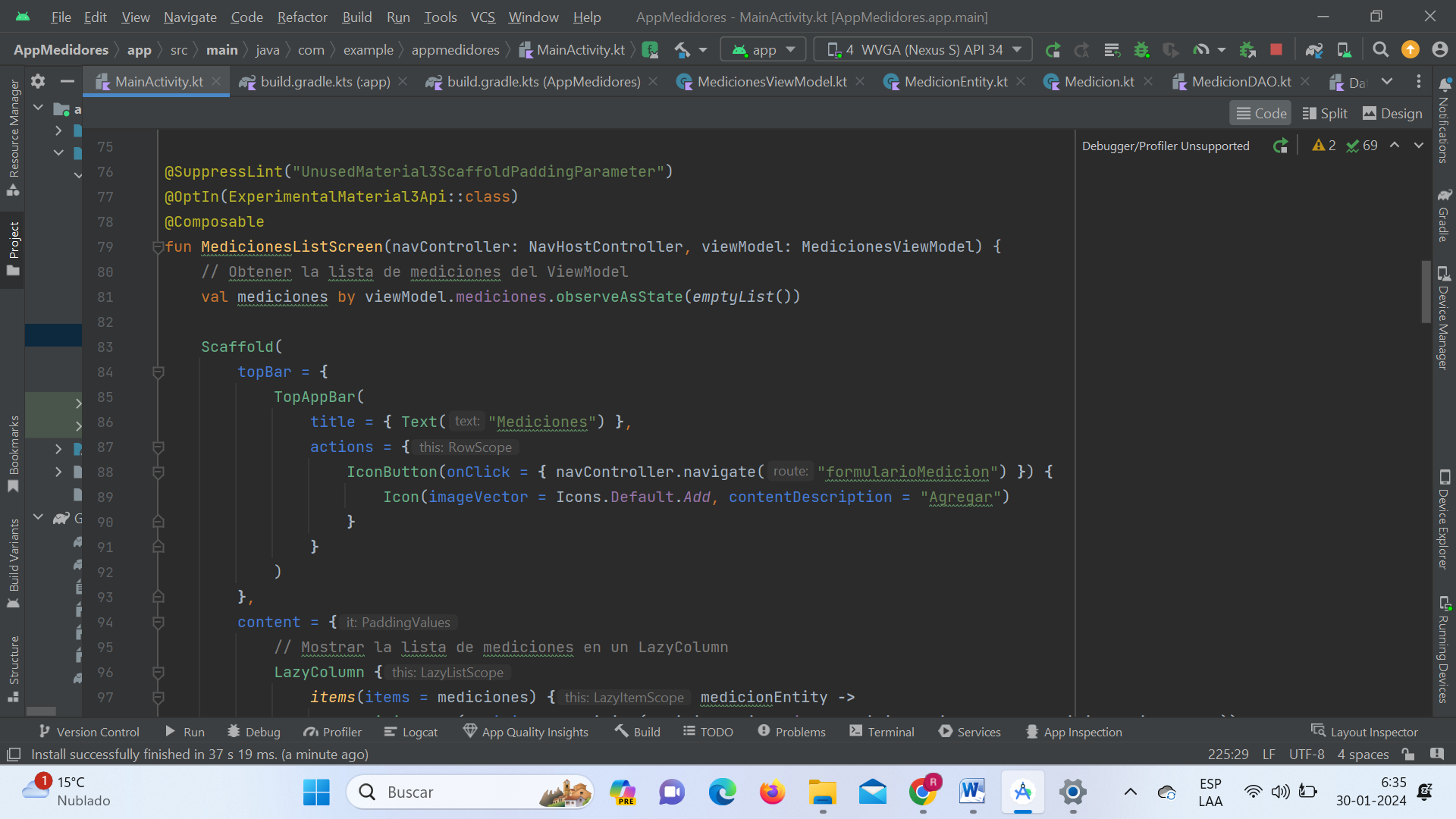
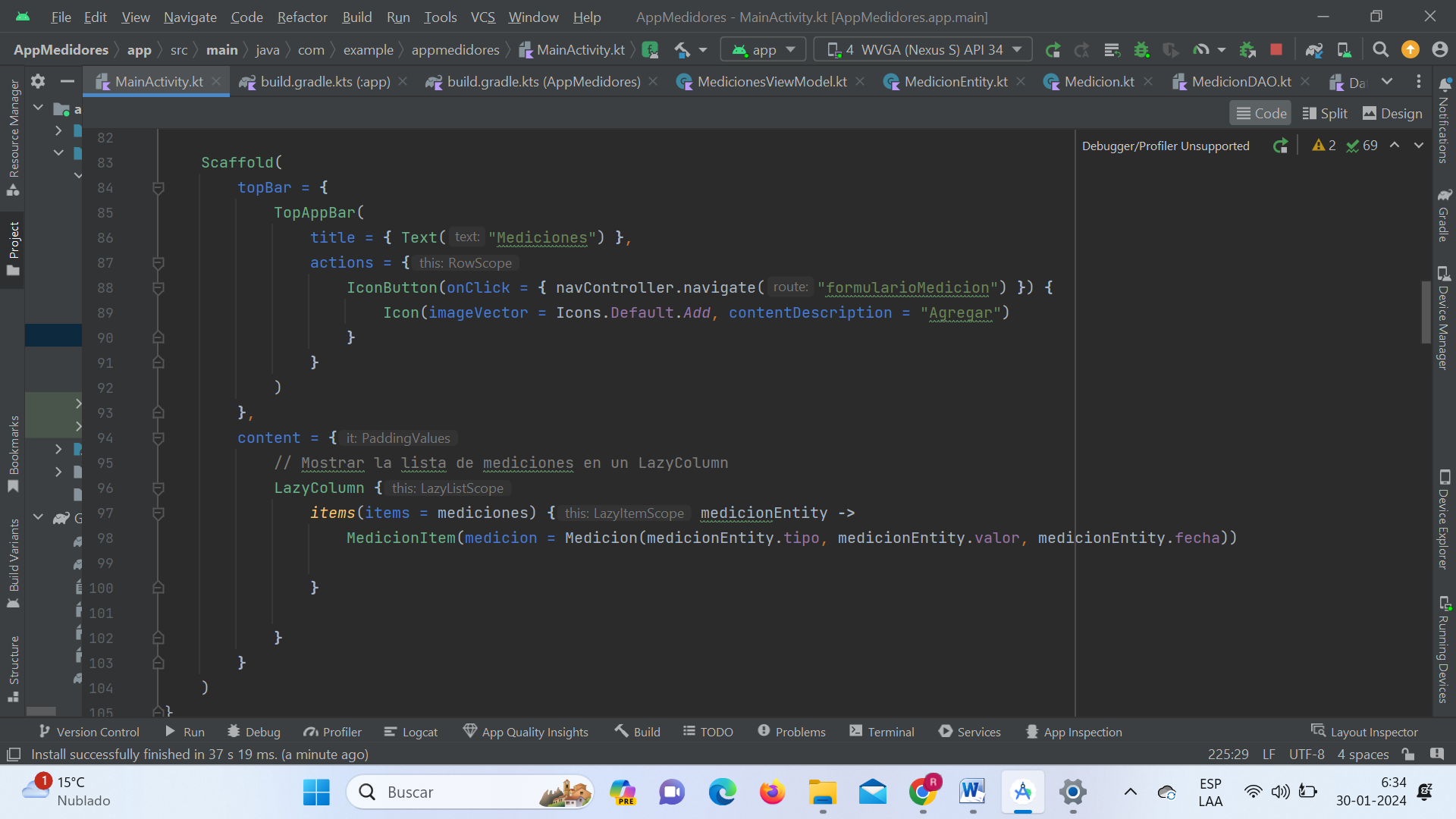
Informe programación 2

1. Diseñar y programar la solución usando Programación Orientada a Objetos en Kotlin. La aplicación debe registrar al menos el tipo de gasto, el valor indicado por el medidor respectivo y la fecha.

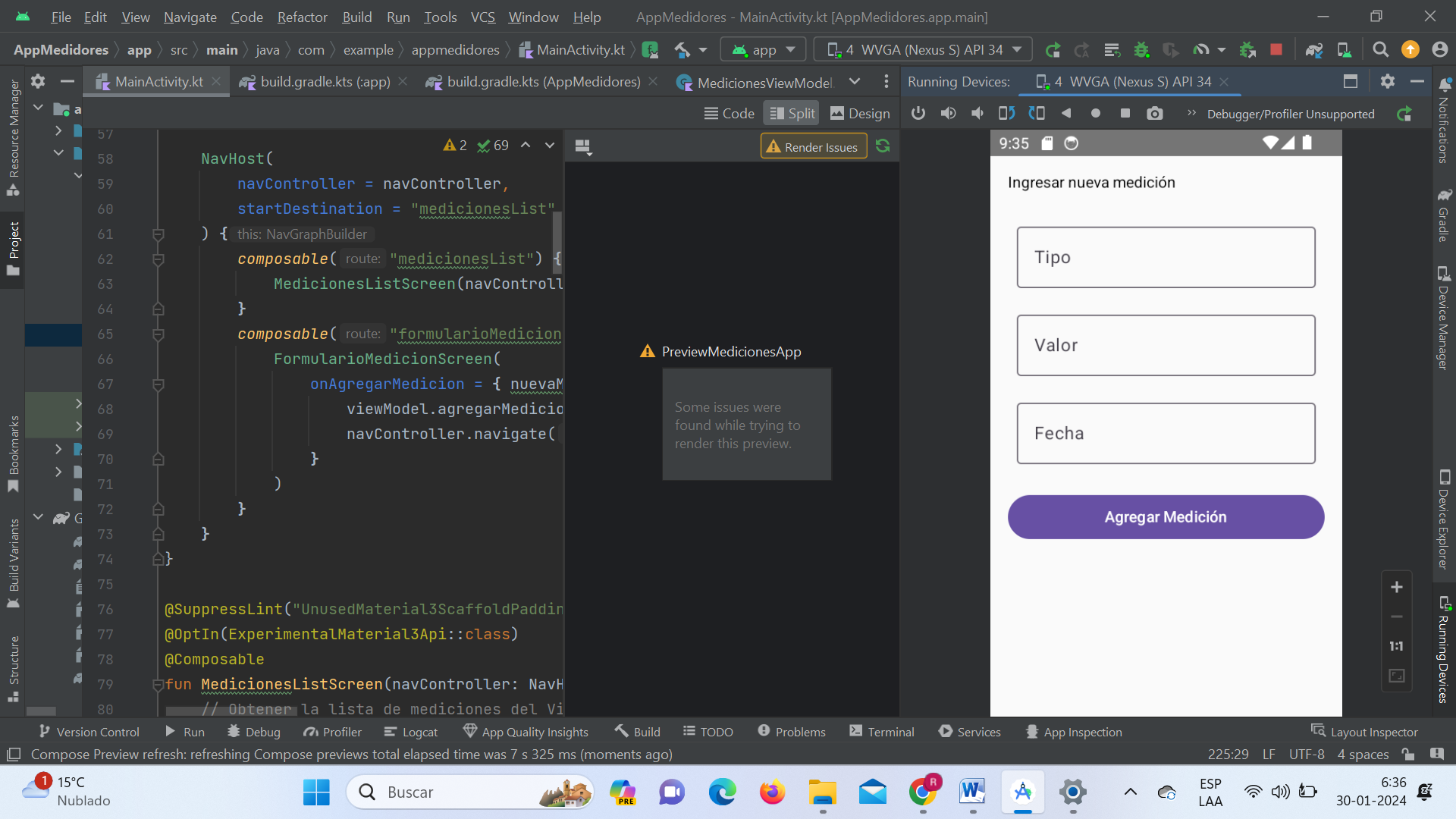
2. Diseñar y codificar 2 pantallas en la aplicación, puede utilizar Compose o Vistas XML:

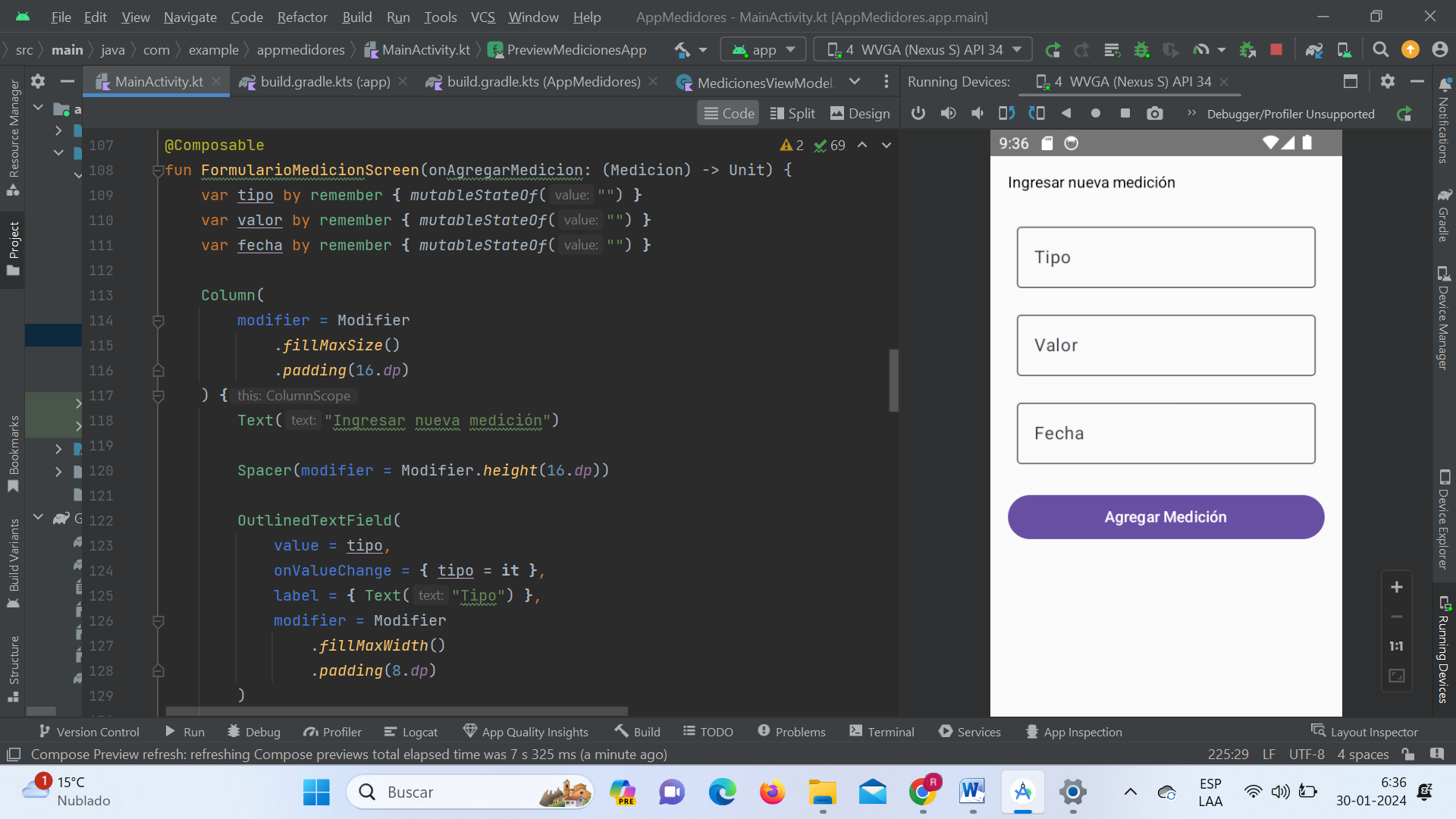
a. Una que muestre el listado de mediciones registradas por los medidores

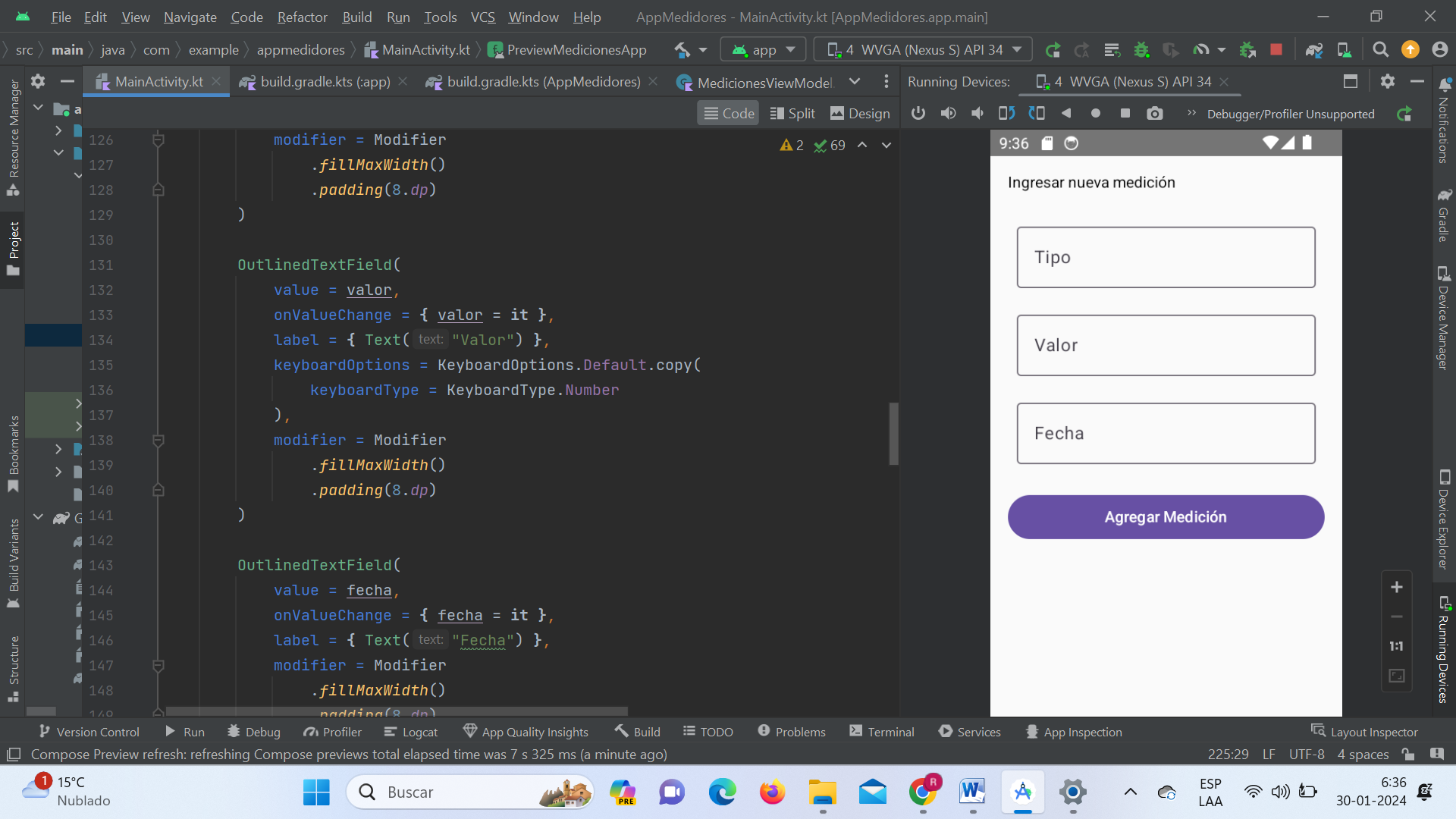




b. Una que muestre un formulario para ingresar el registro



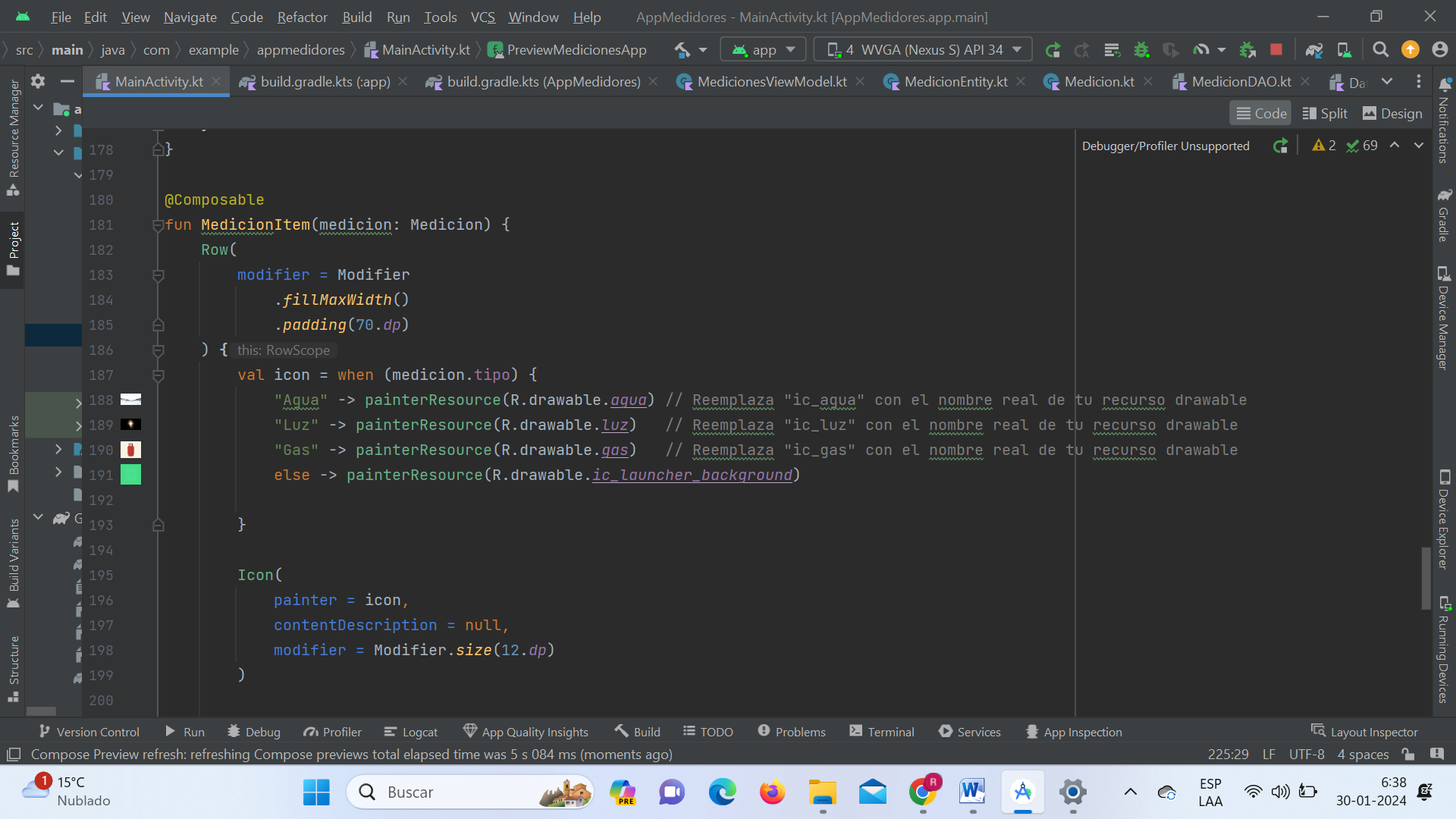


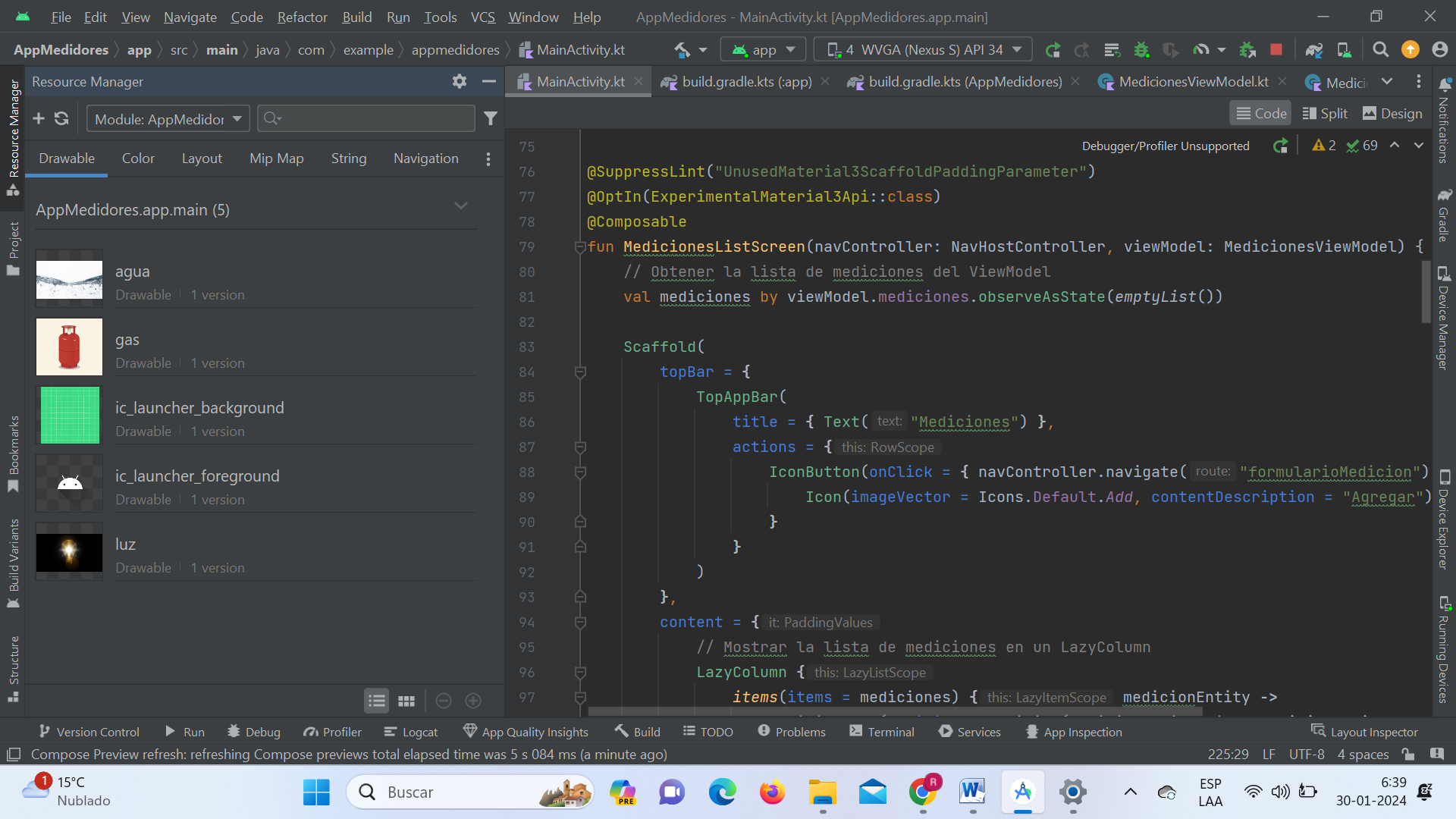




3. Implementar la gestión de eventos para el botón que permite navegar hacia el formulario y el botón que guarda un nuevo registro.

4. Utilizar recursos drawables o íconos para representar los diferentes tipos de registro (agua, luz, gas).

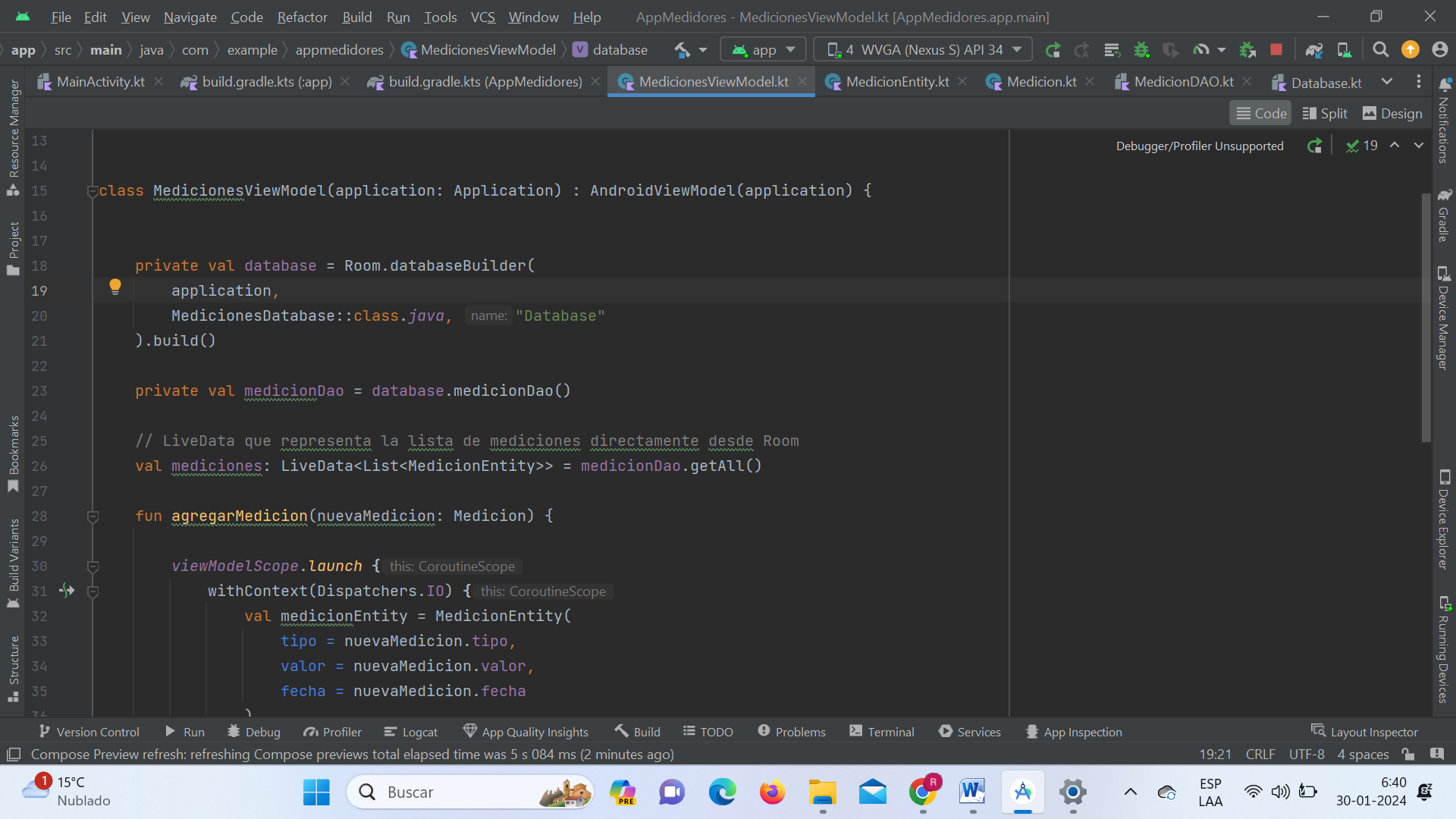


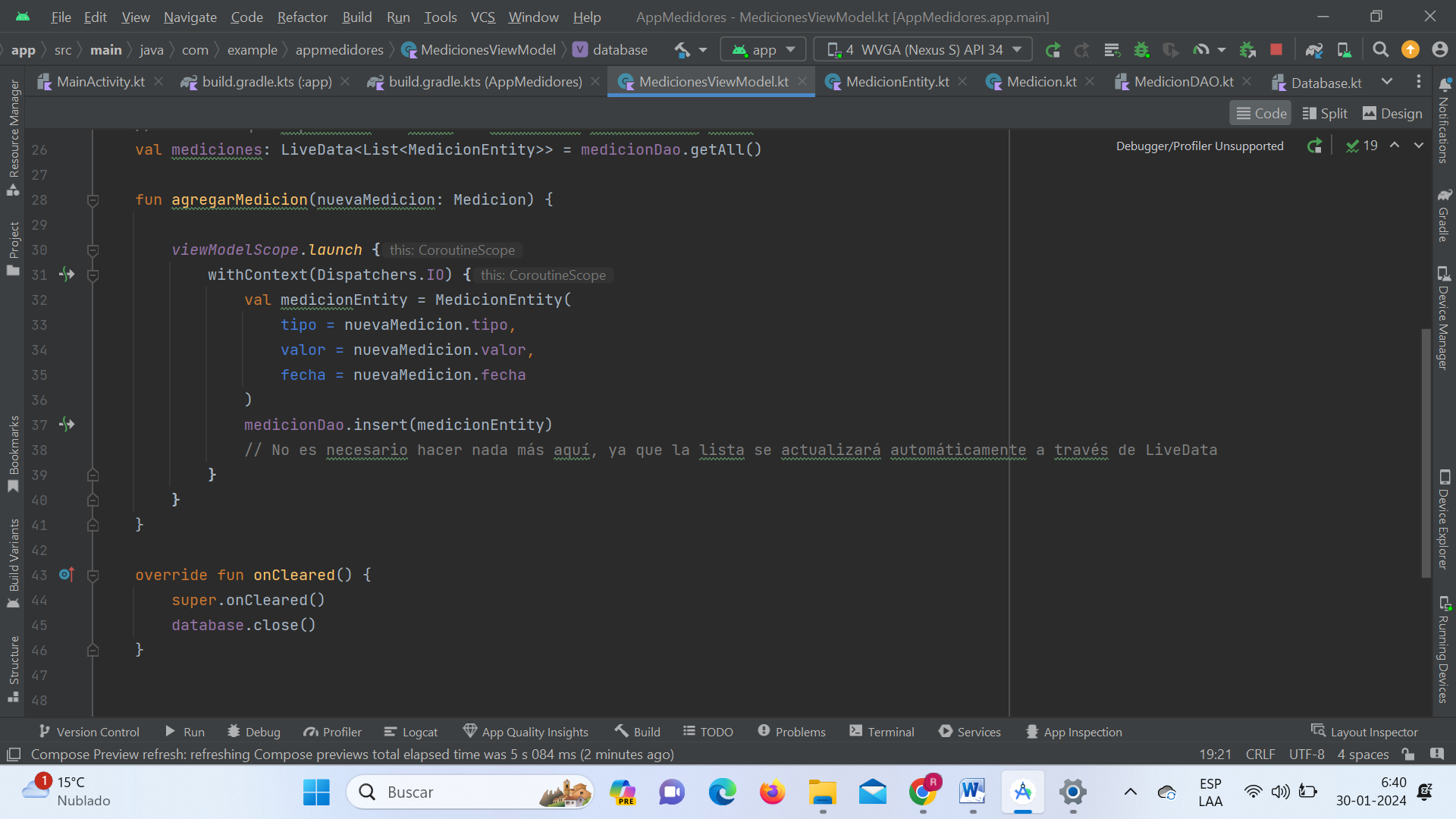


5. Utilice recursos de texto para hacer que los botones y etiquetas soporten inglés y español.

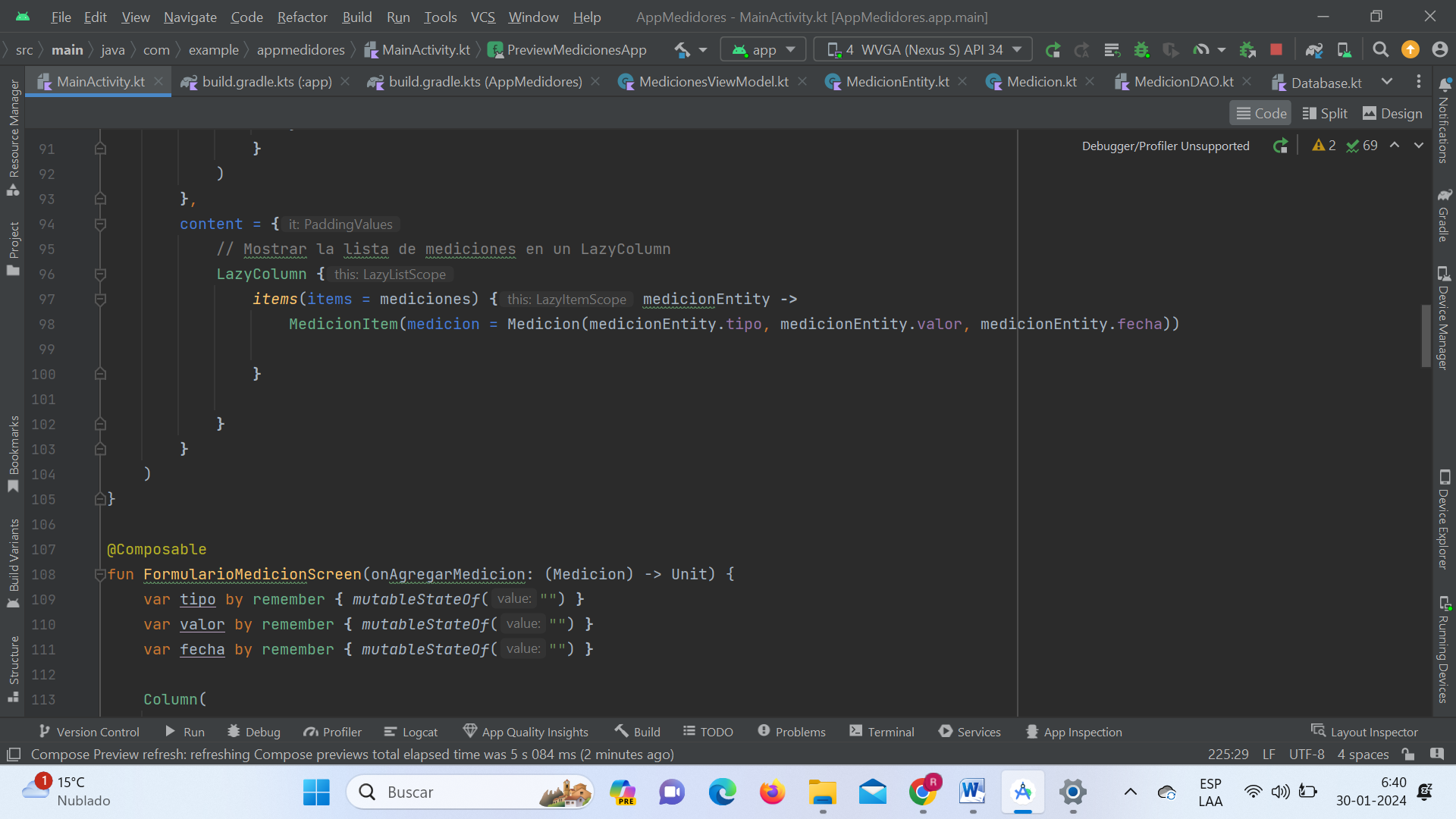
Me faltó.

6. Implementar al menos un ViewModel para gestionar la lógica de presentación y datos.

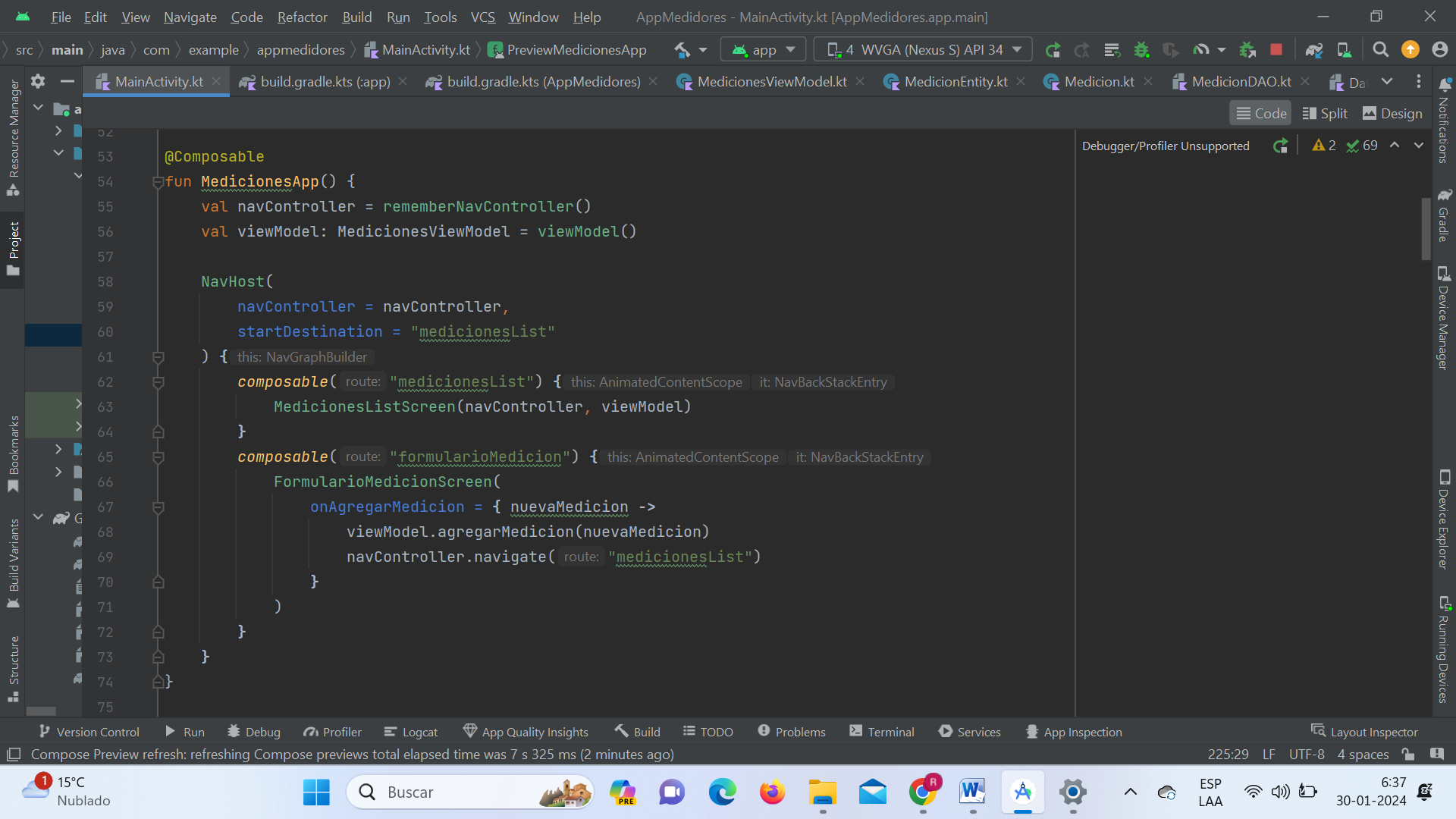




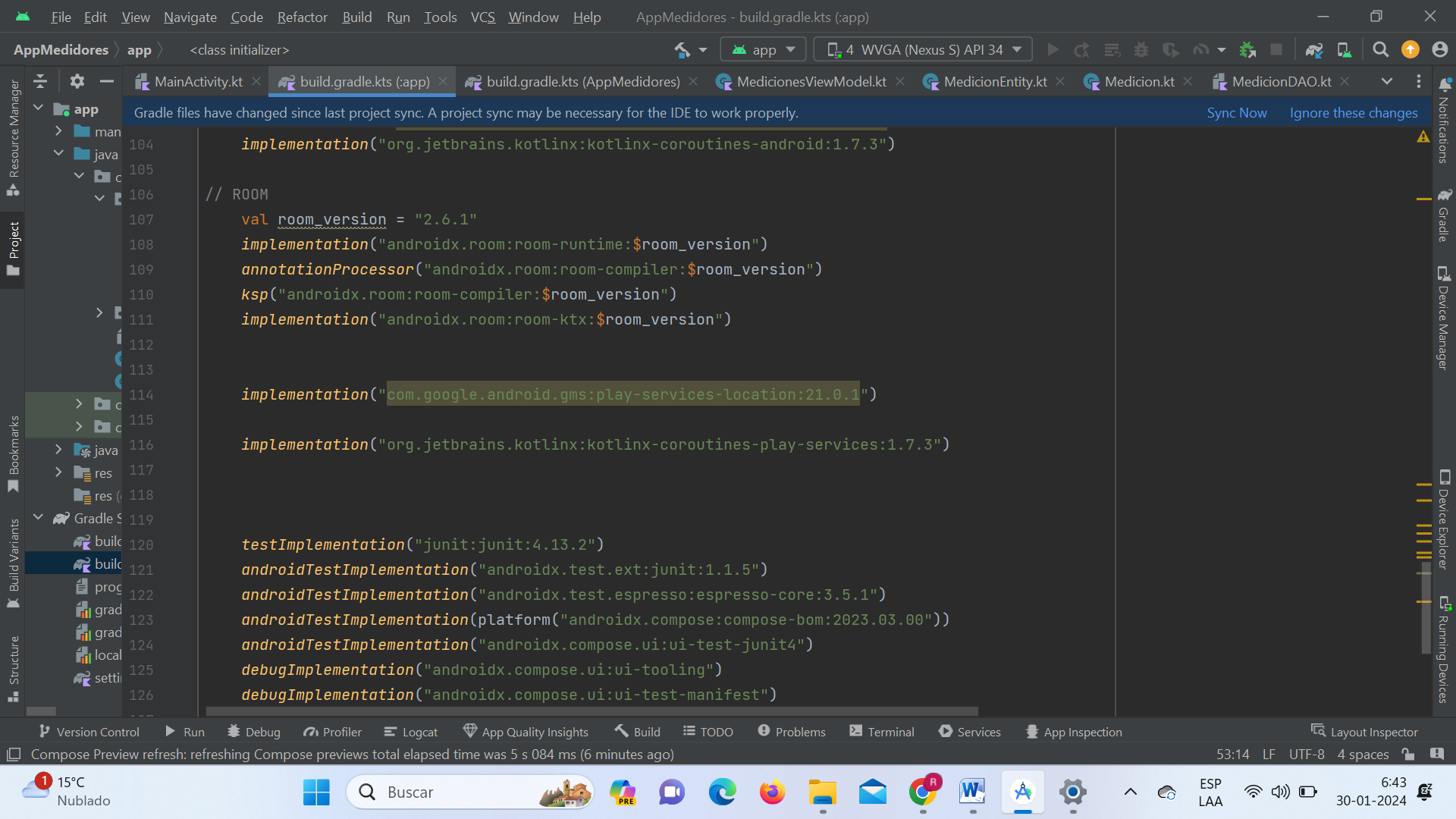
7. Utilizar listas dinámicas, como LazyColumn o RecyclerView, para mostrar los registros en la pantalla de listado



8. Utilice el componente de navegación para moverse de una pantalla a otra



9. Utilizar la biblioteca ROOM para persistir los datos en una base de datos SQLite, asegurando que los registros se mantengan incluso después de cerrar la aplicación.

10. Emplear corrutinas donde sea necesario para evitar bloquear el hilo principal de la aplicación y garantizar una experiencia de usuario fluida.

No logré finalizar esta parte.

Dificultades que tuve. Al igual que en híbrida me dificultó en demasía la incompatibilidad de dependencias ya que me hizo perder mucho tiempo y a veces fallaba un detalle y se desmoronaba todo. Igualmente seguiré profundizando en este apartado de persistencia y coroutines ya que es muy positivo para el desempeño general de la profesión.