

Diseño de bases de datos relacionales con DBDStudio

Jesús M. Jorge Santiso¹, Pavel Nichita²

(1) Departamento de Ingeniería Informática y de Sistemas

(1)(2) Grupo de Optimización y Minería de datos

Universidad de La Laguna

San Cristóbal de La Laguna

jjorge@ull.edu.es, pnichita@ull.edu.es

Resumen

En este artículo se presenta DBDStudio, una herramienta software implementada en formato aplicación web. El objetivo de la misma es la de asistir en el análisis y diseño de bases de datos relacionales que utilicen condiciones de integridad de tipo dependencia funcional. Entre otras, la aplicación permite calcular las claves, el cierre de un conjunto de atributos, el recubrimiento minimal, detectar y normalizar hasta las formas normales de 3FN (tercera forma normal) y FNBC (forma normal de Boyce y Codd). Pensamos que la herramienta puede resultar valiosa para las personas relacionadas con el proceso de enseñanza/aprendizaje del diseño de bases de datos relacionales, facilitándoles a los estudiantes la comprensión de las principales nociones teóricas gracias a los diversos algoritmos implementados.

Abstract

In this paper we present DBDStudio, a software tool, developed as a web-app. The goal is to assist in the analysis and design of relational databases based on functional dependencies. Among others, the app allows to compute keys, closure of attribute sets, minimal cover, detect and normalize till 3NF (third normal form) and BCNF (Boyce and Codd normal form). We think this tool is valuable as a teaching resource for who are involved in the training of relational database design. The tool provides great help for the students, making easier learning the theoretical notions thanks to the different implemented algorithms.

Palabras clave

Bases de datos relacionales, teoría de la normalización, formas normales, aplicación web.

1. Introducción

El diseño de las bases de datos constituye una etapa fundamental y costosa en tiempo, en el desarrollo e implementación de cualquier sistema de información. Efectivamente, un buen diseño simplificará enormemente las operaciones de actualización, minimizará la redundancia e inconsistencia de los datos y hará que todo el sistema en general sea mucho más robusto. Sin embargo, alcanzar estos objetivos no resulta trivial. En el caso de las bases de datos relacionales el diseño está fuertemente condicionado por el tipo de condiciones de integridad consideradas. Cuando todas éstas son del tipo dependencias funcionales y/o plurales, el proceso de diseño ha sido profundamente estudiado y es susceptible de ser altamente automatizado.

En este proyecto nos planteamos desarrollar una herramienta software de carácter docente, que permita asistir a los estudiantes en las operaciones más importantes relacionadas con el análisis y diseño académico de bases de datos relacionales utilizando condiciones de integridad del tipo dependencias funcionales. La herramienta será implementada como una aplicación web que cargará los atributos y dependencias funcionales de una relación dada y, a partir de esta información permitirá, entre otras funcionalidades, calcular sus claves, el cierre de un conjunto de atributos, el recubrimiento minimal, aplicar comprobaciones para FNBC, 3FN, ... así como los algoritmos de descomposición en 3FN y FNBC.

La herramienta está siendo utilizada como material de apoyo en la asignatura de Bases de Datos de la Escuela de Ingeniería y Tecnología (ESIT) de la Universidad de La Laguna (ULL). La asignatura se imparte en el tercer curso del grado, es obligatoria, tiene 6 créditos ECTS y es parte del bloque Fundamentos Tecnológicos de Ingeniería Informática. Actualmente hay matriculados en la asignatura algo más de 100 alumnos, los cuales pueden utilizar la aplicación web a través del aula virtual de la misma. El uso principal de la herramienta por parte de los

alumnos consiste en confirmar que los cálculos realizados a mano por ellos son correctos. Pensamos que con dichas verificaciones los alumnos pueden asimilar de forma más eficaz los conceptos teóricos de la teoría de normalización. Para mejorar la aplicación y profundizar en el proceso de aprendizaje, al final del artículo proponemos una serie de mejoras que fueron extraídas de un cuestionario que ha sido cumplimentado por los alumnos al finalizar la asignatura.

2. Estado del arte

La teoría de la normalización para el diseño de bases de datos relacionales fue propuesta inicialmente por E. F. Codd en 1970. En el artículo [2] se define el modelo relacional y la primera forma normal (1FN).

En 1971 E.F. Codd amplía su teoría de la normalización definiendo la segunda y tercera forma normal (2FN y 3FN) [3]. Las formas normales presentadas no estaban libres de problemas de redundancia, por lo que en 1974 publica junto con R.F. Boyce una forma normal adicional llamada Forma Normal de Boyce y Codd (FNBC), que también es conocida como 3FN más restrictiva [1].

Desde que E.F. Codd expuso su teoría de la normalización, su idea ha seguido desarrollándose y ampliándose. Así, R. Fagin propuso la cuarta forma normal (4FN) [8]. Otros autores han llegado a dar otras definiciones a las formas normales, como C. Zaniolo a la 3FN [11]. Actualmente existen artículos sobre quinta forma (5FN), forma normal de dominio-clave e incluso, la sexta forma normal (6FN) [4].

Para aprovechar eficazmente todas las capacidades de recuperación y procesamiento de la información del hardware actual, los sistemas de información deben tener un diseño eficiente. La búsqueda de esta eficiencia conduce a esquemas de bases de datos complejos. Efectivamente, obtener un diseño que cumpla con los requisitos de no duplicar datos de forma innecesaria y que esté libre de anomalías de inserción, actualización y borrado no es tarea trivial. Con el fin de aliviar esta situación, a lo largo del tiempo se han ido desarrollando herramientas software que asisten en el análisis y diseño de la base de datos. Algunas de ellas están orientadas a docencia, como apoyo teórico y práctico para profesores y/o alumnos, mientras otras son proyectos de investigación con conectores a SGBD¹ profesionales como Oracle² para crear las relaciones resultantes de un proceso de normalización. Las herramientas pueden ser de tipo aplicaciones de escritorio o bien aplicaciones web. Sin embargo, muchas de las herramientas

han tenido poco mantenimiento y, por lo tanto, algunas directamente no están disponibles o no se pueden ejecutar por problemas de compatibilidad. En nuestra revisión hemos analizado las siguientes herramientas:

- *Tutorial de Bases de Datos*³ por Alberto Fernández Villán.
- *Learn DataBase Normalization*⁴ [9] por Nikolay Georgiev en la Universidad de Umeå, Suecia.
- *Reno*⁵ por Julia Martínez Pérez, distribuido con el libro “*Diseño de Bases de Datos Relacionales*” (De Miguel, Piattini y Marcos, Ra-Ma, 1999).
- *Normalization Tool*⁶ por Griffith University, Brisbane, Australia.
- *RDBNorma* [5] por Y. B. Dongare, P. S. Dhabe y S. V. Deshmukh, Wishwakarma Institute of Info.Techonology, Pune, India.
- *Micro* [6] por H. Du y L. Wery.

El Tutorial de Bases de Datos está disponible en un 50%. Los conceptos teóricos que acompañan a la herramienta siguen presentes, pero el applet Java no está disponible. En su momento las funcionalidades de la herramienta eran cálculo de un cierre de un conjunto de dependencias funcionales, cálculo del cierre de un conjunto de atributos, cálculo de las claves de una relación, cálculo del recubrimiento minimal, comprobación de forma normales: 2FN, 3FN y FNBC y descomposición hasta 3FN y FNBC.

Learn DataBase Normalization es una herramienta web que ayuda a comprobar los conocimientos sobre la teoría de normalización de forma fácil y eficiente. Es la única que tiene funcionalidades para profesores. Ellos pueden crear ejercicios que los alumnos deben resolver y comparar el resultado introducido por el profesor con el obtenido por los alumnos. Las funcionalidades de la herramienta son: cálculo de claves de una relación, cálculo del recubrimiento minimal de un conjunto de dependencias funcionales, descomposición de una relación en 2FN, 3FN y FNBC.

Reno es una aplicación de escritorio que permite elegir entre dos conjuntos de algoritmos: los nuevos (mejorados, menor tiempo de ejecución) y los tradicionales. Al introducir las dependencias funcionales se detectan y eliminan aquellas que son triviales y se descomponen las dependencias, pero algunas formas de dependencias triviales no se detectan. Las operaciones que puede realizar la aplicación son: cálculo de claves, cierre de un conjunto de atributos y cálculo del recubrimiento minimal. También comprueba la

³ Web de la herramienta:

www.aic.uniovi.es/~quevedo/docencia/dsi/TutorialNormalizacion/index.htm

⁴ <http://ldbnonline.com/Ldbn.html>

⁵ Web donde se puede descargar el ejecutable: <http://www.inf-cr.uclm.es/www/fruiz/bda/software.htm#Reno>

⁶ Web donde se encuentra la herramienta:

<http://www.ict.griffith.edu.au/~jw/normalization/ind.php>

¹ https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_s_de_datos

² https://en.wikipedia.org/wiki/Oracle_Database

forma normal en que está una relación y si es necesario se pueden realizar descomposiciones por análisis o síntesis. La aplicación no soporta sistemas operativos de 64 bits. Si se consigue utilizar un SO de 32 bits sorprende lo lenta que es. Es bastante vieja y con poco mantenimiento, por lo que concluimos que es una herramienta ya desfasada.

Normalization tool es una aplicación web que permite normalizar desde 2FN hasta FNBC, calcular el recubrimiento minimal, las claves candidatas y comprobar diversas formas normales. Para cada algoritmo la herramienta tiene la posibilidad de mostrar paso a paso que es lo que está haciendo. Esta opción resulta muy adecuada desde el punto de vista docente, pues proporciona mucha más información a los alumnos que la utilizan. De todas las herramientas analizadas ésta es la más moderna y nueva.

Micro es un prototipo para demostrar la posibilidad de automatizar la teoría de la normalización. Implementa un algoritmo para normalizar hasta 2FN propio y también algoritmos hasta 3FN y FNBC. Los autores consideran la normalización como un proceso de un solo paso, por lo tanto, la interacción se limita a introducir datos, especificar la forma normal y sacar los resultados. Es una de las pocas que permite exportar a un SGBD (Microsoft Access). Debido a la fecha de su publicación (1999), se le puede considerar obsoleta.

RDBNorma es una respuesta a *Micro*, que es más eficiente desde el punto de vista computacional. Su limitación es que normaliza hasta 3FN y no permite dependencias funcionales con más de 4 atributos en el antecedente (lado izquierdo de la dependencia). Igual que *Micro* permite exportar las relaciones normalizadas a un SGBD, en este caso Oracle.

De las 6 herramientas enumeradas, sólo 3 están disponibles al público, de las cuales una (RENO) tiene poca compatibilidad con los S.O. actuales. Las dos herramientas restantes (*Learn DataBase Normalization* y *Normalization tool*) cumplen con el propósito de analizar y normalizar una base de datos relacional hasta la forma normal de Boyce y Codd, pero en las dos se echan en falta operaciones adicionales básicas en el análisis como, por ejemplo, la equivalencia entre dos conjuntos de dependencias funcionales, la inferencia de dependencias funcionales, la proyección de un conjunto de dependencias funcionales sobre un conjunto de atributos, etc.

Por todo ello creemos que está justificado el desarrollo de una nueva herramienta en este campo que intente paliar las carencias detectadas.

3. DBDStudio

Al evaluar el estado de arte se ha visto la necesidad de disponer de una aplicación propia (DBDStudio⁷) que sirva como recurso docente en las asignaturas de Bases de Datos universitarias relacionadas con la teoría de la normalización. La herramienta debe proporcionar más funcionalidades/operaciones que las actualmente disponibles en aplicaciones similares y ser un incentivo para los alumnos que piensan que la teoría de la normalización es un tema muy teórico, poco práctico y oscuro de entender.

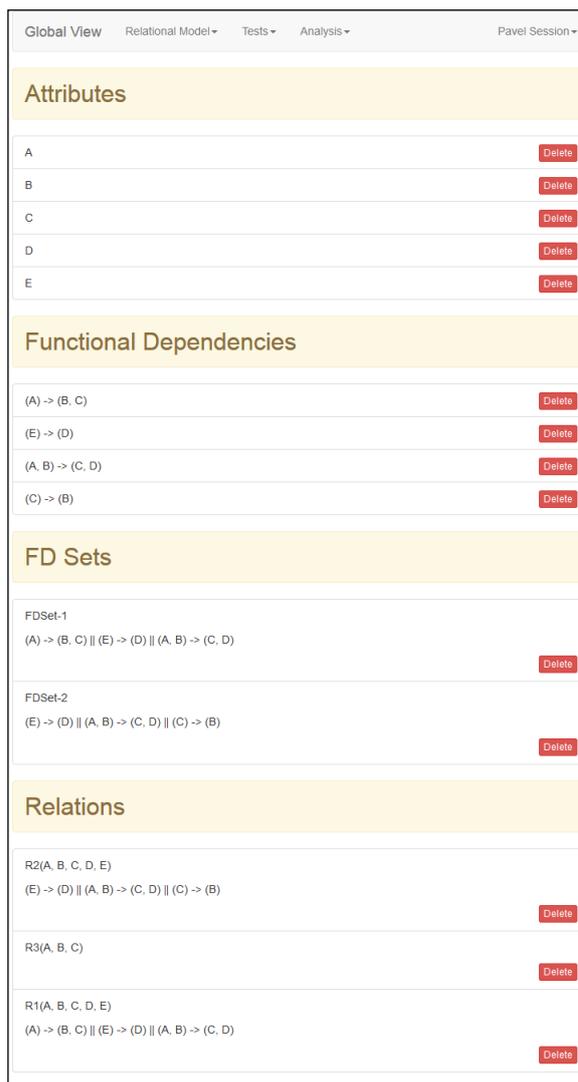


Figura 1: DBDStudio *Global View* - Vista de datos introducidos

Para introducir datos en DBDStudio se utiliza un proceso ordenado. El primer paso consiste en introducir los atributos, seguido por introducir las dependencias funcionales, los conjuntos de dependencias funcionales y al final las relaciones.

⁷ <http://cc.etsii.ull.es/odn>

En cualquier momento se puede añadir nueva información al problema (atributos, dependencias funcionales, ...). El usuario puede ver los datos introducidos mediante el menú Global View (Figura 1).

3.1. Operaciones esenciales

Las operaciones más importantes que se han incorporado en DBDStudio son:

- Normalización hasta tercera forma normal y forma normal de Boyce y Codd, comprobando si se han perdido dependencias funcionales durante el proceso.
- Calculo de las claves candidatas de una relación.
- Calculo del recubrimiento minimal de un conjunto de dependencias funcionales.
- Detección de la forma normal de una relación.
- Comprobaciones explícitas si una relación está en 2FN, 3FN o forma normal de Boyce y Codd.

Los algoritmos de normalización han sido ampliamente descritos en la literatura. En particular para la descomposición de 3FN y FNBC hemos utilizado los algoritmos descritos en [7], p. 481, y en [7], p. 480 respectivamente. Para el cálculo de las claves se utiliza el algoritmo de Ullman descrito en [10], p. 234. Para el cálculo de recubrimiento minimal se ha implementado el algoritmo descrito en [10], p. 236.

En la Figura 2 se puede observar la interfaz de usuario para normalizar una relación y en la Figura 3 el resultado de ello.

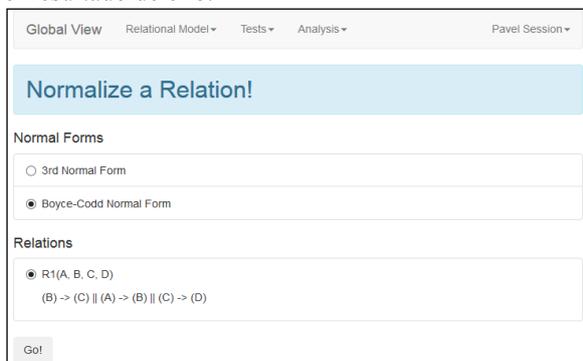


Figura 2: DBDStudio - Selección de la relación y forma normal para normalizar

3.2. Operaciones adicionales

Durante el desarrollo de los algoritmos y estructuras de datos para la automatización de la teoría de normalización, se han implementado varios procedimientos que sirven de apoyo a las operaciones esenciales de normalización, pero que no están disponibles en la mayoría de las herramientas analizadas. Se considera que estas operaciones adicionales son de gran ayuda para los alumnos porque se pueden ejecutar de forma independiente y permiten entender mejor

los conceptos relevantes. Entre ellas destacamos por su exclusividad y/o interés las siguientes:

- Cálculo del cierre de un conjunto de atributos sobre una relación (algoritmo de Ullman).
- Proyección de un conjunto de dependencias funcionales sobre un conjunto de atributos (utilizado al descomponer una relación).
- Comprobar si un conjunto de dependencias funcionales es minimal.
- Comprobar si un atributo o conjunto de atributos es clave o superclave de una relación.
- Comprobar si dos conjuntos de dependencias funcionales son equivalentes.
- Comprobar si a partir de un conjunto de dependencias funcionales se puede inferir otro conjunto de dependencias funcionales.
- Comprobar si una dependencia funcional puede ser inferida a partir de un conjunto de dependencias funcionales.

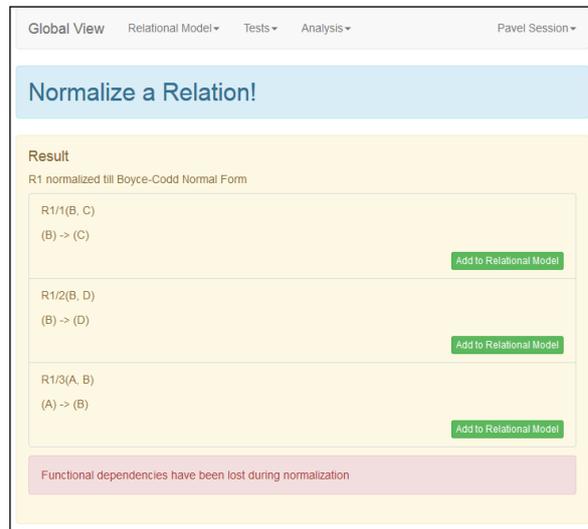


Figura 3: DBDStudio - Resultado de normalizar

4. Desarrollo de la herramienta

Para el desarrollo de la aplicación web DBDStudio se han utilizado una amplia variedad de herramientas, lenguajes de programación y de marcado. Entre ellas caben destacar:

- Java 8⁸ y Oracle JDK⁹
- Spark¹⁰ (no Apache Spark¹¹)
- JUnit¹²
- Cobertura¹³
- SonarQube¹⁴

⁸ <https://go.java/index.html>

⁹ <http://www.oracle.com/technetwork/java/javase/downloads/>

¹⁰ <http://sparkjava.com/>

¹¹ <https://spark.apache.org/>

¹² <http://junit.org>

¹³ <http://cobertura.github.io/cobertura/>

- Javadoc¹⁵
- Git¹⁶ y GitHub¹⁷
- Jenkins¹⁸
- JavaScript¹⁹ y Bootstrap²⁰
- Docker²¹ y Docker Hub²²

El código del actual proyecto está disponible de forma abierta a todo el mundo. Se puede añadir código al mismo, corregirlo o crear un nuevo proyecto propio independiente. El proyecto se puede descargar en los siguientes enlaces:

- <https://github.com/nichitapavel/project>
- <https://github.com/nichitapavel/spark-project>

El código fuente se puede utilizar de forma libre mientras se respete la licencia de éste: Mozilla Public License Version 2.0²³.

La imagen contenerizada de la aplicación web también es accesible al público para su uso. El actual proyecto está disponible en Docker Hub²⁴ y se puede descargar y ejecutar sin restricciones.

5. Experiencia docente

Para evaluar la experiencia docente de la herramienta se ha propuesto a los alumnos de la asignatura de Bases de Datos del curso académico 2017/18 un cuestionario semi-cerrado. El formato nos permite controlar las variables de más interés. Con ello se pretende llegar a conocer qué percepción tienen los alumnos de la aplicación. Es decir, qué visión tienen sobre su uso, sobre su aporte didáctico, sobre su interfaz, los tiempos dedicados a su uso, si detectan carencias de funcionalidades y/o operaciones, ... Existen preguntas con respuesta libre donde el alumno tiene libertad para expresar lo positivo y negativo de la herramienta, esto nos permite conocer aquella información cualitativa que servirá de base para futuras mejoras de la aplicación. Al ser el primer año que se utiliza la aplicación lo estamos enfocando más como una prueba de concepto, intentando conocer si existe potencial para una herramienta de tales características. Los resultados de la encuesta se toman con cautela y nos sirven de incentivo para continuar el desarrollo. La encuesta se puede consultar en el siguiente enlace ²⁵.

De las operaciones ofrecidas, el 100% de los alumnos encuestados afirma que las han utilizado todas. En las primeras posiciones en frecuencia de uso se sitúan las operaciones de normalización hasta 3FN y BCFN junto con el cálculo/comprobación de las claves de una relación. Les siguen las operaciones de cálculo/comprobación del recubrimiento minimal de un conjunto de dependencias funcionales. Llama la atención que dos de los algoritmos esenciales para la aplicación como son el de *Ullman* y el de *proyección de un conjunto de dependencias funcionales sobre un conjunto de atributos* no son muy utilizados por los alumnos. Quizá sea debido a que tienen claro el funcionamiento de los mismos, aunque este hecho merece un estudio más detallado.

A las preguntas relacionadas con la herramienta como recurso docente, la gran mayoría han estado de acuerdo que es una herramienta que aporta mucho en el aprendizaje de la teoría de la normalización. Según se desprende de los resultados, la herramienta ha ayudado a entender mejor y a aprender más rápidamente la teoría de la normalización. Las respuestas en este sentido han resultado muy alentadoras.

La interfaz de usuario es donde más críticas se han recibido. En líneas generales, a la interfaz se le considera sencilla y fácil de utilizar, pero el proceso de introducir datos es considerado lento y tedioso. Las preguntas sobre el tiempo de uso han confirmado este problema, donde alrededor del 50% de los alumnos encuestados afirman haber dedicado la mayor parte del tiempo de uso a la introducción de datos y relaciones. Esta debilidad debe ser mejorada para hacer más atractiva la herramienta.

En cuanto a la funcionalidad, los alumnos están de acuerdo en que la herramienta está muy completa, pero echan de menos una opción para encontrar atributos raros en una dependencia particular y poder calcular el recubrimiento minimal de un conjunto de dependencias funcionales de forma más controlada (seleccionando los atributos raros a eliminar y el orden de las dependencias a estudiar).

Parece que las funcionalidades asociadas a dependencias plurales los alumnos no las ven tan necesarias. Habrá que analizar el motivo.

En general, el resultado de la encuesta ha sido bastante positivo, pues la totalidad de los alumnos participantes en la misma considera útil y volvería a utilizar la herramienta, e incluso la recomendaría a otras personas en proceso de aprendizaje de bases de datos.

6. Conclusiones

En este trabajo hemos presentado la herramienta DBDStudio para asistir en la enseñanza universitaria de la teoría de la normalización de bases de datos relacionales.

¹⁴ <https://www.sonarqube.org/>

¹⁵ <http://www.oracle.com/technetwork/articles/java/index-137868.html>

¹⁶ <https://git-scm.com/>

¹⁷ <https://github.com/>

¹⁸ <https://jenkins-ci.org/>

¹⁹ <https://www.javascript.com/>

²⁰ <https://getbootstrap.com/>

²¹ <https://www.docker.com/>

²² <https://hub.docker.com/>

²³ <https://www.mozilla.org/en-US/MPL/2.0/>

²⁴ <https://hub.docker.com/r/pavelnichita/pavel-tfm-spark/>

²⁵ <https://goo.gl/forms/aCnTmbV5FV4ejw5K2>

Se trata de una aplicación web que permite a los alumnos aprender y reforzar los principales conceptos teóricos y algorítmicos de forma intuitiva, a través de la resolución de problemas.

En su primer año la herramienta ha sido utilizada con cierto éxito en la asignatura de Bases de Datos de la ESIT, ULL. A los alumnos que la han usado se les ha pasado un cuestionario sobre ella y, en líneas generales, los comentarios recibidos han sido positivos. La herramienta es considerada por el alumnado como útil y bastante completa.

El cuestionario reveló determinadas deficiencias en la herramienta relativas a la interfaz de usuario y a la carencia de algunas opciones concretas (posibilidad de realizar trazas de los algoritmos, selección de dependencias funcionales al descomponer, ...), apartados en los que tendremos que trabajar para seguir mejorándola.

Además, queremos extender las pruebas y algoritmos de la aplicación para que también puedan manipular dependencias plurales.

Agradecimientos

Los autores quieren agradecer a los revisores anónimos de JENUI sus comentarios, los cuales han contribuido a mejorar el trabajo presentado.

Referencias

- [1] R. F. Boyce and E. F. Codd, *Recent investigations into relational Data base systems*, IBM Research Report RJ1385, 1974.
- [2] E. F. Codd, *A relation model of data for large shared data banks*, Communications of the ACM Volume 13, 377-388, 1970.
- [3] E. F. Codd, *Further normalization of the data base relational model*, IBM Research Report RJ909, 1971.
- [4] C. J. Date, H. Darwen, N. A. Lorentzos, *Temporal data and the relational model: A detailed investigation into the application of interval and relation theory to the problem of temporal database management*. Oxford: Elsevier, Morgan Kaufman, 1st Edition 172-177, 176, 2003.
- [5] Y. V. Dongare, P. S. Dhabe, S. V. Deshmukh, *RDBNorma: - A semi-automated tool for relational database schema normalization up to third normal form*, CoRR, abs/1103.0633, 2011.
- [6] H. Du y L. Wery, *Micro: A normalization tool for relational database designers*, Journal of network and computer application, Vol.22, pp.215-232, 1999.
- [7] R. Elmasri y S. B. Navathe, *Fundamentos de sistemas bases de datos*, Ed 3, Addison Wesley, 2002.
- [8] R. Fagin, *Multivalued dependencies and a new normal form for relational databases*, ACM Transactions on Database Systems Volume 2, 262-278, 1977.
- [9] N. Georgiev, *Visualization of functional dependencies in a web environment*, UMNAD, 833, 2010.
- [10] A. Silberschatz, H.F. Korth, S. Sudarshan, *Fundamentos de bases de datos*, Ed. 4, McGraw-Hill, 2002
- [11] C. Zaniolo, *A new normal form for the design of relational database schemata*, ACM Transactions on Database Systems Volume 7, 489-499, 1982.