

# Programa de Recuperación de PMDM 2025

Descargar estos apuntes [pdf](#) o [html](#)

## Índice

- ▼ Programa de Recuperación de PMDM 2025
  - ▼ Normativa
    - Acceso a la segunda convocatoria (extraordinaria)
  - ▼ Trabajo a realizar
    - Orientaciones generales para el examen
  - Tutorías
  - ▼ Seguimiento y calendario
    - Marzo 2025
    - Abril 2025
    - Mayo 2025
  - Superación del módulo en convocatoria extraordinaria

# Normativa

La normativa vigente que regula el acceso a la convocatoria extraordinaria de este curso escolar dice:

**RESOLUCIÓN de 6 de julio de 2023**, de la Secretaría Autonómica de Educación y Formación Profesional, por la que se dictan instrucciones sobre la ordenación académica y de organización de la actividad docente de los centros de la Comunitat Valenciana que durante el curso 2023-2024 impartan Formación Profesional de grado C, D y E. [2023/7821]

## Acceso a la segunda convocatoria (extraordinaria)

1. Quien no supere algún módulo profesional de los realizados en el centro educativo en la primera convocatoria dispondrá en el mismo curso escolar de una segunda convocatoria.
2. Con el fin de facilitar al alumnado la superación de los resultados de aprendizaje no adquiridos en los módulos profesionales, el profesorado organizará un programa formativo específico.
3. El programa formativo específico de los módulos no superados se diseñará de forma diferenciada según los periodos o momentos de aplicación, que podrán ser los siguientes:
  - i. Programa formativo específico posterior a la evaluación de la **primera convocatoria** del **primer curso**. Se diseñará para que el alumno o la alumna lo realice entre la primera y la segunda convocatoria, sin asistir a clases ni a tutorías, pero contando con la orientación previa del profesorado.
  - ii. Programa formativo específico tras la evaluación de la **segunda convocatoria** del **primer curso**. Se diseñará para que el alumno o la alumna lo pueda realizar simultáneamente a los módulos de segundo curso, teniendo en cuenta que no se garantizará su asistencia a las clases del módulo o los módulos pendientes.
  - iii. Programa formativo específico de **segundo curso** tras la evaluación de la **primera convocatoria** realizada antes del inicio del primer periodo de realización del módulo profesional de Formación en Centros de Trabajo. **Este programa incorporará las actividades que el alumno o la alumna realizará durante el tercer trimestre del año académico, con docencia directa por parte de profesor o profesora responsable de cada módulo profesional.**
  - iv. Programa formativo específico de **segundo curso** tras la evaluación de la **segunda convocatoria** haciéndolo compatible con la realización del módulo de Formación en Centros de Trabajo o de Proyecto. Se diseñará para que el alumno o la alumna lo realice simultáneamente, compatibilizando **la asistencia a clases o contando con tutorías** y orientación del profesorado correspondiente.

#### 4. Tras la evaluación de la primera convocatoria, el equipo educativo podrá autorizar la renuncia extraordinaria a la segunda convocatoria.

Las razones que se aleguen para la renuncia se deben justificar siempre documentalmente.

## Trabajo a realizar

Además de poder repasar los ejercicios y prácticas realizados durante el curso, el alumno dispondrá de una serie de ejercicios similares a los realizados en exámenes del módulo profesional que le permitirán comprobar su nivel y destreza para superarlo.

Todos los ejercicios estarán disponibles en AULES.

## Orientaciones generales para el examen

Al tratarse de un examen final del módulo, la prueba será similar a la del último trimestre donde se intentan poner en práctica todo lo tratado a lo largo del curso y mínimo necesario para realizar una aplicación móvil operativa. Aspectos generales que que habría que trabajar:

1. Conocer la **arquitectura de la aplicación** descrita en el tema 2, con sus partes y los paquetes propuestos en ella.
2. Saber definir y usar **Hilt** (con Ksp) como librería para la inyección de dependencias.
3. Saber gestionar en la arquitectura una **capa de datos** con una persistencia a través de la librería Room.
  - Definir entidades a partir de un modelo dado. Nombres y tipos de campo adecuados.
  - Definir la BD y los DAO a usar. Eliminando elementos innecesarios como Converters si no se usan.
  - Definir el repositorio y el mapeo entre entidades y modelos y viceversa.
  - Realizar una primera carga de datos en la BD con una fuente de datos de prueba.
  - Realizar todos estos procesos de forma **asíncrona**.
4. Saber gestionar en la arquitectura la **capa de UI** con:
  - Definición **'features'** a modo de **pantalla** (Screen) para ello:
    - Deberemos saber definir y saber interpretar componentes básicos de la interfaz de usuario: **Layouts** , **LazyLayouts** , **Cards** , **TextFields** , **Modificadores** , etc. dentro de los apuntes si fuera necesario.
    - Saber definir una entrada de datos básica, a través de **TextFields** y adicionalmente hacer una validación de los datos introducidos.
  - Deberemos saber definir un **Scaffold** con los elementos vistos en clase, como mínimo:
    - **AppBar** con sus elementos como navegación, título y acciones, etc. Así como menús de desbordamiento (overflowmenu).
    - **FloatingActionButton** .

- `NavigationDrawer` y como gestionarla.
- `BottomAppBar` y como gestionarla.
- `ModalBottomSheet` y como gestionarla.
- Saber definir un `ViewModel` asociado la UI o pantalla como **gestor del estado** y encargado de realizar todo la lógica y comunicación con la capa de datos. En este sentido deberemos saber:
  - Saber definir un `ViewModel` con Hilt e inyectarle clases *colaboradoras* en su constructor.
  - Saber crear una instancia de `ViewModel` con el `ViewModelOwner` adecuado, usando las funciones que proporciona Hilt para ello.
  - Realizar correctamente la gestión del estado de los datos que se necesiten en la UI.
  - Saber lanzar un proceso asíncrono o no bloqueante desde el `ViewModel` y gestionar su resultado.
  - Implementar '*State Hoisting*' de los datos necesarios para la UI hasta el `ViewModel` . Intentando que la UI sea completamente '**stateless**'. Saber definir un componente '**stateful**' si su estado no fuera necesario de manejar en el VM.
  - Saber definir un **patrón 'Sum-Type'** para agrupar callbacks de eventos de la UI si fuera necesario.
- Definir la navegación entre pantallas, ya sea dentro de un mismo Scaffold o entre pantallas con su propio Scaffold.
  - Pasar datos entre pantallas de forma segura y sin acoplamiento usando parámetros en la ruta de navegación.
  - Saber hacer llegar estos datos al ViewModel de destino.

# Tutorías

En caso necesario, el alumno podrá **solicitar** al profesor una tutoría para resolver dudas de forma presencial en el instituto en el horario que se haya preestablecido en el calendario de seguimiento.

## Seguimiento y calendario

El profesor establece un calendario con al menos los siguientes datos:

- Horario y días de tutorías.
- Temas principales a tratar en cada semana.

El profesor calificará y orientará a los alumnos en cada entrega de ejercicios que realicen.

Los domingos se activarán enlaces a las actividades propuestas de reparo y examen de ejemplo **en aules**.

- **10 de marzo de 2025:** Disponibilidad de actividades de marzo.
- **31 de marzo de 2025:** Disponibilidad de actividades de abril.

El horario para tutorías presenciales serán:

- **Pepe:** lunes de 12:00 a 12:55.
- **Xusa:** lunes de 12:00 a 12:55.
- **Juanjo:** lunes de 11:05 a 12:00.



### Nota

En ellas solo se resolverán dudas previa solicitud del alumno del tema propuesto a trabajar o leer para esa semana. Junto con las actividades que se irán proponiendo.

## Marzo 2025

| Lunes | Martes | Miércoles | Jueves | Viernes |  |
|-------|--------|-----------|--------|---------|--|
| 26    | 27     | 28        | 29     | 30      |  |
| 3     | 4      | 5         | 6      | 7       |  |

| Lunes     | Martes | Miércoles | Jueves | Viernes |  |
|-----------|--------|-----------|--------|---------|--|
| <b>10</b> | 11     | 12        | 13     | 14      | Repasar lenguaje Kotlin Tema 1.1 y Tema 1.2. |
| <b>17</b> | 18     | 19        | 20     | 21      | Leer Tema 2.3 y Temas del 3.1 al 3.3         |
| <b>24</b> | 25     | 26        | 27     | 28      | Leer Tema 3.4 ViewModel y Tema 3.5 Hilt.     |

## Abril 2025

| Lunes     | Martes | Miércoles | Jueves | Viernes |   |
|-----------|--------|-----------|--------|---------|---|
| <b>31</b> | 1      | 2         | 3      | 4       | Leer Tema 3.6 Crear una lista con ViewModel + Hilt.           |
| <b>7</b>  | 8      | 9         | 10     | 11      | Leer Tema 4.1 Añadir Scaffold, ModalBottomSheet y ActionBars. |
| <b>14</b> | 15     | 16        | 17     | 18      | Leer Tema 4.2 Navegación con tipos seguros y NavigationBar.   |
| 21        | 22     | 23        | 24     | 25      | Leer Tema 5.1 Cargar datos de Mock a Room.                    |

## Mayo 2025

| Lunes     | Martes | Miércoles | Jueves | Viernes |                                |
|-----------|--------|-----------|--------|---------|--------------------------------|
| <b>28</b> | 29     | 30        | 1      | 2       |                                |
| <b>5</b>  | 6      | 7         | 8      | 9       | Hacer ejemplo de examen        |
| <b>12</b> | 13     | 14        | 15     | 16      | Hacer ejemplo de examen        |
| <b>19</b> | 20     | 21        | 22     | 23      | Dudas generales ejemplo examen |
| <b>26</b> | 27     | 28        | 29     | 30      | Dudas generales examen         |

# Superación del módulo en convocatoria extraordinaria

Los ejercicios realizados y calificados durante este periodo **no tendrán reflejo en la calificación de la convocatoria extraordinaria**. Para superar el módulo en convocatoria extraordinaria, el alumno deberá obtener una calificación igual o superior a 5 en el examen que se realizará de forma presencial en el instituto en el mes de junio, según el calendario que establecerá el equipo directivo y del que se informará al alumnado a través de AULES.

El examen de recuperación extraordinario se publicará en Aules próximamente.