le asignan responsabilidades con el objetivo de que trabajen en la misma área de la aplicación y que no tengan mucha complejidad.

```
class Publication(models.Model):
    article=models.ForeignKey(Document)
    journal=models.ForeignKey(Journal)
    volume=models.IntegerField()
    issue=models.CharField(max_length=10)
    publication_date=models.DateField()
    publication_year=models.IntegerField(max_length=4)

    def __unicode__(self):
        return self.publication_year
```

Figura 10 Ejemplo de la clase Publication.

- **Creador:** el mismo tiene como objetivo “indicar a qué clase se le asigna la responsabilidad de la creación de instancias, puesto que esta posee la información necesaria para la creación del objeto.” (Grosso, 2011)

Los formularios UserCreationForm y AuthenticationForm son responsables de la creación de objetos de tipo formulario.

```
def nuevo_usuario(request):
    if request.method == 'POST':
        form = UserCreationForm(request.POST)
        if form.is_valid():
            form.save()
            return HttpResponseRedirect('/usuario')
    else:
        form = UserCreationForm()
    return render_to_response('administrador/inserciones/usuario.html', {'form': form},
                             context_instance=RequestContext(request))
```

Figura 11 Formulario UserCreationForm para validar un nuevo usuario.



2.7 Modelo físico de los datos

El modelo de datos permite describir los tipos de datos que se encuentran en la base de datos y como se relacionan, cumpliendo con las restricciones requeridas para alcanzar el objetivo propuesto. Los elementos esenciales son:

- **Entidades:** objetos de los cuales el sistema guarda su información, en este caso son los documentos y los vectores.
- **Atributos:** características de las entidades, las cuales se clasifican en obligatorios, opcionales, claves foráneas y claves primarias.
- **Relaciones:** muestra la relación entre las entidades, la cual puede ser de uno a mucho, de uno a uno o de mucho a mucho.

El sistema RBC cuenta con una base de datos relacional. Debido a que el framework de desarrollo utilizado fue Django, la base de datos y tablas que componen la misma, se definen a través de las clases del modelo. A continuación, se describen cada una de ellas:

Document: clase responsable de registrar, modificar, eliminar y listar los detalles de un documento.

Keywords: clase responsable de registrar, modificar, eliminar y listar los detalles de una palabra clave.

Document_keywords: clase responsable de registrar, modificar y eliminar los detalles de una palabra clave de un documento.

Org_unit: clase responsable de registrar, modificar, eliminar y listar los detalles de las unidades organizativas.

Publication: clase responsable de registrar, modificar y eliminar los detalles de una publicación.

Reference: clase responsable de registrar, modificar y eliminar los detalles de una referencia.

Affiliation: clase responsable de registrar, modificar y eliminar los detalles de una afiliación.

Autor_ship: clase responsable de registrar, modificar y eliminar los detalles de una comunidad de autores.

Belongs_to: clase responsable de registrar, modificar y eliminar los detalles que pertenezcan a una unidad organizativa o una organización.



Journal: clase responsable de registrar, modificar, eliminar y listar los detalles de una revista.

Person: clase responsable de registrar, modificar, eliminar y listar los detalles de un autor.

InvestigationArea: clase responsable de registrar, modificar, eliminar y listar los detalles de un área de investigación.

Discipline: clase responsable de registrar, modificar, eliminar y listar los detalles de una disciplina.

Para el desarrollo del módulo propuesto se utilizará la misma base de datos descrita anteriormente, agregando las tablas “Country” y “Organization”, que se explican a continuación:

Country: clase responsable de registrar, modificar, eliminar y listar los detalles de un país.

Organization: clase responsable de registrar, modificar, eliminar y listar los detalles de una organización.

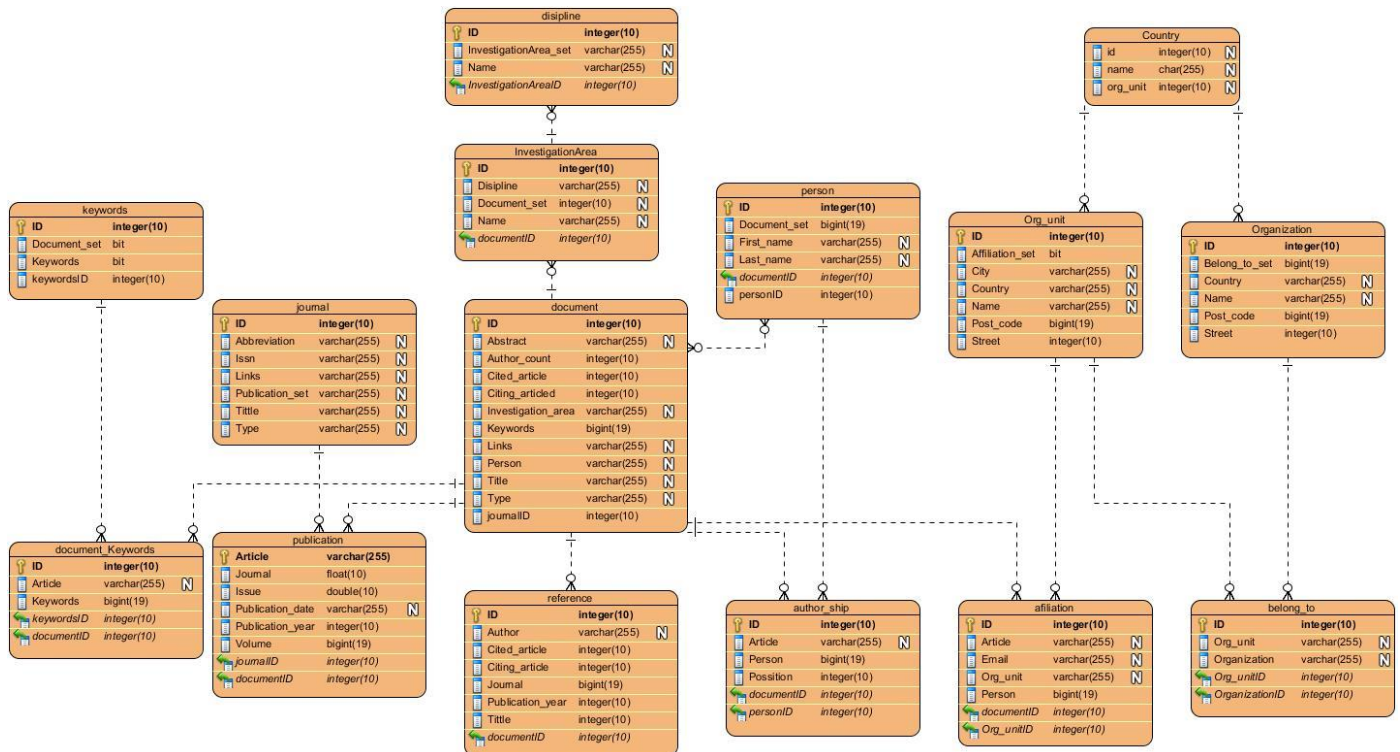


Figura 12. Modelo físico de los datos.

2.8 Diagrama de paquetes



Este diagrama muestra la relación que existe entre los paquetes de un modelo, muestra sus agrupaciones lógicas y la relación entre ellos.

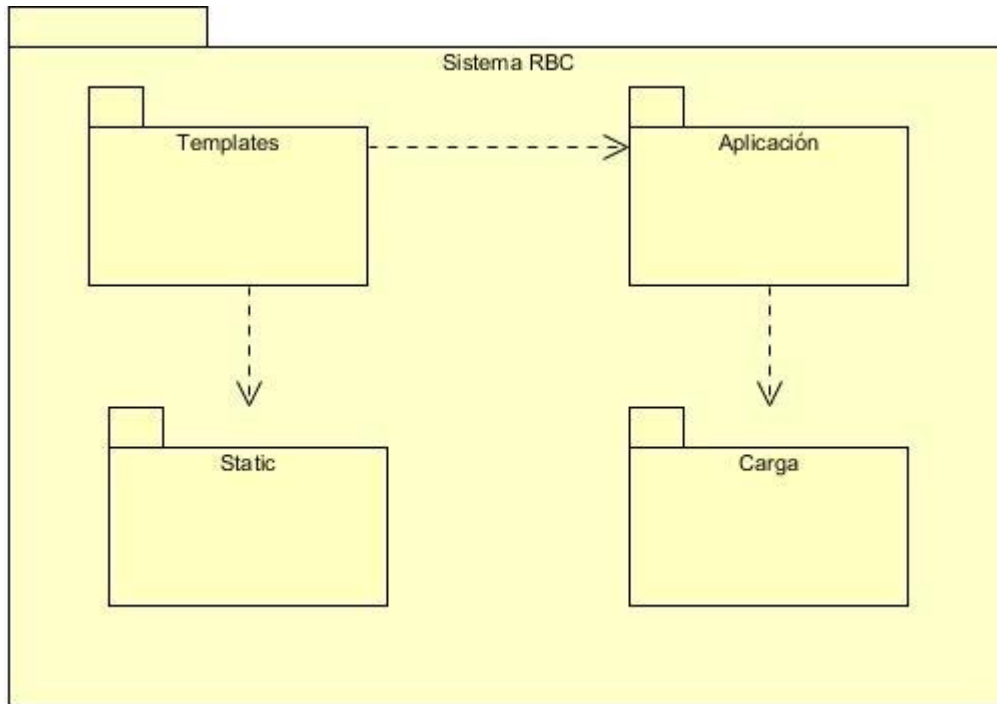


Figura 13 Diagrama de paquetes del sistema RBC.

- **Templates:** contiene todas las plantillas HTML.
- **Aplicación:** contiene las clases y funcionalidades implementadas.
- **Static:** contiene los archivos CSS, JavaScript y las imágenes utilizadas para el diseño y estilo de las plantillas HTML.
- **Carga:** contiene dos subdirectorios, uno llamado Documentos donde se almacena todos los documentos en distintos formatos que se encuentran en la base de datos. El otro subdirectorio es Carga, el cual contiene el documento subido por el usuario.

2.9 Diagrama de clases del diseño utilizando estereotipos web



Es un tipo de diagrama de estructura estática. Muestra las diferentes clases del sistema, además de sus atributos, métodos y la relación entre los objetos.

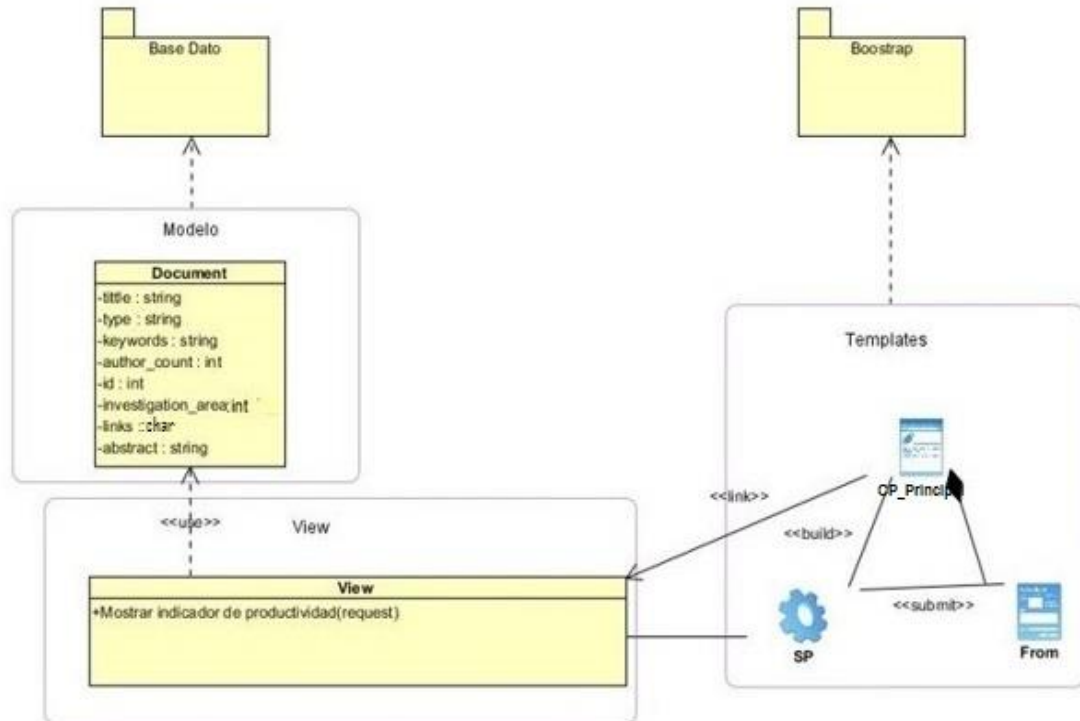


Figura 14. Diagrama de clases del diseño utilizando estereotipos web, Mostrar indicador de productividad.

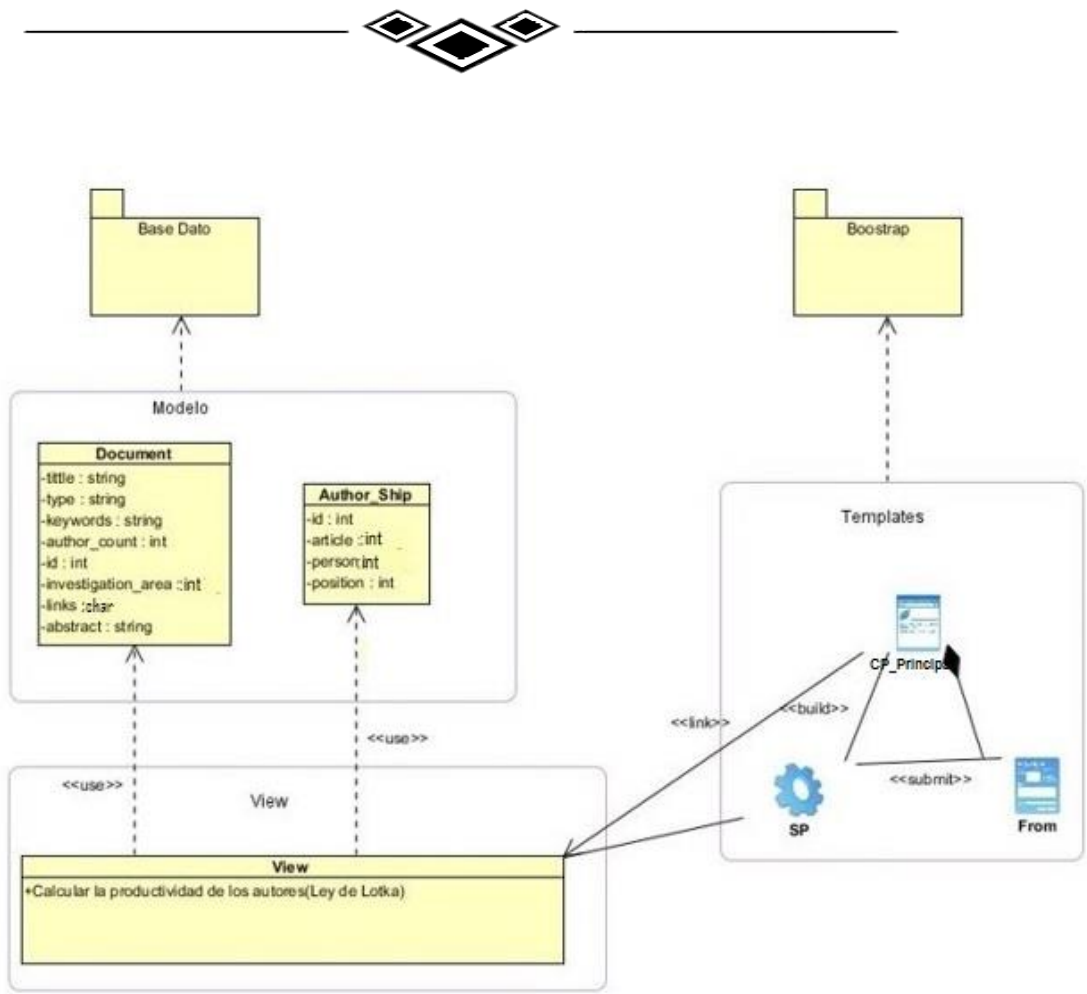


Figura 15. Diagrama de clases del diseño utilizando estereotipos web, Calcular la productividad de los autores (Ley de Lotka).

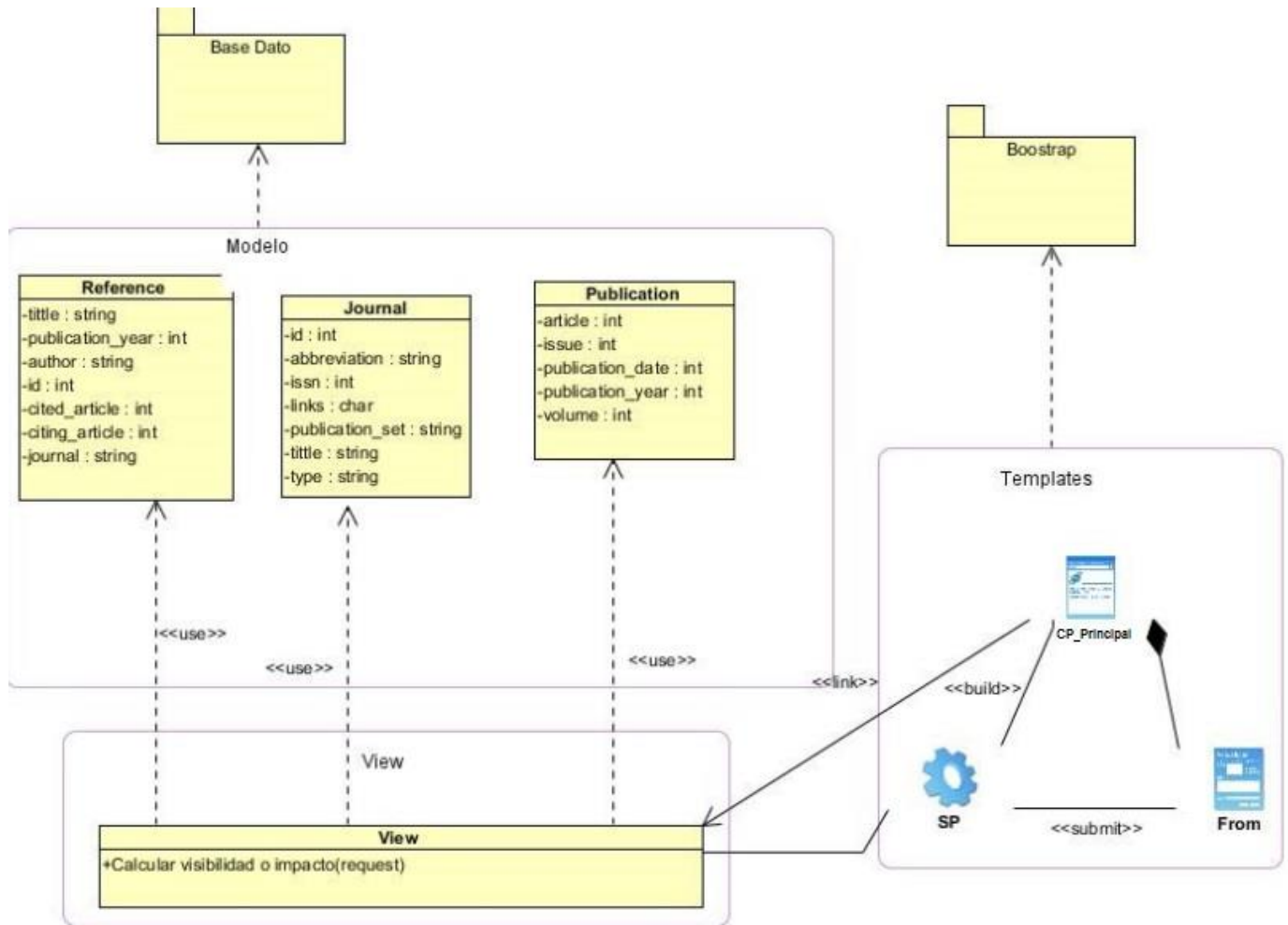


Figura 16. Diagrama de clases del diseño utilizando estereotipos web, Calcular visibilidad o impacto.

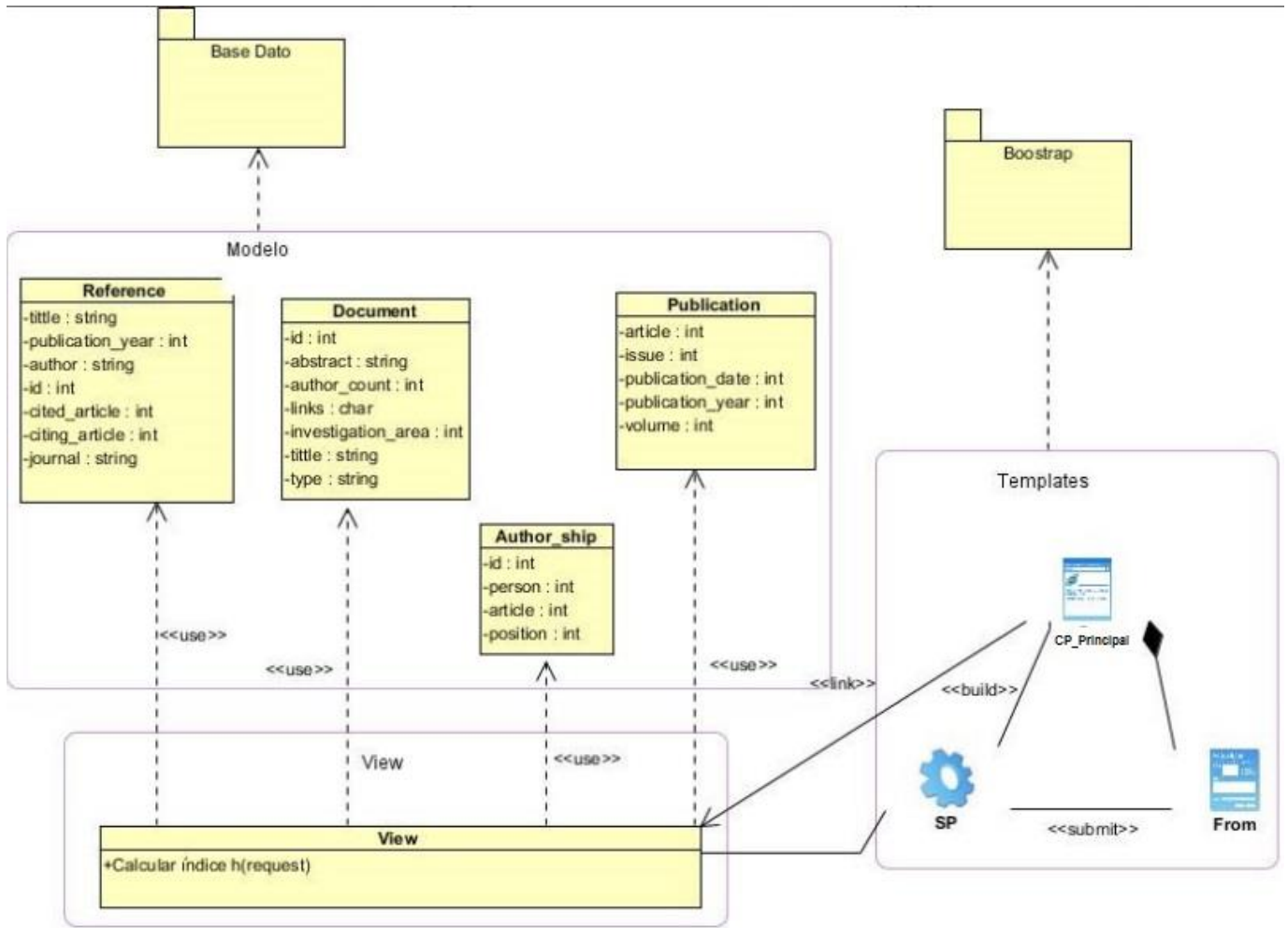


Figura 17. Diagrama de clases del diseño utilizando estereotipos web, Mostrar índice h.

Para consultar el resto de los diagramas de clases de diseño utilizando estereotipos web ver **Anexo II:** Diagramas de clases de diseño.

2.10 Conclusiones parciales del capítulo

En el capítulo se describieron las etapas de planificación y diseño de la metodología de desarrollo utilizada para la implementación del módulo:

- Para un mejor entendimiento del dominio del problema, se describió el modelo conceptual del módulo.



- El trabajo en conjunto del cliente y el equipo de desarrollo, posibilitaron la identificación de seis requisitos funcionales.
- Para el diseño del módulo se utilizó el patrón arquitectónico Modelo-Vista-Plantilla, implementado por el framework de desarrollo Django.
- Se elaboraron los diagramas de clases utilizando estereotipos web, para garantizar un mejor entendimiento del diseño propuesto.



CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBA

3.1 Introducción

El presente capítulo aborda todo lo referente a la implementación del sistema, incluyendo las pruebas que se le realizan para comprobar su correcto funcionamiento. Se describen los diagramas de componente y de despliegue, así como, los estándares de codificación empleados. De esta manera será fácil de comprender el sistema.

3.2 Etapa de implementación

Es la etapa que transcurre entre el diseño de la estructura y su definitivo funcionamiento. Se realiza la codificación de la propuesta de solución.

Para representar el sistema se desarrolla el diagrama de componentes y el de despliegue.

3.2.1 Diagrama de componentes

“El diagrama de componentes se utiliza para la representación de aspectos físicos del sistema, representa los componentes y las dependencias entre ellas”. (Booch, y otros, 1999). En él se aprecian los componentes que pueden compartirse entre sistemas o entre diferentes partes de un sistema.

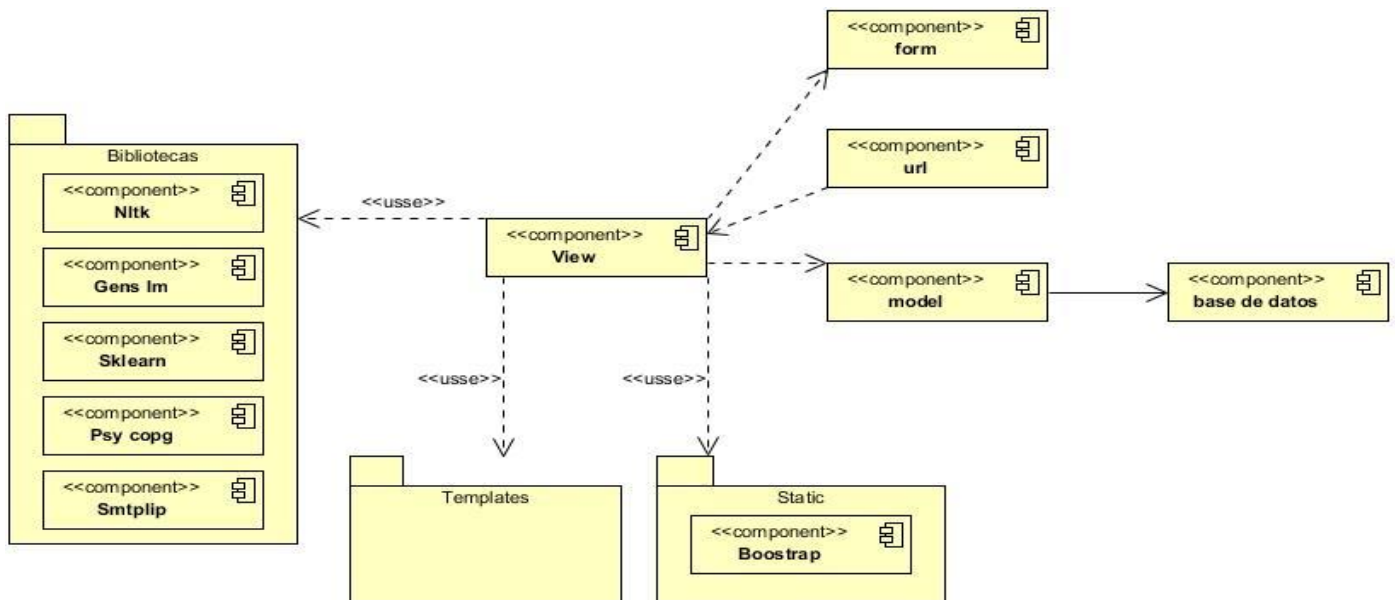


Figura 18. Diagrama de componente



- **Bibliotecas:** se muestran todas las librerías utilizadas.
- **Form:** se crean los objetos de tipo formularios.
- **URL:** contiene las direcciones que permiten el acceso a los métodos de la vista(view).
- **Model:** contiene las clases que permiten el acceso a la base de datos. Es la capa de acceso a datos.
- **Base de Datos:** contiene todas las tablas con los datos correspondientes y los atributos de cada uno de esos datos.
- **View:** contiene los métodos. Es quien maneja la lógica del negocio.
- **Templates:** contiene todas las vistas, los HTML. Es la capa de presentación.
- **Static:** contiene las presentaciones estáticas.

3.3 Estándares de codificación empleados

Un estándar de codificación comprende todos los aspectos de la generación de código. Debe reflejar un estilo armonioso, lo que permite que se puedan retomar partes realizadas por otros integrantes y facilitar un mantenimiento posterior del sistema por otros desarrolladores. Los estándares se realizan con el objetivo de alcanzar uniformidad en el código.

Con el objetivo de cumplir con lo antes expuesto, para el nuevo módulo se utilizaron los mismos estándares de codificación utilizados en el desarrollo de las versiones anteriores del sistema, para que de esa manera haya claridad y concordancia en el código fuente. Además de seguir la guía de estilos y referencia del lenguaje Python (Foundation, 2016), se utilizan:

- Iniciar el nombre de las variables de las clases con letra inicial mayúscula y en caso de ser un nombre compuesto se utiliza guión bajo (_).
- Al inicio de cada clase se realiza un comentario explicando el objetivo de la misma. Cada línea de un comentario empieza con # (numeral).
- Antes y después de la declaración de una clase o de una estructura y de la implementación de una función se deja una línea en blanco.
- Entre operadores lógicos y aritméticos se utiliza espacios en blanco.

3.4 Diagrama de despliegue



“Los elementos del diseño del despliegue indican la forma en la que se acomodarán la funcionalidad del software y los subsistemas dentro del ambiente físico de la computación que lo apoyará... Esto significa que el diagrama de despliegue muestra el ambiente de computación, pero no indica de manera explícita los detalles de la configuración.” (Pressman, 2010). El diagrama de despliegue se encarga de modelar la arquitectura en tiempo de ejecución. Muestra como los artefactos y elementos se trazan en los nodos, y la configuración de los mismos; los cuales se muestran a continuación en forma de una caja en tres dimensiones.

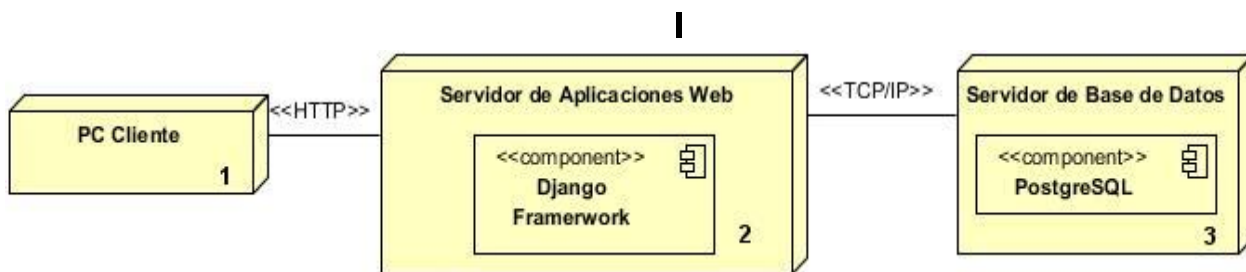


Figura 19. Diagrama de despliegue

1. **PC Cliente:** son las PC mediante las que se accederá al sistema.
2. **Servidor de Aplicaciones Web:** es el servidor donde estará disponible el sistema, además debe estar instalado el lenguaje de programación Python 2.7, el framework Django 1.6.
3. **Servidor de Base de Datos:** almacena los datos con los que interactúa el sistema. En el sistema se utiliza PostgreSQL.

3.4.1 Descripción de los protocolos de comunicación

- **HTTP:** el Protocolo de Transferencia de Hipertexto (por sus siglas en inglés HTTP) es orientado a transacciones. Es un protocolo cliente-servidor que sigue el esquema petición-respuesta. El cliente realiza una petición enviando un mensaje (Los mensajes HTTP son en texto plano, por ellos son más fáciles de depurar, pero también más largos) al servidor. El servidor le envía un mensaje que contiene el estado de la operación y la respuesta.
- **TCP/IP:** el Protocolo de Control de Transmisión/Protocolo de Internet (por sus siglas en inglés TCP/IP), se utiliza para conectar computadoras sobre una red de área local (LAN). En el caso del módulo desarrollado se utiliza para interconectarlo con el servidor de base de datos.



3.5 Pruebas de Software

Al desarrollar sistemas informáticos se corre con un gran riesgo de que se produzcan errores. Con el objetivo de detectar y corregir estos posibles errores se realiza la etapa de prueba. Luego de estas correcciones se debe obtener un producto de calidad, que cumpla con los requisitos planteados.

3.5.1 Estrategia de prueba seguida

Las pruebas al sistema pueden realizarse después de que el producto este culminado, el cual no es efectivo; o diariamente, lo que no es del agrado de los programadores. Por esas inconformidades se crearon pruebas que no están en ninguno de los dos extremos.

“...Esta estrategia toma una visión incremental de las pruebas, comenzando con la de unidades de programa individuales, avanza hacia pruebas diseñadas para facilitar la integración de las unidades y culmina con pruebas que ejercitan el sistema construido. Cada una de estas clases de pruebas se describe en las secciones que siguen.” (Pressman, 2010).

3.5.2 Pruebas de caja blanca

Esta prueba se le realiza al código fuente; probando con diversos valores de entrada y cerciorándose de que los valores de salida sean los correspondientes. Su objetivo es comprobar los flujos de información dentro de cada unidad, e incluso entre unidades y entre subsistemas.

- **Pruebas unitarias**

“...Las pruebas de unidad se enfocan en la lógica de procesamiento interno y de las estructuras de datos dentro de las fronteras de un componente...” (Pressman, 2010)

Para realizarlas se comienza probando el flujo de datos, el cuál es fundamental, luego la interfaz, las estructuras de los datos locales, las condiciones de fronteras y, por último, las diferentes rutas.

Se utilizó la biblioteca de Python (*unit testing*) y la definición de las pruebas se realizaron dentro del archivo tests.py, ya creado por el marco de trabajo Django. Para ejecutar los test o iniciar el servidor de prueba se hace uso del comando *“python manage.py test”*

Se les realizaron pruebas unitarias a los métodos factor de impacto mediante la búsqueda de una revista, a todas las búsquedas sencillas y profundas, lo que incluye la Ley de Lotka y al método autor, para verificar los índices h , g , e $i10$.



```
def test_author(self):
    response = self.client.get(
        '/resultados_búsqueda/autores/' + str(self.person_.id),
    )

    self.assertEqual(response.status_code, 200)
    self.assertEqual(response.context['ind_h'], 1)
    self.assertEqual(response.context['ind_i10'], 0)
    self.assertEqual(response.context['ind_g'], 1)
    self.assertEqual(response.context['citas'], 5)
```

Figura 20 Código de test.

```
Creating test database for alias 'default'...
.....
-----
Ran 6 tests in 1.205s

OK
Destroying test database for alias 'default'...
```

Figura 21. Resultado de la Prueba unitaria.

Se realizaron 6 casos de prueba, uno para cada nueva funcionalidad, en un tiempo de 1.205s.

3.5.3 Pruebas de caja negra

La prueba de caja negra se centra principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

Para preparar los casos de pruebas hacen falta un número de datos que ayuden a la ejecución de los estos casos y que permitan que el sistema se ejecute en todas sus variantes, pueden ser datos válidos o inválidos para el programa según si lo que se desea es hallar un error o probar una funcionalidad. Los datos se escogen atendiendo a las especificaciones del problema, sin importar los detalles internos del programa, a fin de verificar que el programa se ejecute satisfactoriamente.



Para desarrollar la prueba de caja negra, se utilizará la técnica de Partición de Equivalencia. La partición equivalente “...es una de las más efectivas pues permite examinar los valores válidos e inválidos de las entradas existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases de errores, reduciendo así en número de clases de prueba que hay que desarrollar...”. (Pressman, 2010)

- **Técnicas de pruebas**

Son pruebas de caja negra que verifican la entrada y salida de los datos, se realizan en el sistema a través de casos de prueba. Las técnicas utilizadas para comprobar el correcto funcionamiento del sistema fueron partición equivalente para el requisito funcional Calcular la productividad de los autores (Ley de Lotka), el cual requiere datos de entrada; y gráfico de prueba para el requisito funcional Calcular Visibilidad o Impacto, quien no requiere datos de entrada.

La primera técnica: partición equivalente para el requisito funcional Calcular la productividad de los autores (Ley de Lotka).

Tabla 8. Caso de prueba de partición equivalente del RF2 Calcular la productividad de los autores (Ley de Lotka).

Escenario	Descripción	Datos	Respuesta del sistema	Flujo central
EC 1.1 Realizar una búsqueda.	El usuario selecciona el filtro o los filtros según la información que quiera obtener y accede a la página principal donde se muestra la ley de productividad de los autores o Ley de Lotka, y se pone la cantidad de publicaciones	V	El sistema muestra todas las publicaciones que cumplan con el criterio de búsqueda y la Ley de Lotka, devolviendo según el número de	El usuario realiza la búsqueda, escribe el número de publicaciones(n) y obtiene el número de autores con esas publicaciones.



	para los que se quieren conocer los autores.	números positivos	publicaciones que definió el usuario, el número de autores que tienen esas publicaciones.	
EC 1.1 Realizar una búsqueda.	El usuario selecciona el filtro o los filtros según la información que quiera obtener y accede a la página principal donde se muestra la ley de productividad de los autores o ley de Lotka, y se pone la cantidad de publicaciones para los que se quieren conocer los autores.	I Escribir letras o cualquier carácter que no sea número.	Muestra el mensaje "Debe entrar un número entero".	

La segunda técnica: gráfico de prueba para el requisito funcional Calcular Visibilidad o Impacto.

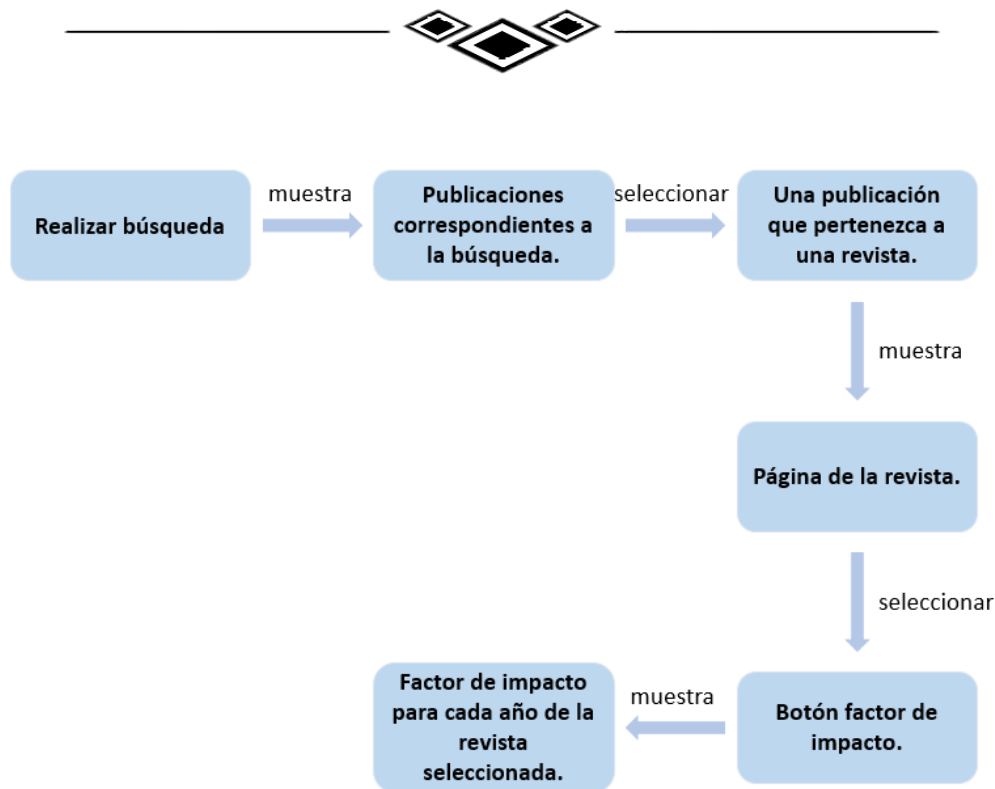


Figura 22. Gráfico correspondiente al caso de prueba del RF3 Calcular Visibilidad o Impacto.

3.5.4 Pruebas de sistema

Se realizaron pruebas de carga para 60 usuarios concurrentemente en un período de subida de 1,2 segundos, para los que se obtuvieron un 0.00% de errores.

Las pruebas fueron realizadas en una PC con requerimientos de 4GB de RAM y un procesador Dual Core. A continuación, se muestra el reporte de la prueba aplicada con la herramienta Jmeter.

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Media de Byt
Petición HTTP	243	2209	711	3640	830,55	0,00%	1,2/sec	1,05	168
Total	243	2209	711	3640	830,55	0,00%	1,2/sec	1,05	168



Figura 23. Prueba de carga

3.5.5 Pruebas de despliegue

“La prueba de despliegue, en ocasiones llamada prueba de configuración, ejercita el software en cada entorno en el que debe operar. Además, examina todos los procedimientos de instalación y el software de instalación especializado (por ejemplo, “instaladores”) que usarán los clientes, así como toda la documentación que se usará para introducir el software a los usuarios finales”. (Pressman, 2010)

Tabla 9. Pruebas de Despliegue

No	Hardware			
1	Procesador: Core 2 Duo	HDD: 200GB	RAM: 360MB	
2	Procesador: Dual Core	HDD: 500GB	RAM: 4GB	
3	Procesador: i3	HDD: 300GB	RAM: 4GB	
4	Procesador: i5	HDD: 500GB	RAM: 4GB	
5	Procesador: i7	HDD: 1T	RAM: 4GB	
Entorno a operar				
Navegador web	Sistema Operativo	Hardware	Resultado	Problemas
Mozilla Firefox Google Chrome Opera	Linux	1	Aceptado	Sí
		2,3,4,5	Aceptado	No
	Windows 7	1	Aceptado	Sí
		2,3,4,5	Aceptado	No
	Windows 8	1	Aceptado	Sí
		2,3,4,5	Aceptado	No

Para el caso 1, ocasionalmente, el sistema tardaba 2 minutos en responder. Por ello, las propiedades recomendadas son superior a 4GB de RAM.

3.6 Resultados de las pruebas

Las pruebas planteadas anteriormente han desarrolladas en varias iteraciones, algunos de los resultados son:

Tabla 10. Resultado de las pruebas.

Pruebas realizadas	Iteración	No conformidades
--------------------	-----------	------------------



Mostrar indicador de productividad	Iteración 1	Desarrollada satisfactoriamente.
Calcular la productividad de los autores (Ley de Lotka)	Iteración 1	Cuando el usuario introducía un valor en el campo no devolvía ningún resultado.
	Iteración 2	Cuando el usuario introducía letras, el sistema devolvía un resultado.
	Iteración 3	Desarrollada satisfactoriamente.
Calcular visibilidad o impacto	Iteración 1	Cuando el usuario daba click en el botón de Factor de Impacto el sistema no devolvía ningún resultado.
	Iteración 2	Desarrollada satisfactoriamente.
Mostrar índice h	Iteración 1	Desarrollada satisfactoriamente.
Mostrar índice g	Iteración 1	Desarrollada satisfactoriamente.
Mostrar índice i10	Iteración 1	Desarrollada satisfactoriamente.

3.7 Conclusiones parciales del capítulo

- Se desarrollan los diagramas de componente y de despliegue, donde se explican cada uno de los elementos asociados a ellos, los que permiten una mejor comprensión del nuevo módulo implementado.
- Se definen los estándares de codificación para la comprensión de próximos programadores.
- Se describieron las pruebas realizadas para verificar el correcto funcionamiento del módulo.
- Se mostraron los resultados obtenidos por las pruebas realizadas, donde se solucionaron todas las no conformidades.



CONCLUSIONES

Una vez culminada la presente investigación y el desarrollo del componente estadístico bibliométrico para la aplicación web RBC, se puede arribar a las siguientes conclusiones:

- El análisis de las principales tendencias relacionadas con los sistemas estadísticos bibliométricos existentes permitió determinar que las mismas no solucionaban el problema planteado, por lo que fue necesario el desarrollo de un nuevo módulo para el sistema RBC.
- La selección de las herramientas, tecnologías y metodologías de desarrollo contribuyó al desarrollo del módulo estadístico bibliométrico para el sistema RBC.
- Al realizar el diseño de la solución se obtuvieron las especificaciones del sistema, posibilitando la descripción y modelado del mismo para satisfacer los requerimientos identificados durante la etapa de análisis.
- Se desarrolló el módulo de información estadístico-bibliométrico para el Sistema RBC, atendiendo a cada una de las funcionalidades especificadas por el cliente.
- Las pruebas realizadas a la solución propuesta en la investigación garantizaron el cumplimiento de los diferentes atributos de calidad, lo que evidencia que el sistema cumple satisfactoriamente con los requisitos definidos.



RECOMENDACIONES

Una vez concluido el desarrollo del nuevo módulo para el sistema RBC a través del cumplimiento del objetivo propuesto en la presente investigación, se recomienda:

- Incorporar nuevas funcionalidades que permitan el cálculo de otros indicadores, tales como el impacto de las fuentes bibliográficas y la dispersión de las publicaciones científicas entre las diversas fuentes.
- Incorporar al sistema la posibilidad de creación y visualización de redes de co-autoría y co-citas, además de posibilitar la creación de mapas temáticos y/o de ciencia.



GLOSARIO

Informetría: la informetría es una disciplina de las ciencias de la información, está fundada en la combinación de técnicas avanzadas para la medición de la productividad de la información, permitiendo la recuperación de esta con estudios cuantitativos. Su objeto de estudio son los datos (información), la información social, que se obtiene y utiliza en todos los campos de la actividad del hombre.

Bibliometría: desarrolla modelos matemáticos que contribuyen con la toma de decisiones, sobre aspectos cuantitativos de la producción, disseminación y uso de la información, además de la bibliotecología. Es decir, se ocupa del análisis de las regularidades que ofrece el documento, los procesos y las actividades bibliotecarias.

Cienciometría: estudia los aspectos cuantitativos de la ciencia como disciplina o actividad económica, incluye, entre otras, las de publicación. Ella emplea, al igual que las otras dos disciplinas estudiadas, técnicas cuantitativas para la evaluación de la ciencia (el término ciencia se refiere, tanto a las ciencias naturales como a las sociales).



REFERENCIAS BIBLIOGRÁFICAS

Ardanuy, D. J. (2012). *Breve introducción a la bibliometría*. Barcelona.

Blanco., Y. V. (2014). *Módulo de Seguridad para los servicios de la plataforma de telecomunicación PLATEL*.

Brookes, B. (1990). Biblio-, Sciento-, Infor- Metrics? What are talking about?

Cañar, K. M., & Heredia, J. B. (2009). *Análisis, Diseño e Implementación de un Sistema Adaptativo de Recomendación de Información basado en Mashups*. Ecuador.

Condori, J. L. (2012). *Revista de Información, Tecnología y Sociedad*.

Dario, R. (20 de marzo de 2014). *Metodologías de Desarrollo Ágiles Vs. Metodologías Tradicionales*. Recuperado el 18 de abril de 2017, de <http://rdsoporteymantenimientodepc.blogspot.com/2014/03/metodologias-de-desarrollo-agiles-vs.html>

Duque, R. G. (1990). *Phyton para todos*. España.

Galindo, R. M. (2012). *Análisis, diseño e implementación de un sistema de información aplicado a la gestión educativa en centros de educación especial*. Peru.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design Patterns. Elements of Reusable Software*. (Addison-Wesley, Ed.) USA.

Grady Booch, J. R. (1999). *El Lenguaje Unificado de Modelado*.

Gutiérrez., J. J. (2014). *¿Qué es un framerwor web?*

Manz. (31 de enero de 2017). *Lenguaje CSS*. Obtenido de Lenguaje CSS: <https://lenguajecss.com/p/css/introduccion/que-es-css>

Maria Josefa Peralta González, M. F. (2015). *Criterios, clasificaciones y tendencias de los indicadores bibliométricos en la evaluación de la ciencia*.

Mikhailov, A. (1984). *Scientific communication and informatics*. Arlington: Information Resources Press.



- Moine, J. M. (2013). *Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo*.
- Mozilla Developer Network. (31 de enero de 2017). Obtenido de <https://developer.mozilla.org/es/docs/Web/JavaScript>
- Price, D. (1961). *Science since Babylon*.
- Pritchard, A. (1969). *Statistical bibliography or bibliometrics?*. *Journal of Documentation*. London.
- Rajan, T. (1985). *Informetrics: the concept and ramifications*.
- Sainero, G. C. (2017). *Hacia un concepto de bibliometría*. Madrid.
- Sanchez, T. R. (2014). *Metodología UCI*. La Habana.
- Schmuller, J. (2000). *Aprendiendo UML en 24 horas*. Mexico.
- Software, I. d. (2005). *Introducción al Modelo Conceptual*.
- Spinak, E. (1996). *Diccionario Enciclopédico de Bibliometría, Cienciometría e Informetría*.
- Spinak, E. (1998). *Indicadores cuantitativos*.
- Tague-Sutcliffe, J. (1992). *An introduction to informetrics*. *Information Processing & Management*.
- Torres, S. S. (2014). *Sistema de descubrimiento de bibliografía científica*. Barcelona.
- Wormell, I. (1998). *Infometría: explorand bases de dados comom instrumentos de análise*. Brazil.



ANEXOS

Anexo I. Preguntas formuladas al cliente durante la entrevista no estructurada

1. ¿Cómo funciona el sistema RBC?
2. ¿Cuáles son las nuevas características que debe tener el sistema?
3. ¿Qué funcionalidades se le agregarán al sistema?
4. ¿Qué información va a ser procesada?
5. ¿Quién utiliza el sistema RBC?
6. ¿Qué se considera un resultado correcto?
7. ¿Cuáles son las restricciones?
8. ¿Cuál es la forma de almacenamiento?



Anexo II. Descripción de los casos de uso del sistema

CU1: Mostrar índice de productividad.

Tabla 11. Descripción del CU: Mostrar índice de productividad.

Nombre	Mostrar índice de productividad.	
Objetivo	Determinar diversos datos de las publicaciones y los autores correspondientes.	
Actor(es)	Usuario	
Resumen	<p>Permite conocer información sobre los autores y las publicaciones de manera general, tales como:</p> <ul style="list-style-type: none"> - cuál es el autor, grupo o institución que más publicaciones realiza sobre un tema determinado. - el número de artículos realizados por un autor, grupo, institución o país. - el número de artículos publicados por las revistas en un período de tiempo determinado. - cantidad de artículos comprendidos por las principales áreas de investigación de una disciplina. - el número de relaciones establecidas entre diferentes autores, grupos, instituciones o países para la publicación de artículos. 	
Complejidad	Alta	
Prioridad	Media	
Precondiciones	Realizar filtrado según la información que se quiere conocer.	
Postcondiciones		
Flujo de eventos		
Flujo básico Mostrar índice de productividad.		
Actor	Sistema	
1.	Redirecciona a la página con las publicaciones que cumplan con el filtrado realizado. Muestra las publicaciones de mayor impacto según el campo del filtrado seleccionado.	
2.	En el borde superior izquierdo muestra el número de las publicaciones que devolvió el filtrado realizado. Lo que responde a cuál o cuáles son las publicaciones de mayor impacto en un área determinada (filtrando por el área seleccionada); el número de artículos realizados por un autor, institución o país, (filtrando por autor, organización, que representa la institución o país); y por último, la cantidad de artículos comprendidos por las principales áreas de investigación de una disciplina, (filtrando por área de investigación).	
3.	En el borde inferior derecho muestra el nombre de los autores que más hayan publicado según el resultado de las publicaciones, la organización y la unidad	



	<p>organizativa. Lo que responde a cuál o cuáles son los autores de mayor impacto en un área determinada de la investigación (filtrando por el área de investigación) y a cuál es el autor, grupo e institución que más publicaciones realiza sobre un determinado tema (filtrando por palabras claves).</p>
<p>Prototipo de interfaz gráfica para el CU: Mostrar indicador de productividad.</p>	
<p>This paper describes a simple protocol, the remotely keyed encryption protocol (rkep), that enables a secure, but bandwidth-limited, cryptographic smartcard to function as a high bandwidth secret-key encryption and decryption engine for an insecure, but fast, host processor. The host processor assumes most of the computational and bandwidth burden of each cryptographic operation without ever learning the secret key stored on the card. By varying the parameters of the protocol, arbitrary...</p> <p>Matt Blaze http://www.princeton.edu/~rblee/ELE572Papers/Fall04Readings/HWEncryptionSmartCards_Blaze.pdf</p> <p>[artículo] <u>Modelo para la extensión de las capacidades de procesamiento y memoria de tarjetas inteligentes Java Card</u> (Citaciones: 0)</p> <p>Las tarjetas inteligentes poseen características distintivas como la portabilidad, dado su tamaño reducido y el bajo costo, para poder ser usadas a gran escala. Asociadas a estas características se encuentran las limitaciones de los recursos de hardware que poseen, relacionadas fundamentalmente con la capacidad de memoria y procesamiento de estos dispositivos. Estas y otras limitaciones propias de la tecnología Java Card, constituyen limitantes significativas para los desarrolladores de aplicaciones para tarjetas inteligentes.</p> <p>Susana María Ramírez Brey, Yusnier Valle Martínez, Adonis Cesar Legon Campo http://rci.uci.edu/index.php?journal=rci</p> <p>[artículo] <u>Portable digital radio communications-an approach to tetherless access</u> (Citaciones: 2)</p> <p>The evolution of current technologies that provide either wireless exchange access or access and communications to people away from their own telephones is outlined. A description is given of a proposed use of digital radio technology as a drop/distribution/loop technology for local exchange access and the integration of this radio technology with network intelligence to provide the added functionality of portability.</p>	<p>Autor más publicado: <i>Li-fung Chang</i></p> <p>Organizaciones más publicadas: <i>Universidad de las Ciencias Informáticas</i> <i>University of Pennsylvania 2</i> <i>Karlsruher Institute of Technology 2</i> <i>Department of Mathematics 2</i> <i>Karlsruher Institute of Technology</i></p> <p>Unidades organizativas más publicadas: <i>AT&T Bell Laboratories</i> <i>IEEE Press Piscataway</i> <i>Desarrollo de aplicaciones</i></p>

CU3: Calcular visibilidad o impacto.

Tabla 12. Descripción del CU: Calcular visibilidad o impacto.

Nombre	Calcular visibilidad o impacto.
Objetivo	Determinar el factor de impacto.
Actor(es)	Usuario
Resumen	El factor de impacto de una revista determinada se calcula por año. Para ello se cuentan las citas que reciben durante dicho año todos los documentos publicados en la revista en los dos años anteriores. El número total de citas es el numerador. Acto seguido, se cuentan todos los "ítems citables"



	publicados en la revista en dichos años y ya tenemos el denominador. El factor de impacto se calcula dividiendo el numerador entre el denominador.	
Complejidad	Alta	
Prioridad	Media	
Precondiciones	Realizar búsqueda.	
Postcondiciones		
Flujo de eventos		
Flujo básico Calcular visibilidad o impacto.		
Actor	Sistema	
1.	Redirecciona a la página con las publicaciones correspondientes a la búsqueda.	
2. Selecciona una publicación que pertenezca a una revista.		
3.	Redirecciona a la página de la revista.	
4. Click en el botón factor de impacto.		
5.	Muestra el factor de impacto para cada año de la revista seleccionada.	
Flujo alternativo Calcular visibilidad o impacto.		
1.2 Realizar búsqueda filtrando por el nombre de la revista.		
2. Click en el botón Factor de impacto.		
3.	Muestra el factor de impacto para cada año de la revista seleccionada.	
Prototipo de interfaz gráfica para el CU: Calcular visibilidad o impacto.		



CU4: Mostrar índice h.

Tabla 13. Descripción del CU: Mostrar índice h.

Nombre	Mostrar índice h.
Objetivo	Determinar el índice h de determinado autor.
Actor(es)	Usuario
Resumen	Se calcula con base en la distribución de las citas que han recibido los trabajos científicos de un investigador. Para determinarlo se deben ordenar los artículos de un autor o de un grupo, por número de veces que han sido citados de forma descendente, e ir transitando la lista hasta encontrar la última publicación cuyo número correlativo sea menor o igual que el número de citas (ese número correlativo es el factor h). Para mejor comprensión, si el factor h vale n, entonces n publicaciones han sido citadas más de n veces.
Complejidad	Media
Prioridad	Alta



Precondiciones	Realizar búsqueda
Postcondiciones	
Flujo de eventos	
Flujo Básico Mostrar visibilidad e impacto.	
Actor	Sistema
1.	Redirecciona a la página con todas las publicaciones de ese autor.
2.	Muestra el índice h.

Prototipo de interfaz gráfica para el CU: Mostrar índice h.

The screenshot displays a user profile for Vyacheslav M. Abramov. Key statistics are shown on the right: Total de citas: 2, Índice h: 1 (highlighted with a green box and arrow), Índice i10: 0, and Índice g: 1. Below the statistics is a line graph titled 'Reporte de las Publicaciones del Autor' showing the number of publications over time.

CU5: Mostrar índice g.

Tabla 14. Descripción del CU: Mostrar índice g.

Nombre	Mostrar índice g.
Objetivo	Determinar el índice g de determinado autor.
Actor(es)	Usuario



Resumen	Un autor tiene un índice g cuando, considerando los g artículos más citados de dicho autor, la cantidad de citas acumuladas por estos g artículos es superior a g al cuadrado, en caso de que la cantidad de citas acumuladas por los g artículos no sea superior a g al cuadrado; el índice g es 0.
Complejidad	Media
Prioridad	Alta
Precondiciones	Realizar la búsqueda.
Postcondiciones	
Flujo de eventos	
Flujo básico Mostrar índice g.	
Actor	Sistema
1.	Redirecciona a la página con todas las publicaciones de ese autor.
2.	Muestra índice g.
Prototipo de interfaz gráfica para el CU: Mostrar índice g.	
	

CU6: Mostrar índice i10.



Tabla 15. Descripción del CU: Mostrar índice i10.

Nombre	Mostrar índice i10.
Objetivo	Determinar índice i10 de determinado autor.
Actor(es)	Usuario
Resumen	Se calcula con base a la distribución de las citas que han recibido los trabajos científicos de un investigador; el mismo recorre las publicaciones que se han citado al menos 10 veces.
Complejidad	Media
Prioridad	Alta
Precondiciones	
Postcondiciones	
Flujo de eventos	
Flujo Básico Mostrar visibilidad e impacto.	
Actor	Sistema
1.	Redirecciona a la página con todas las publicaciones de dicho autor.
2.	Muestra el índice i10.

Prototipo de interfaz gráfica para el CU: Mostrar índice i10.





Anexo III. Diagrama de clases del diseño utilizando estereotipos web

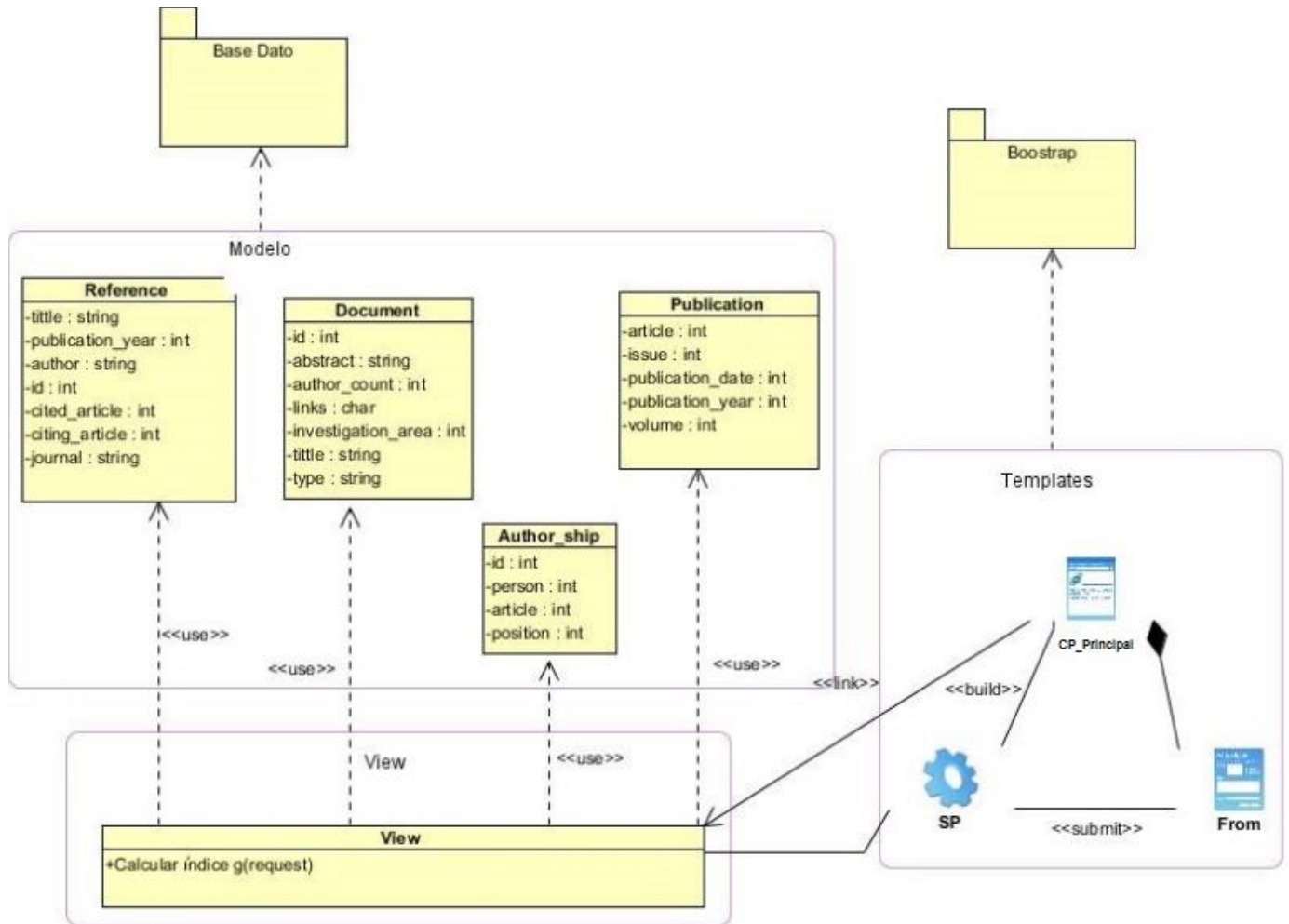


Figura 24. Diagrama de clases del diseño utilizando estereotipos web, Mostrar índice g.



Anexo IV. Pruebas de validación

Tabla 16. Caso de prueba de partición equivalente del RF1 Mostrar índice de productividad.

Escenario	Descripción	Datos	Respuesta del sistema
EC 1.1 Realizar una búsqueda.	El usuario selecciona el filtro o los filtros según la información que quiera obtener y accede a la página principal donde se muestran todas las publicaciones correspondientes al filtrado .	V	El sistema muestra todas las publicaciones que cumplan con el criterio de búsqueda, en el borde superior izquierdo muestra la cantidad de publicaciones correspondientes, y en el borde inferior derecho muestra el nombre de los autores que más hayan publicado según el resultado de las publicaciones, la organización y la unidad organizativa.
		Los datos entrados por el usuario tienen que estar en la base de datos.	
EC 1.2 Realizar una búsqueda.	El usuario selecciona el filtro o los filtros según la información que quiera obtener y accede a la página principal donde se muestran las publicaciones correspondientes.	I	Muestra el mensaje "No se encontraron resultados".
		Entra datos que no se encuentran en la base de datos.	



Tabla 17. Caso de prueba de partición equivalente del RF4,5,6 Mostrar índices h, Mostrar índice g, Mostrar índice i10.

Escenario	Descripción	Datos	Respuesta del sistema
EC 4.1 Realizar una búsqueda.	El usuario selecciona el filtro o los filtros según la información que quiera obtener y accede a la página principal donde se muestran todas las publicaciones correspondientes al filtrado. Selecciona el autor del que quiera conocer los índices.	V Los datos entrados por el usuario tienen que estar en la base de datos.	El sistema muestra todas las publicaciones que cumplan con el criterio de búsqueda y después de la selección del autor muestra los índices i10, h y g.
EC 4.2 Realizar una búsqueda.	El usuario selecciona el filtro o los filtros según la información que quiera obtener y accede a la página principal donde se muestran las publicaciones correspondientes.	I Entra datos que no se encuentran en la base de datos.	Muestra el mensaje "No se encontraron resultados".