

# **Sistema de Administración de Farmacias**

## **Informe de Revisión Técnica Formal (RTF)**

### **Versión 1.0**

### **Historia de revisiones**

Fecha	Versión	Descripción	Autor
09/11/2014	1.0	Elaboración	SQA

# Contenido

<b>1. PRODUCTO REVISADO .....</b>	<b>3</b>
1.1. NOMBRE Y VERSIÓN DEL PRODUCTO REVISADO .....	3
1.2. PARTICIPANTES DE LA REVISIÓN .....	3
1.3. TÉCNICA UTILIZADA .....	3
<b>2. OBJETIVOS DE LA RTF.....</b>	<b>3</b>
<b>3. PROBLEMAS DETECTADOS .....</b>	<b>4</b>
3.1. PROBLEMA DETECTADO 1: NOMBRES DE OPERACIONES NO ESTANDARIZADOS.....	4
3.1.1. <i>Sugerencia de corrección</i> .....	4
3.2. PROBLEMA DETECTADO 2: FALTA DE COMENTARIOS EN EL CÓDIGO. ....	4
3.2.1. <i>Sugerencia de corrección</i> .....	4
3.3. PROBLEMA DETECTADO 3: FALTA DE REUTILIZACIÓN DE CÓDIGO. ....	4
3.3.1. <i>Sugerencia de corrección</i> .....	4
3.4. PROBLEMA DETECTADO 4: CANTIDAD DE LÍNEAS MUY GRANDE EN ALGUNOS CASOS.....	4
3.5. PROBLEMA DETECTADO 5: GRANDES PARTES DE CÓDIGO NO UTILIZADAS (COMENTADAS) .....	5
3.5.1. <i>Sugerencia de corrección</i> .....	5
3.6. PROBLEMA DETECTADO 6: COMENTARIOS INFORMALES .....	5
3.6.1. <i>Sugerencia de corrección</i> .....	5
<b>4. EVALUACIÓN.....</b>	<b>6</b>
4.1. ESTADO ACTUAL DEL PRODUCTO. ....	6
4.2. ACCIONES A TOMAR. ....	6

# **1. Producto revisado**

## **1.1. Nombre y Versión del Producto revisado**

El producto revisado es el código de la aplicación en su versión 2.1.

## **1.2. Participantes de la revisión**

SQA – Alfonso Methol.

## **1.3. Técnica utilizada**

La técnica utilizada es la revisión exhaustiva de cada archivo de la aplicación, asegurando que cumpla tanto con los estándares de implementación establecidos, así como los criterios de calidad relacionados a la facilidad para la continuación del desarrollo por parte del cliente.

# **2. Objetivos de la RTF**

La revisión tiene como objetivo analizar si el código cumple las pautas de implementación definidas al comienzo del proyecto. Para esto se revisara cada archivo que conforma la aplicación chequeando que cumpla dichas pautas. El objetivo es asegurar que la aplicación este lo suficientemente clara para la continuación del desarrollo por parte del cliente. Los puntos clave más revisados serán el formato del código, nombres de operaciones y variables, el uso de comentarios y la correcta reutilización de código.

## 3. Problemas detectados

### 3.1. Problema Detectado 1: Nombres de operaciones no estandarizados.

Si bien cumplen en parte con el estándar definido aparecen operaciones en un mismo archivo con "get", "obtener", "devolver" así como "set" y "asignar" entre otros. No es un error que genere confusión pero sería más correcto utilizar una sola forma.

#### 3.1.1. Sugerencia de corrección

La sugerencia de corrección sería definir una forma y aplicar solo esa. De todas formas creo que el riesgo de realizar este cambio es muy grande como para aplicarlo a esta altura del proyecto.

### 3.2. Problema Detectado 2: Falta de comentarios en el código.

El estándar sugería que antes de cada operación se comentara sus entradas así como su objetivo, esto permite que alguien ajeno al código tenga un mayor y más rápido entendimiento del mismo. Este criterio no se aplicó en gran parte del código.

#### 3.2.1. Sugerencia de corrección

Dar prioridad a las partes claves que requieran una mejor descripción y realizarla.

### 3.3. Problema Detectado 3: Falta de reutilización de código.

Hay varios códigos que básicamente hacen lo mismo salvo algún detalle y como fueron realizados por implementadores distintos en vez de realizar un único código que cumpliera con todos se realizaron varios.

#### 3.3.1. Sugerencia de corrección

Unificar ese tipo de código. Igual se considera que el riesgo de realizar este tipo de corrección a esta altura del proyecto es muy grande.

### 3.4. Problema Detectado 4: Cantidad de líneas muy grande en algunos casos.

Algunos archivos superan las 1700 líneas de código, lo que dificulta en parte su modificación a futuro.

### **3.5. Problema Detectado 5: Grandes partes de código no utilizadas (comentadas)**

Algunos archivos presentan código ya no utilizado y comentado. En algunos casos es un porcentaje alto en relación a las líneas de código totales del archivo.

#### **3.5.1. Sugerencia de corrección**

Eliminar esas líneas de código que ya no son utilizadas.

### **3.6. Problema Detectado 6: Comentarios informales**

Hay comentarios que deberán ser eliminados ya que se realizaron para comunicar cierta información entre implementadores. Todos los comentarios deberán cumplir con cierto grado de formalidad así como ser directos y concisos.

#### **3.6.1. Sugerencia de corrección**

Eliminar los comentarios innecesarios y formalizar algunos otros.

## 4. Evaluación

### 4.1. Estado actual del producto.

El producto se encuentra sufriendo sus últimas modificaciones, el objetivo del tiempo que queda es mejorar y verificar lo existente. Si bien el código no cumple con ciertos aspectos establecidos, en general el formato es muy bueno y permite un correcto entendimiento del mismo. La estructura es clara y permite un fácil acceso a los diferentes elementos de la aplicación.

### 4.2. Acciones a tomar.

Se buscara mejorar los aspectos más fundamentales que impacten en el entendimiento y modificación futura del código.