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

**Tutor**: Alexander Enrique Toscano Ricardo

**GitHub**: Mako001

**Estudiantes**:

Marlon Yesid Cobos Villalobos Nair José Madera Tardecilla Felipe Miguel Patrón de la Osa

Cristian Javier Castaño Martínez



Breve reseña

DARC: Descubre, Aprende Regiones de Colombia, es una página web educativa dirigida a estudiantes de primaria (7-12 años) que transforma el aprendizaje sobre las regiones colombianas en una experiencia lúdica e interactiva. A través de juegos didácticos, mapas interactivos y actividades que fomentan el aprendizaje autónomo y la colaboración, DARC busca que los niños aprendan sobre la geografía, cultura, historia y biodiversidad de su país donde podrán conectar imágenes, audios y videos arrastrándolas en diferentes regiones de Colombia, ofreciendo un recursos académico y didáctico para docentes.

[Etapa](#_bookmark0) [1 Diseño de la](#_bookmark0) [Aplicación y](#_bookmark0) [Análisis de](#_bookmark0) [Requisitos](#_bookmark0) [5](#_bookmark0)

[Introducción 5](#_bookmark1)

[Propósito del Documento 5](#_TOC_250004)

[Alcance del Proyecto 5](#_TOC_250003)

[Definiciones y Acrónimos 5](#_TOC_250002)

[Descripción General 5](#_bookmark2)

[Objetivos del Sistema 5](#_TOC_250001)

[Funcionalidad General 5](#_TOC_250000)

Usuarios del Sistema 5

Restricciones 5

[Requisitos Funcionales 5](#_bookmark3)

Mockup de la Interfaz de Usuario (UI) 6

Casos de Uso 6

Descripción detallada de cada caso de uso 6

Diagramas de Flujo de Casos de Uso 6

Prioridad de Requisitos 6

[Requisitos No Funcionales 6](#_bookmark4)

Requisitos de Desempeño 6

Requisitos de Seguridad 6

Requisitos de Usabilidad 6

Requisitos de Escalabilidad 6

[Modelado E/R 6](#_bookmark5)

Diagrama de Entidad-Relación 6

Diagrama relacional 7

Descripción de Entidades y Relaciones 7

Reglas de Integridad 7

[Anexos (si es necesario) 7](#_bookmark6)

Diagramas Adicionales 7

Referencias 7

[Etapa](#_bookmark7) [2: Persistencia de](#_bookmark7) [Datos con](#_bookmark7) [Backend](#_bookmark7) [8](#_bookmark7)

[Introducción](#_bookmark8) [8](#_bookmark8)

Propósito de la Etapa 8

Alcance de la Etapa 8

Definiciones y Acrónimos 8

[Diseño de la Arquitectura de Backend 8](#_bookmark9)

Descripción de la Arquitectura Propuesta 8

Componentes del Backend 8

Diagramas de Arquitectura 8

[Elección de la Base de Datos 8](#_bookmark10)

Evaluación de Opciones (SQL o NoSQL) 8

Justificación de la Elección 8

Diseño de Esquema de Base de Datos 9

[Implementación del Backend 9](#_bookmark11)

Elección del Lenguaje de Programación 9

Creación de la Lógica de Negocio 9

Desarrollo de Endpoints y APIs 9

Autenticación y Autorización 9

[Conexión a la Base de Datos 9](#_bookmark12)

Configuración de la Conexión 9

Desarrollo de Operaciones CRUD 9

Manejo de Transacciones 9

[Pruebas del Backend 9](#_bookmark13)

Diseño de Casos de Prueba 9

Ejecución de Pruebas Unitarias y de Integración 10

Manejo de Errores y Excepciones 10

Etapa 3: Consumo de Datos y Desarrollo Frontend 11

Introducción 11

Propósito de la Etapa 11

Alcance de la Etapa 11

Definiciones y Acrónimos 11

[Creación de la Interfaz de Usuario (UI) 11](#_bookmark14)

Diseño de la Interfaz de Usuario (UI) con HTML y CSS 11

Consideraciones de Usabilidad 11

Maquetación Responsiva 11

[Programación Frontend con JavaScript (JS) 11](#_bookmark15)

Desarrollo de la Lógica del Frontend 11

Manejo de Eventos y Comportamientos Dinámicos 11

Uso de Bibliotecas y Frameworks (si aplicable) 12

[Consumo de Datos desde el Backend 12](#_bookmark16)

Configuración de Conexiones al Backend 12

Obtención y Presentación de Datos 12

Actualización en Tiempo Real (si aplicable) 12

[Interacción Usuario-Interfaz 12](#_bookmark17)

Manejo de Formularios y Validación de Datos 12

Implementación de Funcionalidades Interactivas 12

Mejoras en la Experiencia del Usuario 12

[Pruebas y Depuración del Frontend 12](#_bookmark18)

Diseño de Casos de Prueba de Frontend 12

Pruebas de Usabilidad 12

Depuración de Errores y Optimización del Código 13

[Implementación de la Lógica de Negocio en el Frontend 13](#_bookmark19)

Migración de la Lógica de Negocio desde el Backend (si necesario) 13

Validación de Datos y Reglas de Negocio en el Frontend 13

[Integración con el Backend 13](#_bookmark20)

Verificación de la Comunicación Efectiva con el Backend 13

Pruebas de Integración Frontend-Backend 13

# Etapa 1 Diseño de la Aplicación y Análisis de Requisitos

## Introducción

DARC: Descubre, Aprende Regiones de Colombia, es una aplicación web educativa dirigida a estudiantes de primaria (7-12 años) que transforma el aprendizaje sobre las regiones colombianas en una experiencia lúdica e interactiva. A través de juegos didácticos, mapas interactivos y actividades que fomentan el aprendizaje autónomo y la colaboración, DARC busca que los niños aprendan sobre la geografía, cultura, historia y biodiversidad de su país donde podrán conectar imágenes, audios y videos arrastrándose en diferentes regiones de Colombia, ofreciendo un recursos académico y didáctico para docentes.

## Propósito del Documento

* Definir los requisitos funcionales y no funcionales de DARC.
* Describir la interfaz de usuario y la experiencia del usuario.
* Modelar la estructura de datos de la aplicación.
* Servir como base para la planificación y el desarrollo del software.

## Alcance del Proyecto

DARC es una aplicación web educativa que busca hacer del aprendizaje sobre las regiones de Colombia una experiencia divertida e interactiva para niños de primaria. El proyecto abarca el desarrollo de una plataforma que permita a los usuarios explorar un mapa interactivo, acceder a información detallada sobre cada región, participar en juegos didácticos y realizar actividades interactivas de arrastrar y soltar. DARC se enfocará en la usabilidad y accesibilidad para niños, con el objetivo de complementar los recursos educativos existentes y ofrecer herramientas adicionales para docentes.

Registrarse con información básica. Explorar un mapa interactivo de Colombia. Acercar y alejar el mapa.

Desplazarse por el mapa.

Acceder a información detallada sobre cada región, incluyendo su ubicación geográfica, cultura y tradiciones, historia y biodiversidad.

Visualizar imágenes, reproducir audios y ver videos de cada región. Jugar a juegos didácticos con barras de progreso, sistemas de puntos y recompensas.

Realizar actividades interactivas de arrastrar y soltar.

Obtener validación de respuestas y recibir retroalimentación visual y sonora en las actividades.

Monitorear el progreso y la actividad de los usuarios. Generar reportes y estadísticas sobre el uso de la plataforma.

Definir sistemas de puntos y recompensas (logros) para los juegos.

## Definiciones y Acrónimos

DARC: Descubre, Aprende Regiones de Colombia UI: Interfaz de usuario

UX: Experiencia de usuario

CRUD: Crear, Leer, Actualizar, Eliminar (operaciones de base de datos)

RIC (Regiones, Identidad y Cultura) CRIE (Culturas Regionales Interactivas de Educación)

RECI (Regiones Educativas de Colombia Interactivas) CRIS (Culturas Regionales de Colombia en un Solo clic) RIDE (Regiones, Identidad y Desarrollo Educativo)

CRECE (Culturas Regionales de Colombia para la Educación) REIN (Regiones Educativas de Colombia Interactivas y Nutridas) CIRE (Culturas e Identidad Regional Educativa)

CITREC (Culturas e Identidad Regional con Tecnología Educativa de Colombia)

TICREC (Tecnologías de la Información para la Cultura Regional Educativa de Colombia)

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: Asincrónico 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).

## Descripción General

DARC se desarrollará como una aplicación web accesible desde cualquier dispositivo con conexión a internet. El sitio web estará diseñado con un enfoque en la usabilidad para niños de 7 a 12 años, con el objetivo de ofrecer un recurso académico y didáctico tanto para estudiantes como para docentes.

## Objetivos del Sistema

Educativo: Facilitar el aprendizaje sobre las regiones de Colombia de forma divertida e interactiva.

Motivador: Despertar el interés de los niños por la geografía, cultura e historia de su país.

Accesible: Estar disponible para cualquier estudiante con acceso a internet. Fácil de usar: Contar con una interfaz intuitiva y amigable para niños.

Recurso didáctico: Proporcionar herramientas y materiales de apoyo para docentes.

## Funcionalidad General

Gestión de Perfiles de Usuario:

* Registrar usuario con información básica.
* Almacenar el progreso del usuario.

.

Mapa Interactivo:

* Renderizado un mapa de Colombia.
* Hacer zoom in.
* Hacer zoom out.
* Hacer restablecer zoom.
* Hacer desplazamiento (panning).
* Resaltar visualmente la región seleccionada por el usuario.

Información por Región:

* Mostrar información detallada.
* Mostrar imágenes de cada región.
* Reproducir audios de cada región.
* Reproducir videos de cada región.

Juegos Didácticos:

* Mostrar progreso.
* Mostrar sistema de puntos.
* Obtener recompensa (logros).
* Mostrar recompensas (logros).

Actividades Interactivas (Drag and Drop):

* Arrastrar y soltar elementos en el mapa.
* Validar las respuestas del usuario.
* Mostrar alerta de resultado.
* Reproducir alerta de resultado.

**Usuarios del Sistema**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Funcionalidad | Administradores | Docente Investigador | Docente Invitado | Alumno | Invitado |
| Registrar usuario |  | **X** |  | **X** |  |
| Almacenar el progreso |  |  | **X** | **X** | X |
| Renderizado de mapa |  |  | **X** | **X** | X |
| Hacer Zoom |  |  | **X** | **X** | X |
| Hacer desplazamiento |  |  | **X** | **X** | X |
| Resaltar regiones |  |  | **X** | **X** | X |
| Mostrar información |  |  | **X** | **X** | X |
| Mostrar imágenes | **X** | **X** | **X** | **X** | X |
| Reproducir audios |  |  |  |  |  |
| Reproducir videos | **X** | **X** |  |  |  |
| Mostrar progreso. | **X** | **X** |  |  |  |
| Mostrar sistema de puntos | **X** |  |  |  |  |
| Obtener recompensa | **X** |  |  |  |  |
| Mostrar recompensas |  |  | **x** | **x** | x |
| Arrastrar y soltar elementos |  |  | **x** | **x** | x |
| Validar respuestas | **x** | **x** |  |  |  |
| Mostrar alerta de resultado |  |  | **x** | **x** | x |
| Reproducir alerta de resultado |  |  | x | x | x |

**Restricciones**

## Requisitos Funcionales

**Mockup de la Interfaz de Usuario (UI) Casos de Uso**

**Descripción detallada de cada caso de uso**

**Diagramas de Flujo de Casos de Uso**

**Prioridad de Requisitos**

## Requisitos No Funcionales

**Requisitos de Desempeño**

**Requisitos de Seguridad**

**Requisitos de Usabilidad**

**Requisitos de Escalabilidad**

## Modelado E/R

**Diagrama de Entidad-Relación**

**Diagrama relacional**

**Descripción de Entidades y Relaciones**

**Reglas de Integridad**

## Anexos (si es necesario)

**Diagramas Adicionales**

**Referencias**

# Etapa 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**

## Etapa 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**