



Universidad Carlos III

Arquitectura de Datos

Curso 2024-25

## Práctica 1

Diseño de una Base de Datos no relacional

**Ingeniería Informática, Cuarto curso**

Adrián Fernández Galán (NIA: 100472182, e-mail: 100472182@alumnos.uc3m.es)

César López Mantecón (NIA: 100472092, e-mail: 100472092@alumnos.uc3m.es)

Manuel Gómez-Plana Rodríguez (NIA: 100472092, e-mail: 100472092@alumnos.uc3m.es)

**Prof . Lourdes Moreno López**

**Grupo: 81**

# Índice

1. Introducción	2
2. Diseño conceptual: diagrama de clases UML	2
3. Diseño de agregados	3
4. Validación del esquema	3

## 1. Introducción

## 2. Diseño conceptual: diagrama de clases UML

En esta sección se describe el primer modelo conceptual teniendo en cuenta tanto la semántica extraída de la descripción de la práctica como de los casos de uso.

Podemos extraer la existencia de 8 entidades importantes en nuestro modelo:

1. **Distrito:** Aglomeración de *áreas recreativas*.
2. **Área:** Representación de cada una de las áreas recreativas. Deben contener información sobre el clima, los *juegos* que contiene, su estado y su accesibilidad; entre otros.
3. **Juego:** Representación de cada uno de los instrumentos en un *área recreativa*. Contiene información sobre el modelo, estado, patrones de desgaste, etc.
4. **Clima:** Contiene información meteorológica para una fecha.
5. **Historial de intervenciones:** Mantiene registro de las *intervenciones* realizadas en un juego.
6. **Intervención:** Representa una revisión o intervención sobre un *juego*. Debe contener información sobre la fecha y observaciones realizadas sobre el *juego*.
7. **Usuario:** Contiene información sobre un usuario como la información de contacto.
8. **Incidencia:** Contiene la información relativa a una incidencia sobre un *juego*. Esto es, naturaleza de la incidencia, lista de usuarios que han reportado la incidencia, estado de la incidencia, empresa encargada de solucionarla, fecha de apertura, fecha de cierre y creador.
9. **Encuesta:** Registra información sobre la satisfacción de los *usuarios* para un *juego*.

Con todo lo anterior, hemos realizado el siguiente diseño del sistema.

[Imagen del Modelo sin Agregados]

### 2.1. Caso de uso A

Este caso se centra en proporcionar un listado completo sobre los juegos instalados en diferentes áreas permitiendo una búsqueda por barrio, distrito o área. Por esto se incluye la entidad *distrito* y el campo *barrio* en *área*.

- La base de datos mantendrá registros sobre todas las incidencias. No obstante la generación de informes es algo que queda fuera del diseño y se contempla como funcionalidad de la aplicación..
- 
- Explicación del modelo básico sin los casos de uso y sin destacar las relaciones entre clases
  - Área (Centrándose en Tipo, lista de juegos, estado, N-juegos, clima)
  - Juego (Centrándose en Estado, Modelo, Historial de intervenciones, lista de incidencias)
- Análisis de los casos de uso y destacar decisiones en el diseño según estos casos
  - Caso de uso A
    - Este caso de uso está centrado en proporcionar un listado completo de los juegos instalados en las diferentes áreas
    - Para poder listar las áreas dado un distrito según el Req 1 se ha creado una entidad Distrito que facilite la búsqueda
    - Para satisfacer el Req 4 se ha añadido a la entidad Juego el atributo "patrón de desgaste"
  - Caso de uso B
    - Este Caso de uso está centrado en la capacidad del sistema de reportar incidencias

- Se ha incorporado una entidad Incidencia con las características asociadas a los requisitos y otra entidad Usuarios
- Para Req 2 se ha integrado en la entidad Incidencia el atributo lista de destinatarios, para conocer los usuarios que han realizado los reportes
- Habría que explicar el por qué de todos los atributos de la entidad Incidencia, ahora mismo no caigo
- Caso de Uso C
  - Este caso de uso está centrado en analizar las condiciones meteorológicas para realizar una planificación eficiente del mantenimiento
  - Para poder cubrir Req 1 se ha creado una entidad clima con los atributos "Temperatura", "Precipitación", "Fecha"
  - Para Req 2 y Req 3 no es necesario realizar ningún cambio dado que se conocen los aspectos meteorológicos de las áreas y por lo tanto de sus juegos
  - Req 4 queda fuera de nuestro alcance como diseñadores de bases de datos
- Caso de Uso D
  - Este caso de uso busca mejorar la toma de decisiones sobre las áreas recreativas a través de la capacidad de generar informes y realizar encuestas de satisfacción por parte de los usuarios
  - Para cubrir Req 1 no es necesario introducir nuevas características ya que se contempló con anterioridad el uso incidencias
  - Se ha incorporado una entidad Encuesta para satisfacer Req 2, esta entidad tiene el usuario que lo ha realizado, la satisfacción que tiene con el juego y un comentario.
  - Para Req 3 proporcionamos la información de las áreas junto a los diferentes reportes
- Caso de Uso E
  - Para Req 1 no es necesario incluir ninguna característica, ya que los cambios realizados para el caso de uso A ya contemplan este requisito
  - Para Req 2 se ha añadido el atributo "N-juegos" (esto no se podría calcular sin necesidad de tenerlo estático??)
  - Req 3 es viable hacerlo??
  - Req 4 queda fuera del nuestro alcance

### 3. Diseño de agregados

Con el objetivo de poder analizar las capacidades que tendrán los agregados es necesario conocer las acciones que conllevan los distintos casos de uso.

- **Lectura:** Casos de Uso A y E
- **Escritura:** Casos de Uso B, C y D

En el diseño de agregados para el sistema de áreas y sus juegos se han creado 2 agregados, cada uno optimizado para casos de uso específicos. En estos agregados se ha buscado un equilibrio entre lecturas y escrituras para aquellos agregados enfocados a añadir elementos.

Además hemos determinado si las relaciones entre entidades deben de ser embebidas, referencias o tablas resumen. Estas decisiones se fundamentan en la frecuencia de lectura, modificación y crecimiento de los datos.

A continuación se describirán los agregados y las entidades que lo conforman, además de abordar las características de los agregados, como son la raíz y el perímetro.

#### 3.1. Agregado sobre Áreas y Juegos

- **Entidades:** Distrito, Área, Clima y Juego
- **Raíz:** Área
- **Perímetro:** Este agregado está enfocado en poder consultar la información relevante sobre las áreas y los juegos que lo comprenden. Cada juego está asociado al área al que pertenece.

- **Casos de Uso optimizados:**

- CU\_A : (Listado detallado de juegos y su estado): Este agregado proporciona consultas rápidas para obtener información sobre los juegos y su estado actual.
- CU\_E : (Informe agrupado por distritos): También es capaz de generar un informe agrupado por distritos que muestre información sobre sus juegos.

[Imagen con agregados]

## 4. Validación del esquema

Para mantener la integridad de los datos, MongoDB ofrece el uso de estrategias como el uso de referencias en vez de embeber los datos. Sin embargo, para aquellos datos que hemos decidido embeber, debemos redactar una serie de reglas para mantener la consistencia de nuestro esquema. Estas reglas son:

- Nombres de *Distrito*: Cada distrito debe tener un nombre único para así evitar duplicados.
- Coordenadas de *Área*: Las coordenadas de las áreas serán una tupla que contenga en grados, minutos y segundos la latitud y longitud del área.
- Estado Operativo *Área*: El estado operativo de las áreas sólo podrán guardar los valores ".operativas", ".En Mantenimiento" o "Fuera de servicio".
- Número de Juegos *Área*: El número de juegos instalados en un área debe ser un entero no negativo y siempre menor o igual al número máximo de juegos de un área.
- Número Máximo de Juegos *Área*: El número máximo de juegos de un área debe ser un entero no negativo.
- Temperatura *Clima*: La temperatura de *Clima* es un float y se medirá en grados Celsius.
- Precipitación *Clima*: La precipitación de *Clima* es un float y se medirá en litros por metro cuadrado.
- Estado Operativo *Juego*: El estado operativo de los juegos sólo podrá guardar dos valores "Disponible." o "No Disponible".
- QR *Juego*: El QR de los juegos debe de ser una imagen.
- Accesibilidad *Juego*: La accesibilidad de un juego debe ser un texto que indique las personas con alguna deficiencia física, mental o sensorial que puedan acceder.
- Fechas *Incidencia*: La fecha de apertura debe ser anterior a la fecha de cierre.
- Fechas: Las fechas de todas las entidades deben seguir el formato: yyyy/mm/dd hh:MM:ss".