**Título del Proyecto de Grado**

**N, Email 1**

**Nombre Estudiante 2, Email 2**

**Reporte final de trabajo de grado para optar al título de Ingeniero de Sistemas**

**Asesor: Nombre de Director, Título académico Director (PhD, MSc) en área de desempeño**



**Universidad de San Buenaventura Colombia**

**Facultad de Ingeniería**

**Programa Académico Ingeniería de Sistemas**

**Santiago de Cali, Colombia**

**Mes, Año**

Cómo citar/referencia:

|  |  |
| --- | --- |
| (Apellido1 & Apellido2, Año) | Apellido 1, InicialNombre1, Apellido 2, InicialNombre2, (Año) “Título del proyecto de grado”. Trabajo de grado Ingeniería de Sistemas, Universidad de San Buenaventura Cali, Facultad de Ingeniería, Año |

.

**[](https://co.creativecommons.org/?page_id=13)**

**Bibliotecas Universidad de San Buenaventura**

Biblioteca Digital (Repositorio)

http://bibliotecadigital.usb.edu.co



* Biblioteca Fray Alberto Montealegre OFM - Bogotá.
* Biblioteca Fray Arturo Calle Restrepo OFM **-** Medellín, Bello, Armenia, Ibagué.
* Departamento de Biblioteca - Cali.
* Biblioteca Central Fray Antonio de Marchena – Cartagena.

**Universidad de San Buenaventura Colombia**

Universidad de San Buenaventura Colombia - http://www.usb.edu.co/

Bogotá **-** http://www.usbbog.edu.co

Medellín **-** http://www.usbmed.edu.co

Cali -http://www.usbcali.edu.co

Cartagena - http://www.usbctg.edu.co

Editorial Bonaventuriana - http://www.editorialbonaventuriana.usb.edu.co/

Revistas - http://revistas.usb.edu.co/

**Dedicatoria**

**Agradecimientos**

**Resumen**

Describir brevemente historia del proyecto, inicios, antecedentes y la problemática a abordar

Breve planteamiento de la propuesta de solución

Descripción de los productos del proyecto (Tecnológicos y No Tecnológicos), incluir algunas virtudes de los productos desarrollados y tecnologías, técnicas o herramientas utilizadas.

Breve descripción del proceso de ingeniería realizado (Modelo de Procesos, fases, actividades principales)

**Abstract**

Versión en idioma inglés del resumen

**GLOSARIO**

Recopilación de definiciones y términos utilizados en el proyecto, incluir términos más importantes, así como los poco conocidos o de difícil interpretación o que no sean comúnmente utilizados en el contexto.

Para esta sección se sugiere emplear el estándar ISO/IEC/IEEE 24765, sobre vocabulario en Ingeniería de Sistemas y Software.

* **Termino:** Significado o definición (Referencia, Año).

**TABLA DE CONTENIDO**

[1. INTRODUCCIÓN 10](#_Toc31720776)

[1.1. Contexto 10](#_Toc31720777)

[1.2. Planteamiento del Problema 10](#_Toc31720778)

[1.3. Justificación del Problema 11](#_Toc31720779)

[1.4. Descripción y análisis de la complejidad del problema 11](#_Toc31720780)

[1.5. Objetivo General 11](#_Toc31720781)

[1.5 Objetivos Específicos 11](#_Toc31720782)

[1.6 Descripción de la Solución 11](#_Toc31720783)

[1.6.1 Proceso de Ingeniería 11](#_Toc31720784)

[1.6.2 Productos de Software 11](#_Toc31720785)

[1.6.3 Productos No Software 11](#_Toc31720786)

[2. MARCO TEÓRICO 13](#_Toc31720787)

[2.1 Sección 1 13](#_Toc31720788)

[3. PROCESO DE INGENIERÍA DE SOFTWARE 15](#_Toc31720789)

[3.1 Modelo de proceso de desarrollo de software. 15](#_Toc31720790)

[3.1.1 Descripción de Iteraciones realizadas 15](#_Toc31720791)

[3.2 Educción de Requisitos 16](#_Toc31720792)

[3.2.1 Técnica utilizada: 16](#_Toc31720793)

[3.2.2 Requisitos funcionales documentados 16](#_Toc31720794)

[3.3 Análisis y Diseño 16](#_Toc31720795)

[3.3.1 Atributos y restricciones Arquitecturales. 16](#_Toc31720796)

[3.3.2 Requisitos No Funcionales. 16](#_Toc31720797)

[3.4 Diseño de la Arquitectura. 17](#_Toc31720798)

[3.4.1 Vista de Alto Nivel. 17](#_Toc31720799)

[3.4.2 Vista de Procesos. 18](#_Toc31720800)

[3.4.3 Vista Lógica. 18](#_Toc31720801)

[3.4.4 Vistas de Desarrollo. 18](#_Toc31720802)

[3.5 Patrones de Diseño Utilizados 18](#_Toc31720803)

[3.6 Justificación de Diseño 18](#_Toc31720804)

[3.7 Desarrollo 19](#_Toc31720805)

[3.7.1 Tecnologías utilizadas 19](#_Toc31720806)

[3.7.2 Tecnologías utilizadas y su Relación con los Atributos de Calidad. 19](#_Toc31720807)

[3.8 Prototipos del Software 19](#_Toc31720808)

[3.8.1 Desarrollo de prototipos Versión 1.0 (Iteración No 1) 19](#_Toc31720809)

[3.9 Pruebas y Evaluación de Software 20](#_Toc31720810)

[3.9.1 Casos de Prueba 20](#_Toc31720811)

[3.9.1.1 Caso de Prueba No 1 20](#_Toc31720812)

[3.9.2 Resumen Pruebas Funcionales 20](#_Toc31720813)

[3.10 Acceso a la Aplicación 20](#_Toc31720814)

[3.11 Estándares de ingeniería empleados durante el proyecto 21](#_Toc31720815)

[4. ANÁLISIS DE RESULTADOS 23](#_Toc31720816)

[4.1 Análisis de Impactos 23](#_Toc31720817)

[4.2 Utilización de herramientas 23](#_Toc31720818)

[4.3 Análisis de diseño de componentes 23](#_Toc31720819)

[4.4 Cumplimiento de Objetivos 24](#_Toc31720820)

[4.5 Conclusiones 24](#_Toc31720821)

[4.6 Experiencia de diseño en ingeniería de software 24](#_Toc31720822)

[4.7 Trabajos Futuros 24](#_Toc31720823)

[5. REFERENCIAS 26](#_Toc31720824)

[6. ANEXOS 28](#_Toc31720825)

**LISTA DE TABLAS**

[Tabla 3.1 Especificación y priorización de requisitos funcionales 14](#_Toc466976700)

[Tabla 3.2 Matriz de especificación y clasificación de requisitos no funcionales. 14](#_Toc466976701)

**LISTA DE FIGURAS**

[Figura 1.1 Descripción de la figura. 8](#_Toc466976609)

[Figura 2.1 Descripción de la figura. 11](#_Toc466976610)

[Figura 3.2 Resumen iteraciones, fases y artefactos. 13](#_Toc466976611)

[Figura 3.4 Descripción Diagrama de arquitectura de alto nivel. UML 15](#_Toc466976612)

**1. INTRODUCCIÓN**

* 1. Contexto
  2. Planteamiento del problema
  3. Justificación del problema
  4. Descripción y análisis de la complejidad del problema
  5. Objetivo general
  6. Objetivos específicos
  7. Descripción de la solución

Objetivo General

o

# INTRODUCCIÓN

### Contexto

Esta sección describe cuales son los temas que contextualizan el problema a abordar (sector, población, evidencias, retos y oportunidades).

### Planteamiento del Problema

El planteamiento del problema debe estar relacionado con el objetivo general y debe describir clara y concretamente cuál es el problema, quién o quiénes son los involucrados, cual es el contexto y cómo, cuándo y dónde se hace evidente.

Se pueden relacionar figuras o diagramas que complementen o ilustren el problema abordad.

Las figuras deben citarse en un encabezado donde se explica lo que se está representando

Ejemplo:

A continuación, en la **Figura 1.1** se ilustra…

**FIGURA**

Figura 1.1 Descripción de la figura.

### Justificación del Problema

La justificación del problema debe contestar por qué es pertinente el problema, por qué es un problema de ingeniería.

### Descripción y análisis de la complejidad del problema

Los problemas complejos de ingeniería incluyen una o más de las siguientes características: abordan cuestiones técnicas de gran alcance o contradictorias y que no tienen una solución obvia, abordan problemas o temática que no están incluidos en las normas y códigos actuales, involucran diversos grupos de partes interesadas, incluyen múltiples aspectos o sub-problemas, reúnen múltiples disciplinas, o tienen consecuencias significativas en una serie de contextos diferentes.

### Objetivo General

### Objetivos Específicos

### Descripción de la Solución

### Proceso de Ingeniería

Descripción del modelo de proceso utilizado en el desarrollo de la solución y explicación de las fases del proceso de ingeniería realizado.

### Productos de Software

* Entregrables de proceso de software
* Datos
* Documentación
* Código Fuente

### Productos No Software

* Modelos
* Artículos

**2. MARCO TEÓRICO**

* 1. Sección 1
  2. Sección 2

# 2. MARCO TEÓRICO

### Sección 1

Constituye las bases teóricas donde se sustentará la propuesta de desarrollo del proyecto.

Se pueden relacionar figuras o diagramas que complementen o ilustren cada tema.

Las figuras deben citarse en un encabezado donde se explica lo que se está representando

Ejemplo:

A continuación, en la **Figura 2.1** se ilustra…

**FIGURA**

Figura 2.1 Nombre de la figura.

1. **PROCESO DE INGENIERÍA DE SOFTWARE**
   1. SELECCIÓN DE METODOLOGÍA DE DISEÑO MULTIMEDIA
   2. Modelo de proceso de desarrollo de software
   3. Educción de requisitos
   4. Análisis y diseño
   5. Diseño de la arquitectura.
   6. Patrones de diseño utilizados
   7. Justificación de diseño
   8. Desarrollo
   9. Prototipos del software
   10. Pruebas y evaluación de software
   11. Acceso a la aplicación
   12. Estándares de ingeniería empleados durante el proyecto

# PROCESO DE INGENIERÍA DE SOFTWARE

En este capítulo se presentan los procesos específicos de ingeniería aplicados para el desarrollo de este proyecto. Los procesos son: elección de un modelo de proceso, la educción de requisitos funcionales y no funcionales, definición de artefactos necesarios para el análisis y diseño, la elección de las herramientas, patrones y tecnologías que fueron usadas durante el proceso de desarrollo del software, culminando con las respectivas pruebas que validarán el cumplimiento de los objetivos propuestos.

La presentación de dichos procesos tiene como objetivo evidenciar el desarrollo de competencias en términos de desarrollo de software, por parte de los autores del documento.

### Modelo de proceso de desarrollo de software.

Describir detalladamente por qué se escogió el modelo de proceso utilizado

Describir como se ajusta este modelo a las condiciones del proyecto

### Descripción de Iteraciones realizadas

A continuación, se describe el proceso realizado en las iteraciones definidas:

|  |  |
| --- | --- |
| **Iteración No.1** |  |
| **Artefactos de Entrada:** |  |
| **Descripción:** |  |
| **Artefactos de Salida:** |  |

|  |  |
| --- | --- |
| **Iteración No.2** |  |
| **Artefactos de Entrada:** |  |
| **Descripción:** |  |
| **Artefactos de Salida:** |  |

Ilustrar con una figura las actividades realizadas (Fases, iteraciones, fechas)

Figura 3.2 Resumen iteraciones, fases y artefactos.

### Educción de Requisitos

* + 1. Técnica utilizada:

### Requisitos funcionales documentados

En la Tabla 3.1 se relaciona la respectiva especificación y priorización de requisitos funcionales y no funcionales que fueron educidos utilizando las técnicas anteriormente descritas.

El proyecto exige la creación de un documento de especificación de requisitos (SRS) que debe relacionarse como anexo.

|  |  |  |  |
| --- | --- | --- | --- |
| **Ref.** | **Prioridad** | **Descripción Requisito** | **Fuente de Info.** |
| **RF-1** |  |  |  |

**Tabla 3.1 Especificación y priorización de requisitos funcionales**

### Análisis y Diseño

### Atributos y restricciones Arquitecturales.

En esta sección se describen los requisitos de calidad del software que tienen un impacto significativo en el diseño de la arquitectura.

* **Atributo de calidad:** Explicación y descripción de requisitos No funcionales relacionados.

### Requisitos No Funcionales.

La Tabla 3.2 presenta los requisitos no funcionales documentados.

Ejemplo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ref.** | **Descripción Requisito** | **Interoperabilidad** | **Seguridad** | **Operatividad** | **Usabilidad** |
| **RNF-1** |  |  |  |  |  |

**Tabla 3.2 Matriz de especificación y clasificación de requisitos no funcionales.**

La matriz permite evidenciar que los atributos de calidad se enfocan en ciertos atributos de calidad.

### Diseño de la Arquitectura.

Describir los estilos arquitectónicos utilizados para el diseño del software

### Vista de Alto Nivel.

**Nombre Diagrama:** La siguiente vista, Figura 3.4, representa…

DIAGRAMA

Figura 3.4 Descripción Diagrama de arquitectura de alto nivel. UML

Explicación del diagrama diseñado.

### Vista de Procesos.

### Vista Lógica.

### Vistas de Desarrollo.

### Patrones de Diseño Utilizados

### Justificación de Diseño

Justificación de cómo el diseño propuesto satisface los atributos de calidad identificados

Relacionar los patrones y componentes identificados durante el diseño de la arquitectura y que permiten cumplir con los requisitos de calidad son:

### Desarrollo

### Tecnologías utilizadas

* **Tecnología:** Descripción de la tecnología, incluir referencia (Tecnología, Fecha).

### Tecnologías utilizadas y su Relación con los Atributos de Calidad.

Relacionar los atributos de calidad con las tecnologías implementadas, definiendo cual es el aporte al producto desarrollado.

### Prototipos del Software

Incluir evidencias del software desarrollado en cada una de sus fases, para evidenciar su evolución

**Ejemplo**

### Desarrollo de prototipos Versión 1.0 (Iteración No 1)

|  |  |
| --- | --- |
| **Iteración 1 -** Skecth No.1 - Portal usuarios | |
| **Escena que representa:** | Dashboard o pantalla de inicio del usuario. |
| **Funcionalidades:** | Visualización de contenidos más importantes asignados al rol. |
| **Tipo Usuario:** | Usuario de negocio. |

### Pruebas y Evaluación de Software

Describir el proceso de pruebas realizado, especificando técnicas y herramientas utilizadas.

### Casos de Prueba

Los casos de prueba especificados se describen de las Tablas del 3.5-314.

Ejemplo de plantilla para especificación de casos de prueba

### Caso de Prueba No 1

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **CP-1** | | |
| **Nombre del Caso de Prueba:** |  | | |
| **Proceso o Funcionalidad:** |  | | |
| **Descripción:** |  | | |
| **ID de los Requisitos Relacionados:** |  | **Tipo de Prueba:** |  |
| **Pasos para Ejecución:** |  | | |
| **Variables de Entrada:** |  | | |
| **Resultados Esperados:** |  | | |
| **Observaciones:** |  | | |
| **Estado:** |  | **Tipo de Error:** |  |

### Resumen Pruebas Funcionales

Descripción de resultados del proceso de pruebas, incluir evidencias y gráficos para ilustrar.

### Acceso a la Aplicación

La aplicación fue desplegada en el servidor del grupo Lidis.

Se diseñó un video para mostrar el funcionamiento y uso de la aplicación desarrollada a manera de guía para los dos tipos de usuarios que tiene la aplicación (Administradores y Usuarios de negocio).

**Acceso**

1. **Video:** [**http://lidis.usbcali.edu.co/owncloud/index.php/s/tAN1hStWmstNGzd**](http://lidis.usbcali.edu.co/owncloud/index.php/s/tAN1hStWmstNGzd)
2. **Aplicación (Servidor LIDIS):** [**http://lidis.usbcali.edu.co:8080/octopusb**](http://lidis.usbcali.edu.co:8080/octopusb)

**Credenciales para autenticación:**

**Administrador (BackOffice)**

**Descripción:**

**Login:** admin@usbcali.edu.co

**Password**: 12345678

**Usuario (FrontOffice)**

**Descripción:**

**Login:** user@usbcali.edu.co

**Password**: 12345678

Roles de Pruebas

**Rol de Director Comercial (FrontOffice)**

**Login:** comercial@usbcali.edu.co

**Password**: 12345678

**Rol de Gerente Adminitrativo (FrontOffice)**

**Login:** administrativo@usbcali.edu.co

**Password**: 12345678

### Estándares de ingeniería empleados durante el proyecto

Descripción de los estándares, o buenas prácticas ampliamente difundidas y aceptadas por la comunidad, que hayan sido empleados durante el desarrollo del proyecto o en su documentación.

1. **ANÁLISIS DE RESULTADOS**
   1. Análisis de impactos
   2. Utilización de herramientas
   3. Análisis de diseño de componentes
   4. Cumplimiento de objetivos
   5. Conclusiones
   6. Experiencia de diseño en ingeniería de software
   7. Trabajos futuros

# ANÁLISIS DE RESULTADOS

### Análisis de Impactos

La propuesta de ingeniería discutida en el presente documento tiene diferentes impactos en el escenario en el cual se contextualiza el problema. Estos impactos pueden ser analizados y discutidos desde las siguientes dimensiones o perspectivas: ambiental, social, económica, técnica, e individual [1].

* **Ambiental:** se relaciona con los efectos a largo plazo de las actividades humanas sobre los sistemas naturales. Esta dimensión incluye ecosistemas, recursos naturales, cambio climático, producción de alimentos, agua, contaminación, entre otros.
* **Social:** se relaciona con las comunidades sociales (entendidos como grupos de personas u organizaciones) así como con los factores que atentan contra la confianza de la sociedad. Esta dimensión incluye equidad social, justicia, empleo, y democracia, entre otros.
* **Económico:** se concentra en activos, capital y valor añadido. Esto incluye generación de riqueza, prosperidad, retorno a la inversión, inversión de capital, flujo entrante, y flujo saliente entre otros.
* **Técnica:** se relaciona con la durabilidad de la información, sistemas, infraestructura, así como con su evolución y cambio en relación a las condiciones predominantes en el entorno. Esta perspectiva incluye mantenimiento, innovación, obsolescencia, integridad de los datos, entre otros.
* **Individual:** serelaciona con el bienestar de las personas como individuos. Esto incluye bienestar tanto físico como mental, educación, desarrollo de habilidades, adquisición y aplicación de conocimiento, entre otros.

Adicionalmente, sobre estas perspectivas es posible efectos de primer, segundo y de tercer orden.

Los efectos de primer orden se relacionan con los impactos y con las oportunidades generadas por la inmediata y mera existencia de un sistema de software, incluyendo sus ventajas y desventajas de diseño. Los efectos de según orden son generados por el uso y la aplicación del software. Los efectos de tercer orden se asocian con los impactos de un largo grupo de personas usando la tecnología a largo plazo.

### Utilización de herramientas

Justificación de la selección y el uso de las herramientas tecnológicas durante el proyecto

### Análisis de diseño de componentes

Durante el proceso de ingeniería de software realizado se utilizaron vistas:

Utilizando las técnicas expuestas en:

El análisis y evaluación de los componentes, se realizó utilizando:

### Cumplimiento de Objetivos

Breve explicación y justificación de cómo a lo largo del desarrollo del trabajo se ha dado cumplimiento a los diferentes objetivos.

### Conclusiones

Conciso resumen de los hallazgos o información clave en relación a su proyecto y al producto de software asociado al mismo. Destaque los principales resultados de su proyecto y explique su importancia.

### Experiencia de diseño en ingeniería de software

Lo presentado en las secciones de Análisis de Impactos, Utilización de herramientas y Análisis de diseño de componentes se argumenta como una evidencia de que, a lo largo del proceso de formación de la carrera de ingeniería de sistemas, que culmina con esta experiencia de proyecto de grado, se lograron alcanzar los siguientes resultados de aprendizaje:

* Capacidad para identificar, formular y resolver problemas complejos de ingeniería mediante la aplicación de principios de ingeniería, ciencia y matemáticas.
* Capacidad para aplicar diseño de ingeniería en la producción de soluciones que satisfagan necesidades específicas, considerando tanto la salud pública, la seguridad y el bienestar, como factores de carácter global, cultural, social, ambiental y económico.
* Capacidad para reconocer responsabilidades éticas y profesionales en el ejercicio de la ingeniería y tomar decisiones de manera informada sobre soluciones de ingeniería considerando contextos de carácter global, económico, ambiental y social.
* Capacidad para planear y realizar experimentación adecuada, analizar e interpretar datos, y aplicar juicio de ingeniería en la obtención de conclusiones.

### Trabajos Futuros

Como trabajos futuros se proponen:

Descripción de trabajos futuros y relación con lo realizado en el actual proyecto.

1. **REFERENCIAS**

# REFERENCIAS

Seguir el estilo de Citas IEEE (https://pitt.libguides.com/citationhelp/ieee)

[1] C. Becker et al., "Sustainability Design and Software: The Karlskrona Manifesto," 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Florence, 2015, pp. 467-476.

1. **ANEXOS**

# ANEXOS

SRS (Nombre Anexo, Fecha) Especificación de Requisitos de software.

A2 (Nombre Anexo, Fecha) Titulo o Descripción anexo.