

# **ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**



## **FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

### **INGENIERIA DE SOFTWARE**

### **APLICACIONES INFORMÁTICAS II**

### **PARALELO: OCTAVO 1**

## **Análisis y diseño de sistemas informáticos**

## **Actividad autónoma individual**

### **Estudiante:**

- Kevin Urbano - 6523

### **Docente:**

- Dr. Julio Santillán

---

OCTUBRE 2024 - MARZO 2025

## **TABLA DE CONTENIDOS**

### Índice

1.	Introducción .....	3
2.	Desarrollo .....	3
2.1.	<b>Primeros Pasos: Algoritmos Básicos</b> .....	3
2.2.	<b>Proyectos Intermedios: Aplicaciones con Base de Datos</b> .....	3
2.3.	<b>Proyecto Final: Sistema Informático para una Empresa en Riobamba</b> 4	
3.	Análisis y Conclusiones .....	5

## 1. Introducción

El análisis y diseño de sistemas informáticos es una fase crucial en el desarrollo de software, ya que permite estructurar soluciones tecnológicas que satisfagan las necesidades de los usuarios y las organizaciones. A lo largo de mi carrera como estudiante de Ingeniería de Software, he tenido la oportunidad de aplicar estos conceptos en diversos proyectos, desde algoritmos básicos hasta sistemas completos con bases de datos y metodologías ágiles como Scrum.

En este documento, analizaré mi experiencia en el análisis y diseño de sistemas, destacando ejemplos reales de proyectos académicos y profesionales, así como las implicaciones de este proceso en la calidad del software desarrollado.

## 2. Desarrollo

### 2.1. Primeros Pasos: Algoritmos Básicos

Al inicio de mi formación, el enfoque estaba en la lógica de programación y el desarrollo de algoritmos básicos. Si bien estos ejercicios no involucraban un diseño complejo de sistemas, sentaron las bases para entender estructuras de datos, modularización y optimización de código.

#### Ejemplos:

- Desarrollo de un programa en c++ para resolver operaciones matemáticas básicas, aplicando principios de programación estructurada.
- Desarrollo de un programa en c++ para ordenar un arreglo con los distintos métodos que nos enseñaban.
- Desarrollo de un programa en c++ para una empresa de venta de vehículos implementando programación orientada a objetos.

### 2.2. Proyectos Intermedios: Aplicaciones con Base de Datos

En la mitad de la carrera, los proyectos se volvieron más complejos, integrando bases de datos y una interacción más cercana con clientes (en este caso, profesores o usuarios simulados).

#### Ejemplos:

- Desarrollo de un sistema de gestión de ventas en línea para productos de cómputo:  
**Backend:** Php + MySQL.  
**Frontend:** Bootstrap + javascript + css.

**Metodología:** Primer acercamiento a Scrum (sprints semanales, reuniones de planificación).

Este proyecto me permitió entender la importancia del **diseño de bases de datos** (modelo entidad-relación) y la **arquitectura MVC** (Modelo-Vista-Controlador).

- Desarrollo de una aplicación de escritorio de gestión de inventario:

**Backend:** c# + MySQL + visual studio 2019

**Metodología:** Se utilizo metodología scrum.

Este proyecto me permitió entender la importancia del **diseño de bases de datos** (modelo entidad-relación) y como funcionan las aplicaciones de escritorio.

### 2.3. Proyecto Final: Sistema Informático para una Empresa en Riobamba

El trabajo de titulación fue mi experiencia más completa en análisis y diseño de sistemas. Desarrollé un aplicativo para una empresa real, siguiendo una metodología ágil (Scrum) y aplicando técnicas avanzadas de ingeniería de software.

#### Ejemplo:

- Desarrollo de una aplicación web para la gestión de los servicios en la empresa “hospital del computador” integrando principios ux/ui

#### Tecnologías:

- **Lenguaje:** Typescript
- **Backend:** NestJs
- **Frontend:** Next.js.
- **Base de datos:** PostgreSQL + TypeOrm.

#### Metodología Scrum aplicada:

**Product Backlog:** Lista de funcionalidades priorizadas.

- **Historias de usuario:** Descripción de requerimientos desde la perspectiva del cliente.
- **Sprints:** Iteraciones de 1 semana con entregables funcionales.

Este proyecto reforzó mi capacidad para:

Realizar **levantamiento de requisitos** mediante entrevistas con stakeholders.

Diseñar **diagramas UML** (Casos de uso, Secuencia, Clases).

Implementar **patrones de diseño** como Singleton y Factory.

### 3. Análisis y Conclusiones

Considero que el análisis y diseño de sistemas tiene un impacto directo en:

- **Calidad del Software:** Un buen diseño previene errores y facilita el mantenimiento.
- **Experiencia del Usuario:** Un sistema bien estructurado mejora la usabilidad.
- **Escalabilidad:** Un diseño modular permite futuras actualizaciones.

**Y algunas dificultades que encontré a lo largo de todo este trabajo tras el análisis de este:**

- Al inicio, la falta de experiencia llevaba a diseños poco flexibles.
- La comunicación con clientes reales requirió mejorar habilidades blandas

#### **Conclusiones:**

A lo largo de mi formación académica y profesional, he evolucionado desde el desarrollo de algoritmos simples hasta el diseño de sistemas empresariales completos. La aplicación de metodologías como Scrum y el uso de herramientas de modelado (UML) han sido clave para mejorar mis habilidades en análisis y diseño.

#### **Recomendaciones para futuros proyectos:**

- Realizar un análisis de requisitos exhaustivo antes de diseñar.
- Utilizar patrones de diseño para soluciones más robustas.
- Mantener una retroalimentación constante con los usuarios.
- 

Este proceso no solo ha fortalecido mi capacidad técnica, sino también mi visión para crear software que realmente resuelva problemas reales.