

DISEÑO DE UNA APLICACIÓN QUE PERMITA NORMALIZAR A TERCERA FORMA  
NORMAL, APLICANDO EL ALGORITMO DE BERNSTEIN

FABIO JULIAN BERNAL  
LUIS GUILLERMO CORREA  
CRISTHIAN LEONARDO TOLOSA

GRUPO 2

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS  
ESPECIALIZACIÓN EN INGENIERÍA DE SOFTWARE  
FACULTAD DE INGENIERÍA  
BASES DE DATOS  
BOGOTÁ  
2014

# Tabla de contenido

Introducción.....	3
Objetivo.....	3
Alcance.....	3
Diagrama de contexto.....	3
Método de desarrollo de software.....	4
Diseño de Arquitectura.....	5
Restricciones funcionales.....	9
Referencias.....	9

# **1 Introducción**

Este documento presenta una propuesta de solución a la implementación del algoritmo de Bernstein o una mejora que propone un esquema en tercera forma normal (3FN).

## **2 Objetivo**

Construir una solución haciendo uso del algoritmo de Bernstein o una mejora del mismo para entregar un esquema en tercera forma normal basado en un conjunto de atributos y un conjunto de dependencias funcionales.

## **3 Alcance**

El alcance de este documento se refiere a la especificación de la arquitectura, solución y definición de la metodología a utilizar para desarrollar el proyecto.

## **4 Diagrama de contexto**

El siguiente diagrama de casos de uso representa las funcionalidades básicas que el sistema va a satisfacer. La principal funcionalidad del sistema es permitir a un usuario normalizar en tercera forma normal a partir de un conjunto de atributos y dependencias funcionales.

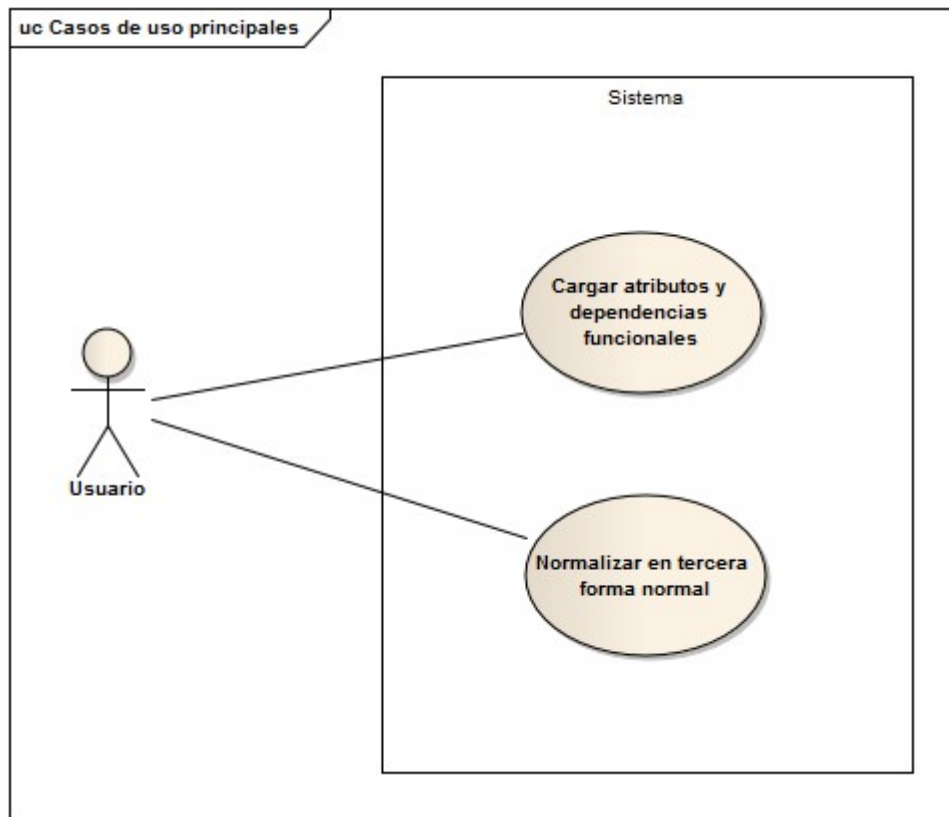


Figura 1. Diagrama de casos de uso principales

## 5 Método de desarrollo de software

Para la realización de este proyecto se utilizará el método Kanban [1] [2] una técnica relativamente nueva para manejar procesos de desarrollo de software de manera altamente efectiva. Sustentado en el sistema de producción de Toyota “Just in time” (JIT). En este acercamiento, el proceso, desde la definición de tareas hasta la entrega al usuario se encuentra visible a todo el equipo para que ellos puedan seleccionar actividades de la cola.

En su forma más simple, Kanban consiste en un gran tablero con cartas o sticky notes colocadas en columnas. Las cartas representan ítems de trabajo y como fluyen a través del proceso de desarrollo representado por las columnas.

A continuación se muestra un ejemplo de un tablero Kanban y como fluyen las tareas a través del proceso.

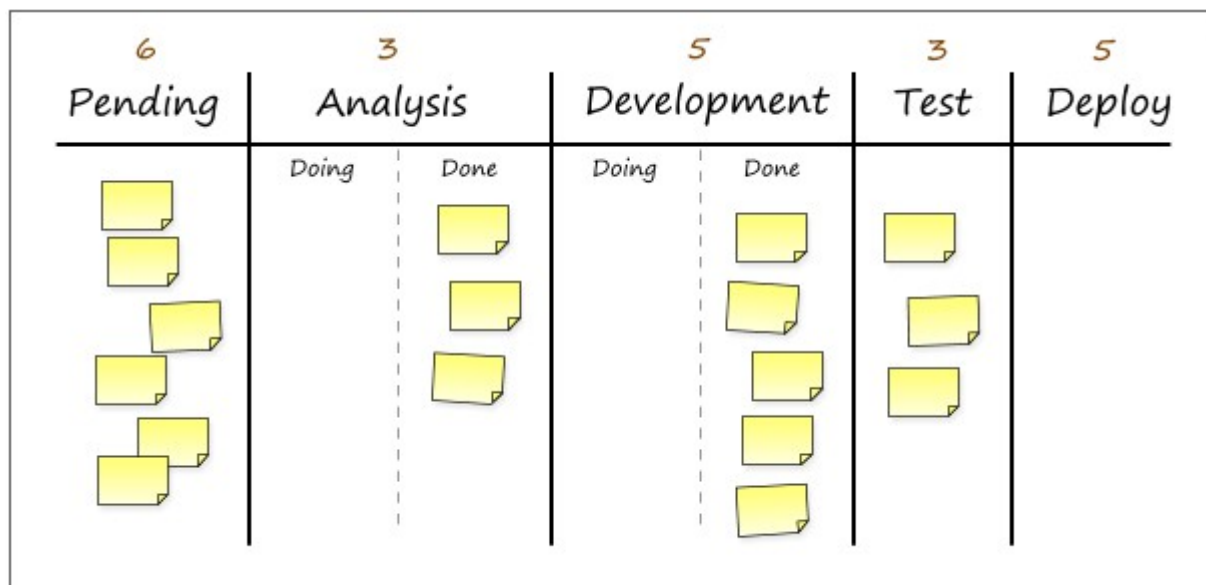


Figura 2. Ejemplo de tablero Kanban. Tomado de <http://kanbanblog.com/explained/>.

## 6 Diseño de Arquitectura

Para esta solución se realizará una aplicación Standalone el cual se ejecutará en modo de aplicación de escritorio, contará con una interfaz gráfica de usuario que le permitirá al usuario realizar los casos de uso principales, el aplicativo tendrá una separación lógica de componentes para cada una de las responsabilidades.

A continuación se muestran diagramas en notación UML describiendo el diseño de arquitectura.

### 6.1 Modelo de componentes

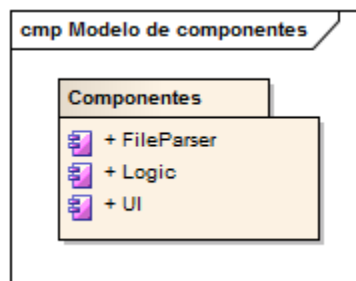


Figura 3. Modelo de componentes

Para la implementación de la solución se seleccionó la tecnología JAVA que permite desarrollar aplicaciones independiente de la plataforma o sistema operativo, ofrece un modelo orientado a objetos robusto que permite diseñar adecuadamente la aplicación, en rendimiento el kit de desarrollo estándar (JDK) y el lenguaje ofrecen un Garbage Collector que recicla los objetos en memoria no utilizados.

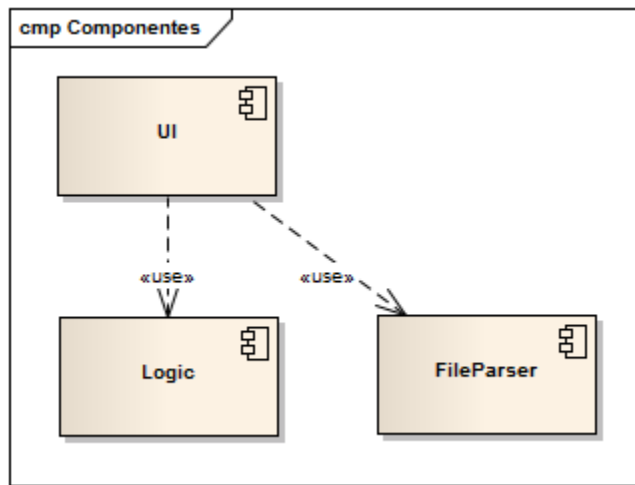
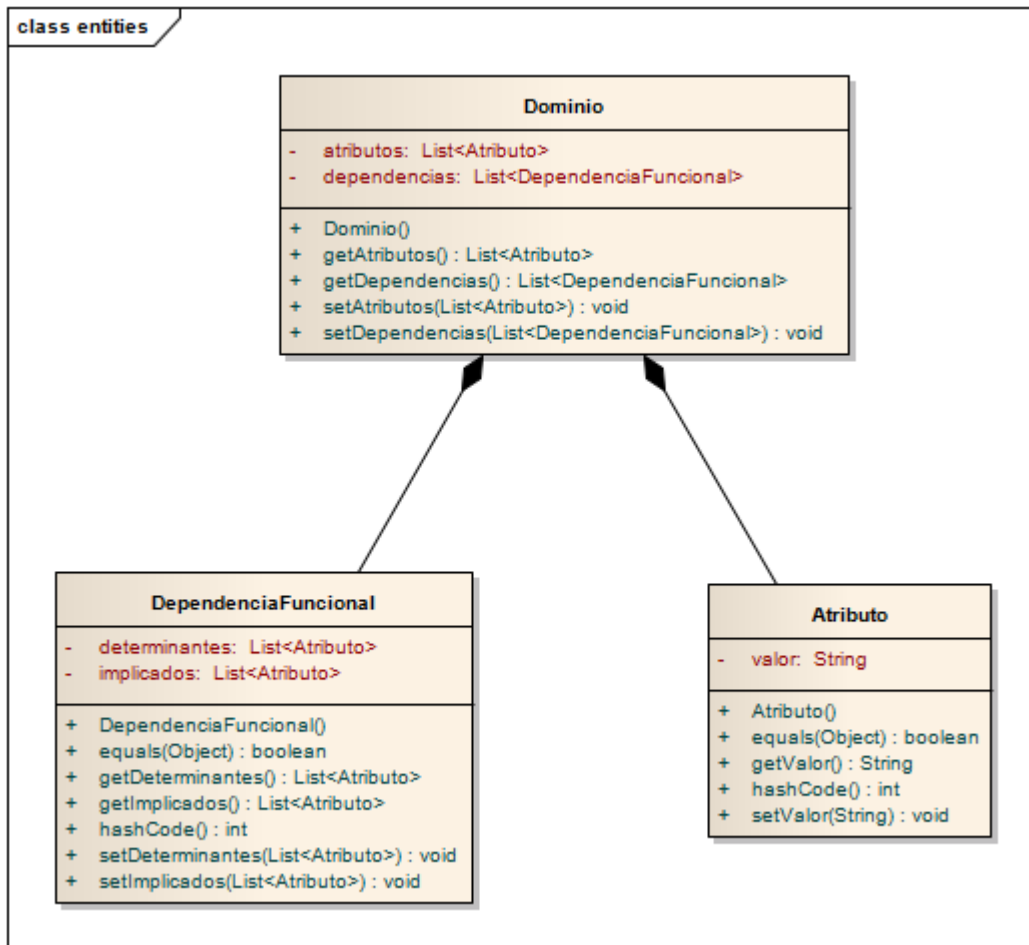


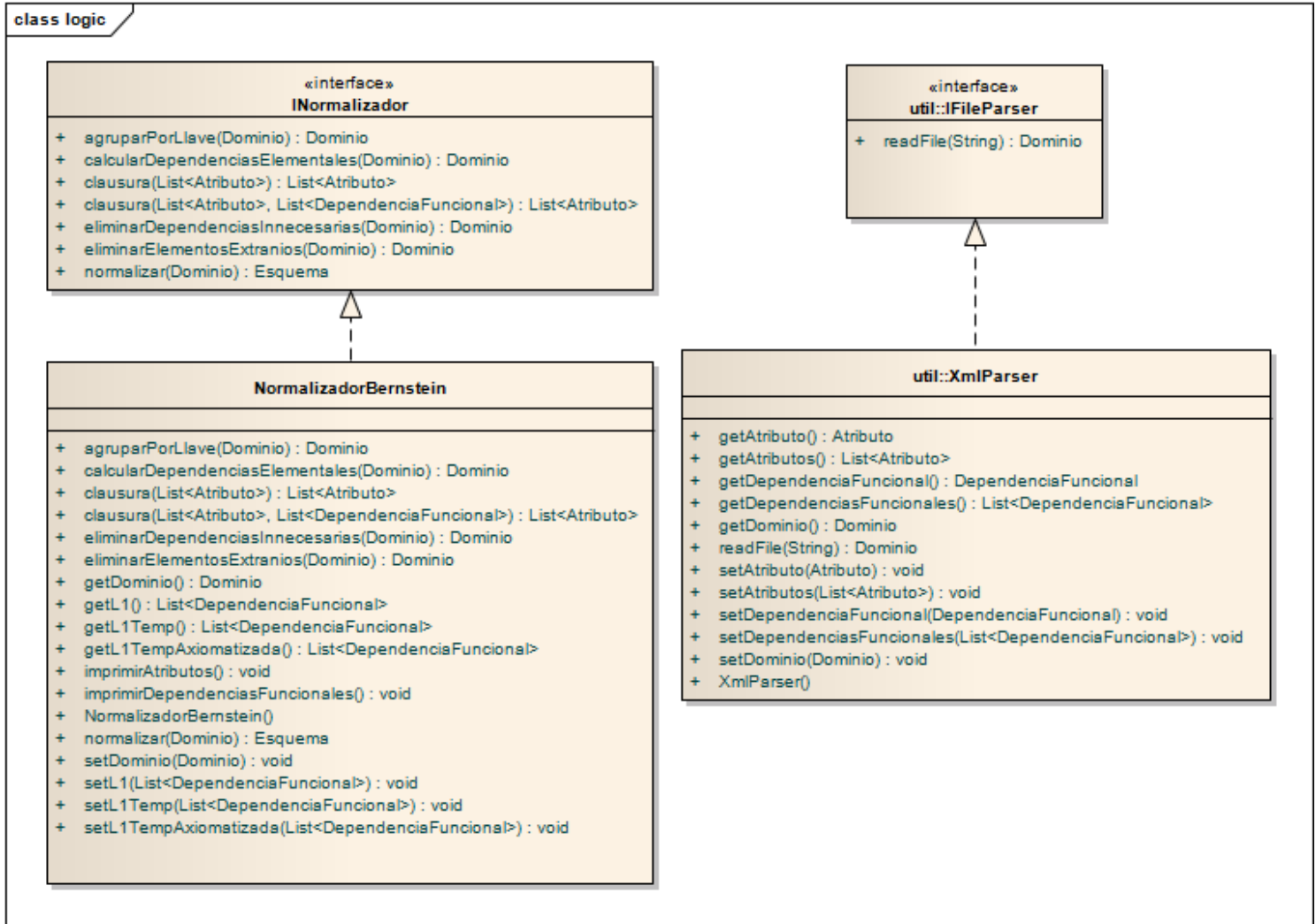
Figura 4. Diagrama de componentes

## 6.2 Modelo de clases

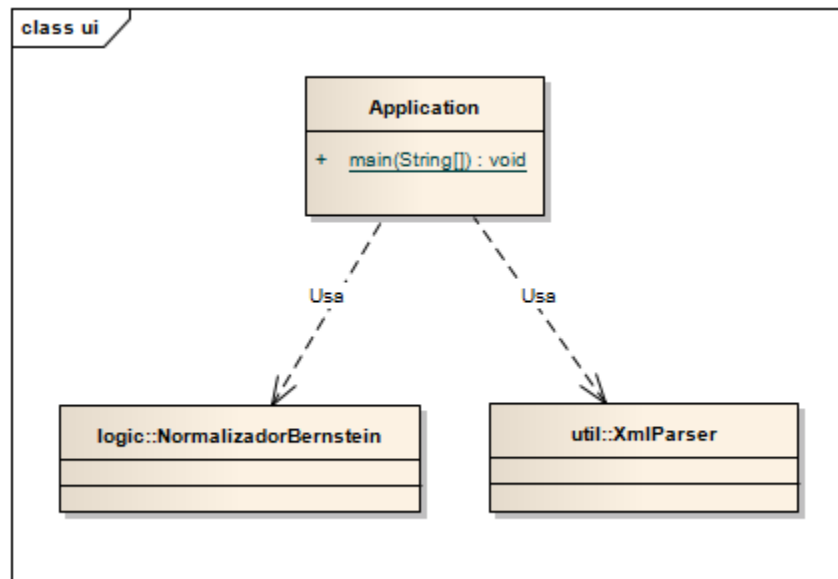
Modelo de clases de entidades.



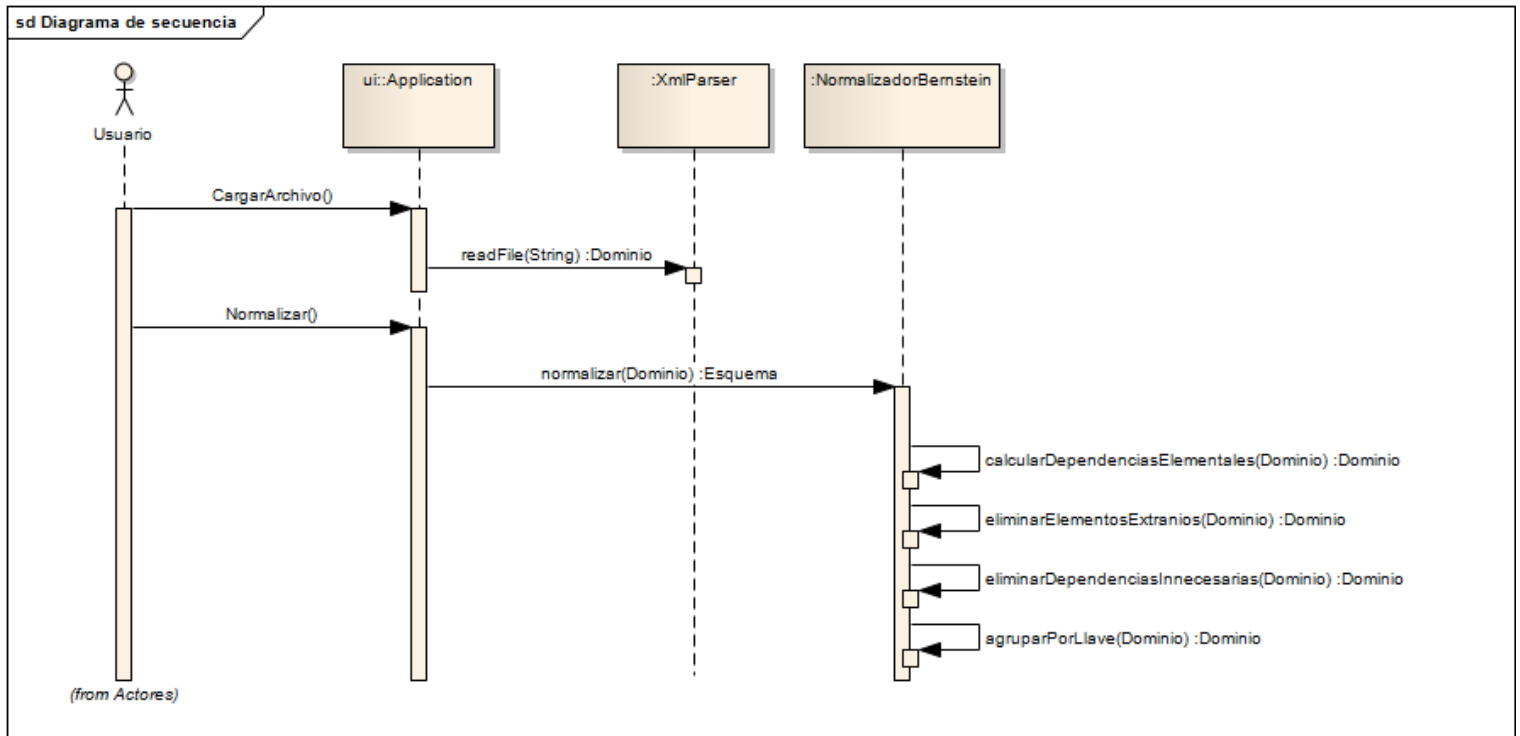
## Modelo de clases lógica de negocio.



## Modelo de clases interfaz de usuario.



## 6.3 Modelo de secuencia





## 7 Restricciones funcionales

La entrada de datos se realizará mediante un archivo xml en el cual se especifican el conjunto de atributos de entrada y el conjunto de dependencias funcionales, a continuación se muestra un posible esquema del archivo xml.

```
<Root>
  <Attributes>
    <Attribute>A</Attribute>
    <Attribute>B</Attribute>
    <Attribute>C</Attribute>
    <Attribute>D</Attribute>
  </Attributes>
  <FDs>
    <FD>
      <Left>
        <Attribute>A</Attribute>
        <Attribute>B</Attribute>
      </Left>
      <Rigth>
        <Attribute>C</Attribute>
      </Rigth>
    </FD>
    <FD>
      <Left>
        <Attribute>C</Attribute>
      </Left>
      <Rigth>
        <Attribute>D</Attribute>
      </Rigth>
    </FD>
  </FDs>
</Root>
```

## 8 Referencias

- [1] “What is Kanban?” [Online]. Available: <http://kanbanblog.com/explained/>. [Accessed: 22-Sep-2014].
- [2] “What is Kanban? | LeanKit.” [Online]. Available: <http://leankit.com/kanban/what-is-kanban/>. [Accessed: 22-Sep-2014].