

**Universidad de Guayaquil**  
**Facultad de Ciencias Matematicas y Fisicas**  
Proyecto de Implantación de un Sistema de Gestión de la Calidad ISO  
9001:2015 en el Software SINFIG  
**Integrantes:**  
Ramirez Abarca Alex  
Ramirez Rios Edinson  
Nicole Navarrete Briones  
Curso: SOF-S-MA-3-2

**Tabla de Contenido**

<b>1. Normas ISO</b>	_____
<b>2. Caso de Estudio</b>	_____
<b>3. Planificación</b>	_____
<b>4. Código Fuente</b>	_____
<b>5. Conclusión</b>	_____

**1. NORMAS ISO**

Las normas internacionales hacen que las cosas funcionen. Proporcionan especificaciones de clase mundial para productos, servicios y sistemas, para garantizar la calidad, seguridad y eficiencia. Son fundamentales para facilitar el comercio internacional. Normas ISO 9001 ISO 9001: 2015 especifica los requisitos para un sistema de gestión de calidad cuando una organización: a) necesita demostrar su capacidad para proporcionar constantemente productos y servicios que cumplan con los requisitos legales y reglamentarios aplicables del cliente, y b) tiene como objetivo mejorar la satisfacción del cliente a través de la aplicación efectiva del sistema, incluidos los procesos para la mejora del sistema y el aseguramiento de la conformidad con el cliente y los requisitos legales y reglamentarios aplicables. Todos los requisitos de ISO 9001: 2015 son genéricos y están destinados a ser aplicables a cualquier organización, independientemente de su tipo o tamaño, o los productos y servicios que proporciona.

**2. CASO DE ESTUDIO**

Sinfig es un paquete de animación 2D basado en vectores. Está diseñado para ser capaz de producir animaciones con calidad de largometraje. Elimina la necesidad de interpolación, evitando la necesidad de dibujar a mano cada cuadro. Synfig presenta independencia de resolución espacial y temporal (nítida y suave a cualquier resolución o velocidad de fotogramas), imágenes de alto rango dinámico y un sistema de plugin flexible. Synfigstudio es el estudio de animación para synfig y proporciona la GUI interfaz para crear animaciones synfig que se guardan en synfig .sif o formato .sifz.

### **Características**

- Produce animaciones de calidad cinematográficas
- Mantiene una resolución independiente mente del territorio
- Tiene una independencia de resolución temporal
- Es orientada a un diseño de Artista
- Las animaciones se trabajan por capas

### **Determinación de los requisitos**

#### **Requisitos Funcionales**

- El software producirá animaciones utilizando vectores e ilustraciones de mapas de bits
- El software deberá tener una amplia paleta de herramientas para mejorar la producción de animaciones
- El software debe exportar los proyectos en archivos .mp4
- El software debe proporcionar mensajes informativos, de ayuda y de error cuando existan
- El software debe contener un módulo de ayuda para los usuarios

#### **Requisitos No Funcionales**

- • El acceso al software solo debe ser cambiado por el administrador
- El software debe estar disponibles para plataformas Windows, OS X y Linux
- El software debe ocupar pocos recursos del sistema
- El software debe permitir el uso de periféricos de entrada

### **Compatibilidad de estándares del mercado**

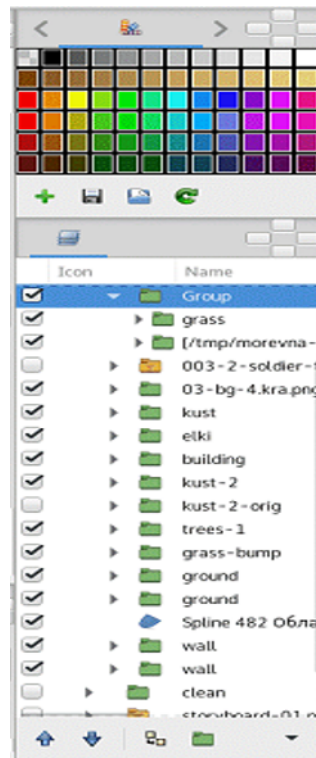
ISO 20252

### **Diseño y Desarrollo**

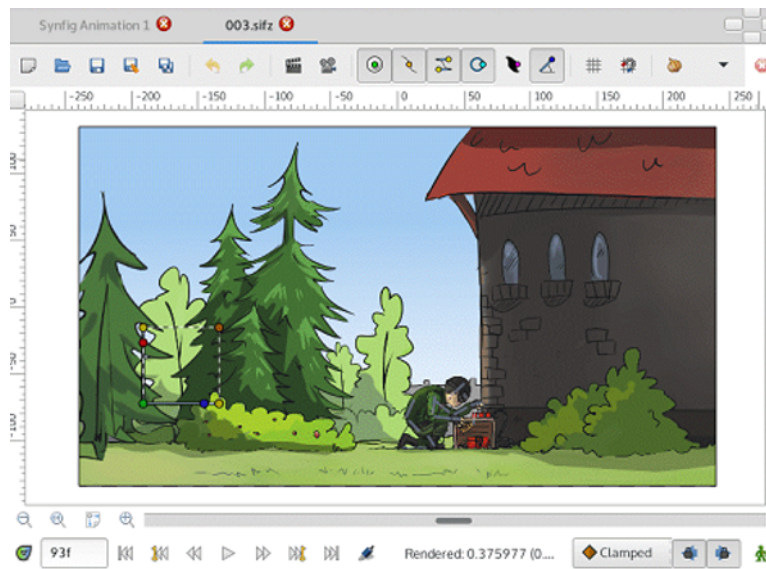
El software implementa las siguientes interfaces:



Barra de Herramientas



Ventanas de Lienzo-Barra de Navegación

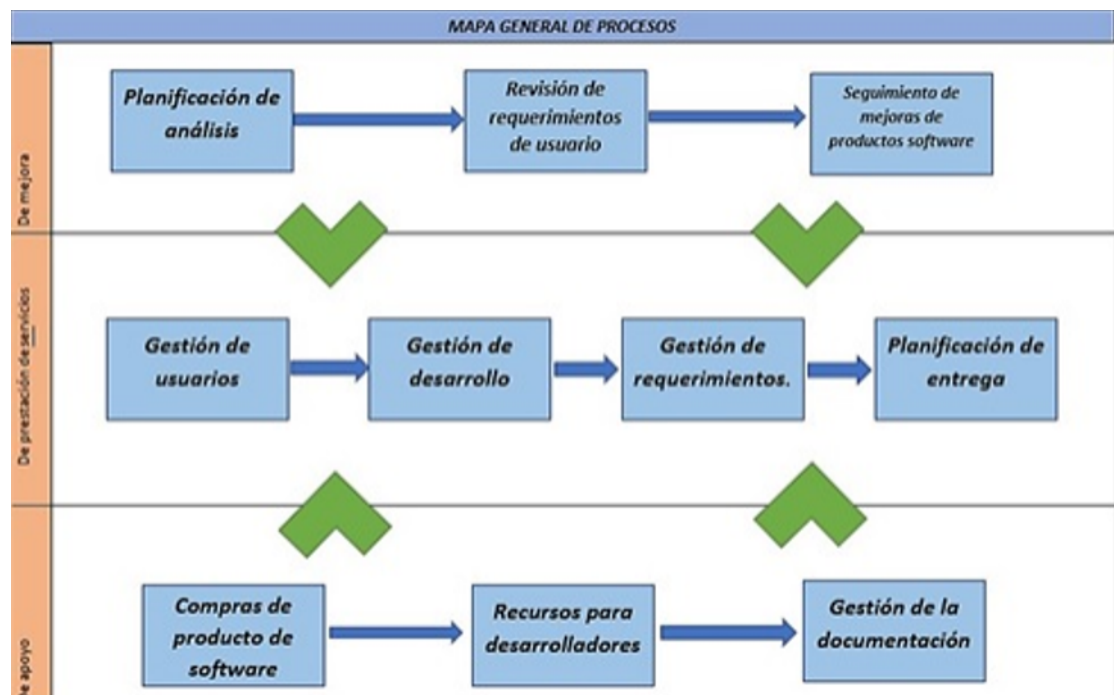


Línea de Tiempo

### 3. PLANIFICACIÓN

Al planificar el sistema de gestión de la calidad, Synfig. tiene en consideración los riesgos y oportunidades que es necesario abordar con el fin de:

- a) Asegurar que el sistema de gestión de calidad puede lograr los resultados previstos.
- b) Aumentar los efectos deseables.
- c) Prevenir o reducir efectos no deseados.
- d) Lograr la mejora.



Mapa general de procesos

### 4. CÓDIGO FUENTE

```

23
24 /* === H E A D E R S ===== */
25
26 #ifndef USING_PCH
27 #    include "pch.h"
28 #else
29 #ifndef HAVE_CONFIG_H
30 #    include <config.h>
31 #endif
32
33 #include <synfig/general.h>
34
35 #include "app.h"
36 #include <iostream>
37 #include "ipc.h"
38 #include <stdexcept>
39
40 #include <gui/localization.h>
41 #include <glibmm/convert.h>
42
43 #ifdef _WIN32
44 #include "main_win32.h"
45 #endif
46
47 #endif
48
49 /* === U S I N G ===== */
50
51 using namespace std;
52 using namespace etl;
53 using namespace synfig;

```

```

51 using namespace std;
52 using namespace etl;
53 using namespace synfig;
54 using namespace studio;
55
56 /* === M A C R O S ===== */
57
58 /* === G L O B A L S ===== */
59
60 /* === P R O C E D U R E S ===== */
61
62 /* === M E T H O D S ===== */
63
64 /* === E N T R Y P O I N T ===== */
65
66 int main(int argc, char **argv)
67 {
68
69 #ifdef _WIN32
70     if (consoleOptionEnabled(argc, argv))
71     {
72         redirectIOToConsole();
73     }
74     else
75     {
76         // QuickHack: to avoid strange bug with stderr
77         freopen("NUL", "w", stdout);
78         freopen("NUL", "w", stderr);

```

```

74     else
75     {
76         // QuickHack: to avoid strange bug with stderr
77         freopen("NUL", "w", stdout);
78         freopen("NUL", "w", stderr);
79         freopen("NUL", "r", stdin);
80         ios::sync_with_stdio();
81     }
82 #endif
83
84     String binary_path = synfig::get_binary_path(String(argv[0]));
85
86 #ifdef ENABLE_NLS
87     String locale_dir;
88     locale_dir = etl::dirname(etl::dirname(binary_path))+ETL_DIRECTORY_SEPARATOR+"share"+ETL_DIRECTORY_SEPARATOR+"locale";
89     setlocale(LC_ALL, "");
90     bindtextdomain(GETTEXT_PACKAGE, Glib::locale_from_utf8(locale_dir).c_str() );
91     bind_textdomain_codeset(GETTEXT_PACKAGE, "UTF-8");
92     textdomain(GETTEXT_PACKAGE);
93 #endif
94     {
95         SmartFILE file(IPC::make_connection());
96         if(file)
97         {
98             cout << endl;
99
100             cout << " " << _("synfig studio is already running") << endl << endl;
101             cout << " " << _("the existing process will be used") << endl << endl;
102
103             // Hey, another copy of us is open!
104             // don't bother opening us, just go ahead and
105             // tell the other copy to load it all up
106             if (argc>1)
107                 fprintf(file.get(),"F\n");
108
109             while(--argc)
110                 if((argv)[argc] && (argv)[argc][0]!='-')
111                     fprintf(file.get(),"O %s\n",etl::absolute_path((argv)[argc]).c_str());
112
113             fprintf(file.get(),"F\n");
114
115             return 0;
116         }
117     }

```

```

116     }
117
118     cout << endl;
119     cout << " " << _("synfig studio -- starting up application...") << endl << endl;
120
121     try
122     {
123         studio::App app(etl::dirname(binary_path), &argc, &argv);
124
125         app.run();
126     }
127     catch(int ret)
128     {
129         std::cerr<<"Application shutdown with errors ("<<ret<<')<<std::endl;
130         return ret;
131     }
132     catch(string& str)
133     {
134         std::cerr<<"Uncaught Exception:string: "<<str.c_str()<<std::endl;
135         throw;
136     }
137     catch(std::exception& x)
138     {
139         std::cerr<<"Standard Exception: "<<x.what()<<std::endl;
140         throw;
141     }
142     catch(Glib::Exception& x)

```

## 5. CONCLUSIÓN

Podemos concluir que la importancia de implementar un sistema de gestión de la calidad, radica en el hecho de que sirve de plataforma para desarrollar en la organización una serie de actividades, procesos y procedimientos, encaminados a lograr que las características del producto o del servicio cumplan con los requisitos del cliente, que en pocas palabras sean de calidad, lo cual ofrece mayores posibilidades de que sean adquiridos por este, logrando así el porcentaje de ventas planificado por la organización, lo que repercute directamente en los beneficios de todas las partes implicadas.

### LINK

<https://github.com/IIIAnthonyIII/Documentacion-Proyecto.git>