

# **Diseño Arquitectónico de una Aplicación**

**PAMN - Programación de Aplicaciones Móviles Nativas**

Chamil José Cruz Razeq

15 de octubre de 2023

## Índice

<b>1</b>	<b>Introducción</b>	<b>3</b>
<b>2</b>	<b>Dominio</b>	<b>3</b>
<b>3</b>	<b>Arquitectura</b>	<b>3</b>
<b>4</b>	<b>Casos de Uso</b>	<b>5</b>
<b>5</b>	<b>Conclusion</b>	<b>5</b>

## 1. Introducción

Este documento recoge los resultados de la tarea propuesta para la confección de un diseño arquitectónico de una aplicación. Para ello se establecerá un dominio y arquitectura, presentando sus características principales acompañadas de un diagrama.

Todas las tareas propuestas se encuentran disponibles en el siguiente repositorio de GitHub.

## 2. Dominio

Aplicación alternativa al servicio web del cabildo para la solicitud de permisos e información sobre las áreas recreativas.

Como funcionalidades comunes con el servicio web encontramos:

- Listado de áreas disponibles.
- Información sobre las instalaciones y estado de cada área.
- Solicitud de permisos para su uso (hornos de piedra, acampadas, caravanas...).

Como funcionalidades adicionales ofrecidas por la aplicación:

- Búsqueda de áreas facilitada a través de filtros.
- Historial de solicitudes.
- Información en tiempo real del estado de las áreas y sus instalaciones.
- Advertencias en tiempo real en situaciones de emergencia.

## 3. Arquitectura

Para el desarrollo del dominio anterior se ha seleccionado la arquitectura MVVM <sup>1</sup> por su afinidad con el desarrollo para aplicaciones móviles y atractivas características

---

<sup>1</sup>El patrón MVVM ayuda a separar limpiamente la lógica de presentación y negocios de una aplicación de su interfaz de usuario [1, MVVM].

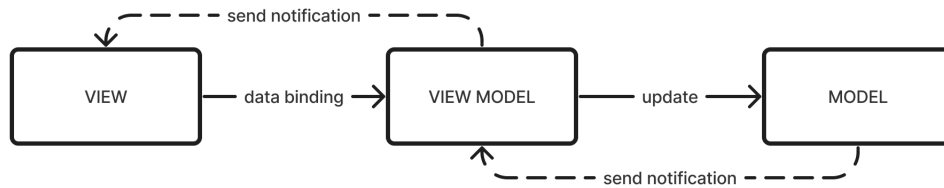


Figura 1: Diagrama de MVVM

que facilitan el desarrollo y la realización de pruebas [Figura 1].

Para el dominio planteado, podemos identificar:

- VistaAreas, ModeloVistaAreas y ModeloAreas: con información reducida de cada area [Figura 2].
- VistaAreaEspecifica, ModeloVistaAreaExpecifica y ModeloAreaEspecifica: con información detallada de un área especificada.
- VistaSOS, ModeloVistaSOS y ModeloSOS: con información de los medios de seguridad y emergencias.
- VistaSolicitud, ModeloVistaSolicitud y ModeloSolicitud: gestionando el formulario de solicitudes de uso de áreas recreativas.
- VistaSolicitudes, MModeloVistaSolicitudes y ModeloSolicitudes: gestionando las solicitudes de uso de áreas recreativas.

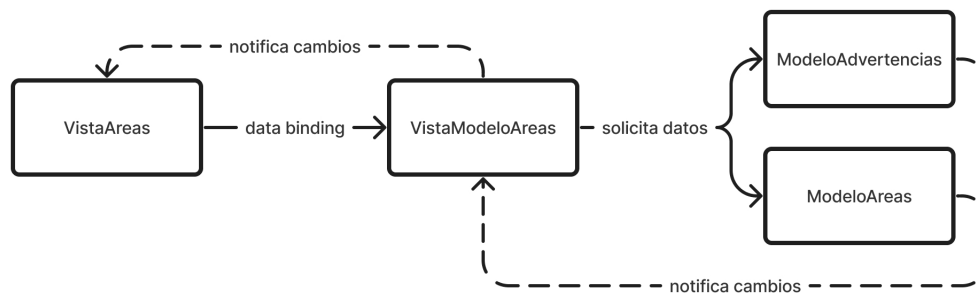


Figura 2: Diagrama de funcionalidad de áreas

## 4. Casos de Uso

Tomando como referencia la [Figura 2] podemos identificar dos casos de uso fundamentales para la aplicación propuesta.

- El usuario puede ver las áreas recreativas disponibles: en este caso siguiendo lo observado en [Figura 2], el usuario (usando o no filtros), tendrá a su disposición una lista de áreas con información esencial. Siguiendo la arquitectura MVVM, el "Modelo" proveera con la información actualizada al "ModeloVista" que notificara a la "Vista". En caso de que el usuario interactue con el sistema, se procederá a solicitar la información actualizada al "Modelo".
- El usuario puede seleccionar y ver información adicional de un área: en este caso la "Vista", el "ModeloVista" y el "Modelo" cambian al caso de "ÁreaEspecífica". En este contexto solo se hará uso de la información provista por el "Modelo", los cambios que se hayan podido producir sin interacción del usuario.

## 5. Conclusion

El modelo MVVM demuestra ser una arquitectura versátil, idónea para cualquier proyecto de desarrollo móvil. En nuestro caso facilitaría no solo la creación de los diferentes

componentes del software, sino que tambien facilitaría la confección de pruebas (a través del "ModeloVista"). Las principales ventajas de MVVM frente a otros modelos son la posibilidad de realización de pruebas facilmente (frente a MVC) y su sencillez, dado el dominio planteado, ya que otros modelos plantean un manejo mas estricto de la información (frente a MVI).

## Referencias

- [1] Microsoft Learn - <https://learn.microsoft.com/es-es/dotnet/architecture/maui/mvvm>