



# Documento de Propuesta de Diseño de Software I, II y III.

---

Componente Gestión de Ova – Fase I

Keimer Enrique Muñoz Mora

Gloria Elena Cordero Almario

Luis Carlos Suárez Bravo

**Tutor:** Alexander Toscano



## Breve reseña

En muchos entornos educativos, los recursos educativos están dispersos en diferentes plataformas y formatos, lo que dificulta su acceso y gestión. El componente propuesto es un sistema integral diseñado para la recopilación, gestión y almacenamiento de Objetos Virtuales de Aprendizaje (OVA), abarcando una amplia variedad de formatos de contenidos. Ofrecerá una interfaz de usuario intuitiva y accesible desde múltiples dispositivos, junto con un sistema de búsqueda avanzada para facilitar la localización de recursos relevantes. Además, permitirá la personalización según las necesidades de los usuarios y organizaciones educativas, integrándose de manera fluida con sistemas de gestión del aprendizaje existentes y priorizando la seguridad de los datos mediante medidas como la encriptación y el control de acceso. El componente propuesto proporcionará una solución integral y escalable para la recopilación y gestión de OVA, ofreciendo una experiencia de aprendizaje enriquecida y adaptada a las necesidades individuales y organizativas.

<b>ETAPA 1 DISEÑO DE LA APLICACIÓN Y ANÁLISIS DE REQUISITOS .....</b>	<b>5</b>
<b>INTRODUCCIÓN .....</b>	<b>5</b>
PROPÓSITO DEL DOCUMENTO .....	5
ALCANCE DEL PROYECTO.....	6
DEFINICIONES Y ACRÓNIMOS .....	6
<b>DESCRIPCIÓN GENERAL.....</b>	<b>6</b>
OBJETIVOS DEL SISTEMA .....	7
FUNCIONALIDAD GENERAL .....	8
USUARIOS DEL SISTEMA .....	8
RESTRICCIONES .....	8
<b>REQUISITOS FUNCIONALES .....</b>	<b>8</b>
CASOS DE USO.....	8
DESCRIPCIÓN DETALLADA DE CADA CASO DE USO.....	8
DIAGRAMAS DE FLUJO DE CASOS DE USO .....	8
PRIORIDAD DE REQUISITOS.....	8
<b>REQUISITOS NO FUNCIONALES .....</b>	<b>8</b>
REQUISITOS DE DESEMPEÑO.....	8
REQUISITOS DE SEGURIDAD.....	9
REQUISITOS DE USABILIDAD .....	9
REQUISITOS DE ESCALABILIDAD .....	9
<b>MODELADO E/R.....</b>	<b>9</b>
DIAGRAMA DE ENTIDAD-RELACIÓN .....	9
DESCRIPCIÓN DE ENTIDADES Y RELACIONES.....	9
REGLAS DE INTEGRIDAD.....	9
<b>ANEXOS (SI ES NECESARIO).....</b>	<b>9</b>
DIAGRAMAS ADICIONALES .....	9
REFERENCIAS .....	9

<b>ETAPA 2: PERSISTENCIA DE DATOS CON BACKEND .....</b>	<b>10</b>
<b>INTRODUCCIÓN .....</b>	<b>10</b>
PROPÓSITO DE LA ETAPA .....	10
ALCANCE DE LA ETAPA .....	10
DEFINICIONES Y ACRÓNIMOS .....	10
<b>DISEÑO DE LA ARQUITECTURA DE BACKEND.....</b>	<b>10</b>
DESCRIPCIÓN DE LA ARQUITECTURA PROPUESTA .....	10
COMPONENTES DEL BACKEND.....	10
DIAGRAMAS DE ARQUITECTURA .....	10
<b>ELECCIÓN DE LA BASE DE DATOS .....</b>	<b>10</b>
EVALUACIÓN DE OPCIONES (SQL o NoSQL).....	10
JUSTIFICACIÓN DE LA ELECCIÓN.....	10
DISEÑO DE ESQUEMA DE BASE DE DATOS .....	11
<b>IMPLEMENTACIÓN DEL BACKEND.....</b>	<b>11</b>
ELECCIÓN DEL LENGUAJE DE PROGRAMACIÓN.....	11
CREACIÓN DE LA LÓGICA DE NEGOCIO .....	11
DESARROLLO DE ENDPOINTS Y APIS.....	11
AUTENTICACIÓN Y AUTORIZACIÓN .....	11
<b>CONEXIÓN A LA BASE DE DATOS .....</b>	<b>11</b>
CONFIGURACIÓN DE LA CONEXIÓN .....	11
DESARROLLO DE OPERACIONES CRUD.....	11
MANEJO DE TRANSACCIONES .....	11
<b>PRUEBAS DEL BACKEND .....</b>	<b>11</b>
DISEÑO DE CASOS DE PRUEBA.....	11
EJECUCIÓN DE PRUEBAS UNITARIAS Y DE INTEGRACIÓN.....	12
MANEJO DE ERRORES Y EXCEPCIONES .....	12

<b>ETAPA 3: CONSUMO DE DATOS Y DESARROLLO FRONTEND .....</b>	<b>13</b>
<b>INTRODUCCIÓN .....</b>	<b>13</b>
PROPÓSITO DE LA ETAPA .....	13
ALCANCE DE LA ETAPA .....	13
DEFINICIONES Y ACRÓNIMOS .....	13
<b>CREACIÓN DE LA INTERFAZ DE USUARIO (UI) .....</b>	<b>13</b>
DISEÑO DE LA INTERFAZ DE USUARIO (UI) CON HTML Y CSS .....	13
CONSIDERACIONES DE USABILIDAD .....	13
MAQUETACIÓN RESPONSIVA .....	13
<b>PROGRAMACIÓN FRONTEND CON JAVASCRIPT (JS).....</b>	<b>13</b>
DESARROLLO DE LA LÓGICA DEL FRONTEND .....	13
MANEJO DE EVENTOS Y COMPORTAMIENTOS DINÁMICOS .....	13
USO DE BIBLIOTECAS Y FRAMEWORKS (SI APLICABLE) .....	14
<b>CONSUMO DE DATOS DESDE EL BACKEND.....</b>	<b>14</b>
CONFIGURACIÓN DE CONEXIONES AL BACKEND .....	14
OBTENCIÓN Y PRESENTACIÓN DE DATOS .....	14
ACTUALIZACIÓN EN TIEMPO REAL (SI APLICABLE) .....	14
<b>INTERACCIÓN USUARIO-INTERFAZ .....</b>	<b>14</b>
MANEJO DE FORMULARIOS Y VALIDACIÓN DE DATOS .....	14
IMPLEMENTACIÓN DE FUNCIONALIDADES INTERACTIVAS .....	14
MEJORAS EN LA EXPERIENCIA DEL USUARIO .....	14
<b>PRUEBAS Y DEPURACIÓN DEL FRONTEND .....</b>	<b>14</b>
DISEÑO DE CASOS DE PRUEBA DE FRONTEND .....	14
PRUEBAS DE USABILIDAD.....	14
DEPURACIÓN DE ERRORES Y OPTIMIZACIÓN DEL CÓDIGO.....	15
<b>IMPLEMENTACIÓN DE LA LÓGICA DE NEGOCIO EN EL FRONTEND .....</b>	<b>15</b>
MIGRACIÓN DE LA LÓGICA DE NEGOCIO DESDE EL BACKEND (SI NECESARIO) .....	15
VALIDACIÓN DE DATOS Y REGLAS DE NEGOCIO EN EL FRONTEND .....	15
<b>INTEGRACIÓN CON EL BACKEND .....</b>	<b>15</b>
VERIFICACIÓN DE LA COMUNICACIÓN EFECTIVA CON EL BACKEND .....	15
PRUEBAS DE INTEGRACIÓN FRONTEND-BACKEND .....	15

# Etapa 1 Diseño de la Aplicación y Análisis de Requisitos

## Introducción

### Propósito del Documento

El presente documento tiene como finalidad documentar el proceso de diseño, análisis e implementación de software de tipo educativo, comercial, OVA, componente o módulo de aplicaciones. Se divide en tres etapas para facilitar el entendimiento y aplicación a gran escala en la asignatura de diseño de software.

### - Etapa 1 Diseño de la Aplicación y Análisis de Requisitos

Esta etapa cumple la tarea de recoger todas las competencias desarrolladas en todas las áreas de formación del currículo de la licenciatura en Informática y Medios Audiovisuales y ponerlas a prueba en el diseño y análisis de un producto educativo que se base en las teorías de aprendizaje estudiadas, articule las estrategias de enseñanza con uso de TIC y genere innovaciones en educación con productos interactivos que revelen una verdadera naturaleza educativa. Estos productos deben aprovechar las fortalezas adquiridas en las áreas de tecnología e informática, técnicas y herramientas, medios audiovisuales y programación y sistemas, para generar productos software interactivos que permitan a los usuarios disfrutar de lo que aprenden, a su propio ritmo. Todo esto en el marco de un proceso metodológico (metodologías de desarrollo de software como MODESEC, SEMLI, etc.) que aproveche lo aprendido en la línea de gestión y lo enriquezca con elementos de la Ingeniería de Software.

### - Etapa 2: Persistencia de Datos con Backend – Servidor

En la etapa 2 se continua con los lineamientos de la etapa 1, para seguir adicionando elementos de diseño e implementación de software, enfocados en el desarrollo de APIs, servidores o microservicios que permitan soportar aplicaciones cliente del software educativo; en este sentido, el curso presenta los conceptos de los sistemas de bases de datos, su diseño lógico, la organización de los sistemas manejadores de bases de datos, los lenguaje de definición de datos y el lenguaje de manipulación de datos SQL y NoSQL; de tal manera que los estudiantes adquieran las competencias para analizar, diseñar y desarrollar aplicaciones para gestionar y almacenar grandes cantidades de datos, mediante el uso de técnicas adecuadas como el diseño y modelo lógico y físico de base datos, manejo de los sistemas de gestión de bases de datos, algebra relacional, dominio del lenguaje SQL como herramienta de consulta, tecnología cliente / servidor; igualmente, se definirán los elementos necesarios para el acceso a dichas bases de datos, como la creación del servidor API, utilizando tecnologías de vanguardia como node.js, express, Nest.js, Spring entre otros; para, finalmente converger en el despliegue de la API utilizando servicios de hospedaje en la nube, preferiblemente gratuitos. También podrá implementar servidores o API's con inteligencia artificial o en su defecto crear una nueva capa que consuma y transforme los datos obtenidos de la IA. El desarrollo del curso se trabajara por proyectos de trabajo colaborativo que serán evaluados de múltiples maneras, teniendo en cuenta más el proceso que el resultado.

### - Etapa 3: Consumo de Datos y Desarrollo Frontend – Cliente

La etapa 3 el estudiante está en capacidad de establecer la mejor elección de herramientas de consumo de datos y técnicas en aras de lograr el mejor producto a nivel de software o hardware acorde a los requerimientos funcionales y no funcionales del problema a solucionar. En este punto el estudiante puede

consumir los datos a través de un cliente que puede ser una aplicación de celular, una aplicación de escritorio, una página web, IoT(internet de las cosas) o incluso, artefactos tecnológicos. El diseño gráfico es de los requisitos esenciales en la capa de presentación, por lo tanto, se requieren los cursos de diseño gráfico vistos previamente. Los elementos anteriores nos permiten elegir el paradigma y tecnología para desarrollar nuestras aplicaciones, teniendo en cuenta que podríamos desarrollar aplicaciones de tipo cliente.

## **Alcance del Proyecto**

En entornos educativos, la dispersión de recursos en diferentes plataformas dificulta su acceso y gestión. El componente propuesto es un sistema integral para la recopilación, gestión y almacenamiento de Objetos Virtuales de Aprendizaje (OVA), con una interfaz intuitiva y búsqueda avanzada, personalizable y seguro, se integra con sistemas de gestión del aprendizaje existentes, a continuación, se describirán una serie de características concebidas para la presente versión y otras que se proponen para versiones futuras.

- Recopilación de OVA
- Gestión de Contenidos
- Adaptabilidad y Personalización
- Integración con Plataformas Educativas
- Seguridad y Privacidad
- Actualización y Mantenimiento

Funcionalidades Futuras:

- Análisis de Datos de Uso
- Generación Automática de Metadatos
- Recomendación Personalizada de Contenidos
- Interoperabilidad con Herramientas de Autoría
- Ampliación de Formatos de Contenido
- Evaluación Integrada

## **Definiciones y Acrónimos**

API: Interfaz de Programación de Aplicaciones (Application Programming Interface).

DBMS: Sistema de Gestión de Bases de Datos (Database Management System).

SQL: Lenguaje de Consulta Estructurada (Structured Query Language).

HTTP: Protocolo de Transferencia de Hipertexto (Hypertext Transfer Protocol).

REST: Transferencia de Estado Representacional (Representational State Transfer).

JSON: Notación de Objetos de JavaScript (JavaScript Object Notation).

JWT: Token de Web JSON (JSON Web Token).

CRUD: Crear, Leer, Actualizar y Borrar (Create, Read, Update, Delete).

ORM: Mapeo Objeto-Relacional (Object-Relational Mapping).

MVC: Modelo-Vista-Controlador (Model-View-Controller).

API RESTful: API que sigue los principios de REST.

CI/CD: Integración Continua / Entrega Continua (Continuous Integration / Continuous Delivery).

SaaS: Software como Servicio (Software as a Service).

SSL/TLS: Capa de sockets seguros/Seguridad de la Capa de Transporte (Secure Sockets Layer/Transport Layer Security).

HTML: Lenguaje de Marcado de Hipertexto (Hypertext Markup Language).

CSS: Hojas de Estilo en Cascada (Cascading Style Sheets).

JS: JavaScript.

DOM: Modelo de Objeto del Documento (Document Object Model).

UI: Interfaz de Usuario (User Interface).

UX: Experiencia del Usuario (User Experience).

SPA: Aplicación de Página Única (Single Page Application).

AJAX: Asíncrono JavaScript y XML (Asynchronous JavaScript and XML).

CMS: Sistema de Gestión de Contenido (Content Management System).

CDN: Red de Distribución de Contenido (Content Delivery Network).

SEO: Optimización de Motores de Búsqueda (Search Engine Optimization).

IDE: Entorno de Desarrollo Integrado (Integrated Development Environment).

CLI: Interfaz de Línea de Comandos (Command Line Interface).

PWA: Aplicación Web Progresiva (Progressive Web App).

OVA: Objetos Virtuales de Aprendizaje (Virtual Learning Objects)

## Descripción General

### Objetivos del Sistema

El sistema tiene como objetivos principales facilitar la recopilación, gestión y almacenamiento de Objetos Virtuales de Aprendizaje (OVA) desde diversas fuentes y formatos, así como proporcionar una interfaz de usuario intuitiva y accesible para una fácil navegación y recuperación de recursos educativos. Además, busca permitir la personalización y adaptabilidad de los OVA según las necesidades específicas de los usuarios y las organizaciones educativas, integrarse fluidamente con sistemas de gestión del aprendizaje existentes y priorizar la seguridad y privacidad de los datos del usuario y los contenidos educativos almacenados en el sistema.

**Conceptos de las entidades**

**Funcionalidad General**

**Usuarios del Sistema**

**Restricciones**

**Requisitos Funcionales**

**Casos de Uso**

**Descripción detallada de cada caso de uso**

**Diagramas de Flujo de Casos de Uso**

**Diagramas de Secuencia**

**Prioridad de Requisitos**

**Requisitos No Funcionales**



**Requisitos de Desempeño**

**Requisitos de Seguridad**

**Requisitos de Usabilidad**

**Requisitos de Escalabilidad**

**Modelado E/R**

**Caracterización de los datos**

**Diagrama de Entidad-Relación**

**Diagrama relacional**

**Descripción de Entidades y Relaciones**

**Reglas de Integridad**

**Anexos (si es necesario)**

**Diagramas Adicionales**

**Referencias**

# **Etapla 2: Persistencia de Datos con Backend**

## **Introducción**

## **Propósito de la Etapa**

## **Alcance de la Etapa**

## **Definiciones y Acrónimos**

## **Diseño de la Arquitectura de Backend**

## **Descripción de la Arquitectura Propuesta**

## **Componentes del Backend**

## **Diagramas de Arquitectura**

## **Elección de la Base de Datos**

## **Evaluación de Opciones (SQL o NoSQL)**

## **Justificación de la Elección**

**Diseño de Esquema de Base de Datos**

**Implementación del Backend**

**Elección del Lenguaje de Programación**

**Creación de la Lógica de Negocio**

**Desarrollo de Endpoints y APIs**

**Autenticación y Autorización**

**Conexión a la Base de Datos**

**Configuración de la Conexión**

**Desarrollo de Operaciones CRUD**

**Manejo de Transacciones**

**Pruebas del Backend**

**Diseño de Casos de Prueba**

## **Ejecución de Pruebas Unitarias y de Integración**

## **Manejo de Errores y Excepciones**

## **Etapas 3: Consumo de Datos y Desarrollo Frontend**

### **Introducción**

#### **Propósito de la Etapa**

#### **Alcance de la Etapa**

#### **Definiciones y Acrónimos**

### **Creación de la Interfaz de Usuario (UI)**

#### **Diseño de la Interfaz de Usuario (UI) con HTML y CSS**

#### **Consideraciones de Usabilidad**

#### **Maquetación Responsiva**

### **Programación Frontend con JavaScript (JS)**

#### **Desarrollo de la Lógica del Frontend**

#### **Manejo de Eventos y Comportamientos Dinámicos**

**Uso de Bibliotecas y Frameworks (si aplicable)**

**Consumo de Datos desde el Backend**

**Configuración de Conexiones al Backend**

**Obtención y Presentación de Datos**

**Actualización en Tiempo Real (si aplicable)**

**Interacción Usuario-Interfaz**

**Manejo de Formularios y Validación de Datos**

**Implementación de Funcionalidades Interactivas**

**Mejoras en la Experiencia del Usuario**

**Pruebas y Depuración del Frontend**

**Diseño de Casos de Prueba de Frontend**

**Pruebas de Usabilidad**

**Depuración de Errores y Optimización del Código**

**Implementación de la Lógica de Negocio en el Frontend**

**Migración de la Lógica de Negocio desde el Backend (si necesario)**

**Validación de Datos y Reglas de Negocio en el Frontend**

**Integración con el Backend**

**Verificación de la Comunicación Efectiva con el Backend**

**Pruebas de Integración Frontend-Backend**