

Reporte de Perfilamiento de la Aplicación

1. Captura de Datos.

Para poder establecer esta comparación, las instrucciones a seguir con ambos proyectos son las siguientes:

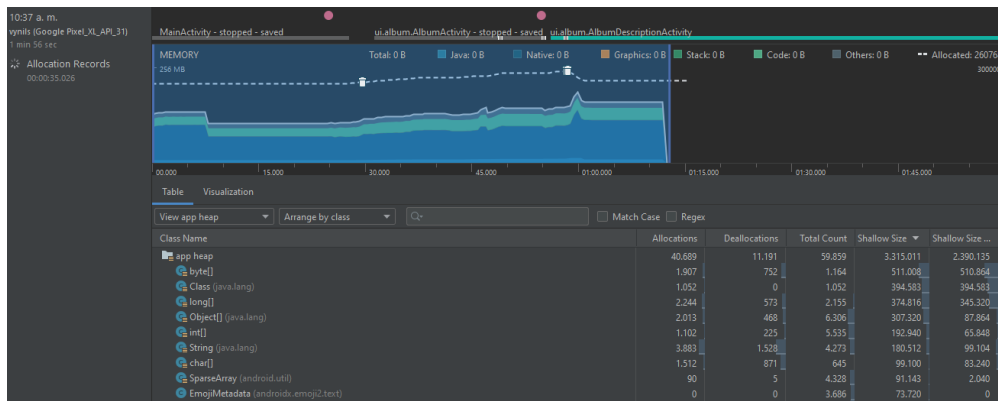
- Ejecutar el perfilamiento con el botón "*profile 'app'*". En caso de que la información de los coleccionistas no cargue al inicio, volver a presionar este botón.
- Seleccionar *Listar Albumes*. Esperar un tiempo a que el *profiler* registre el evento del clic y la creación del fragmento nuevo.
- Seleccionar el primer álbum. Esperar un tiempo a que se registre el evento y la interfaz muestre el comentario.

Se lleva un registro del tiempo en que ejecutó cada evento.

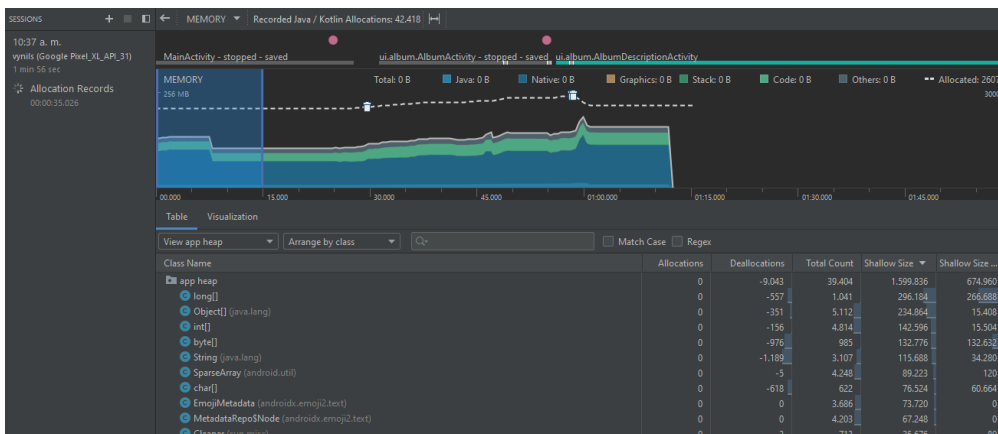
➤ Perfilar Rendimiento de la Aplicación – Previo Micro Optimizaciones

Para capturar el perfil de rendimiento de la app en el estado inicial se ejecutaron las pruebas desde la rama *feature/HU06_frontend* ya que en esta rama se tenía el desarrollo completo de la Iteración #2.

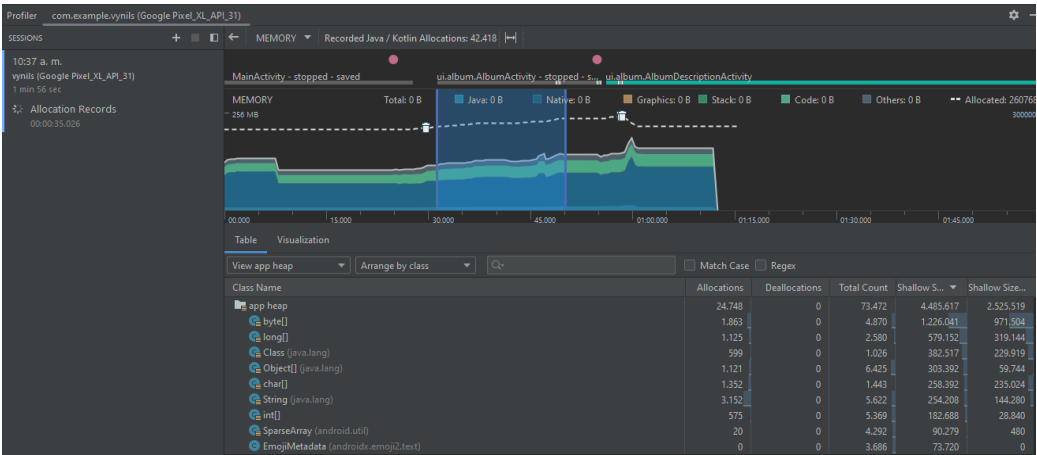
Consumo de Memoria durante todo el test.



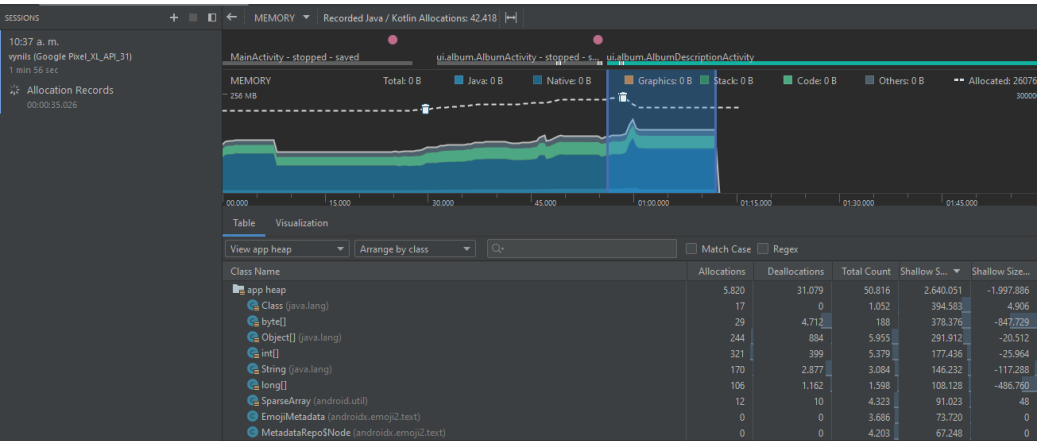
Consumo de Memoria Main Activity.



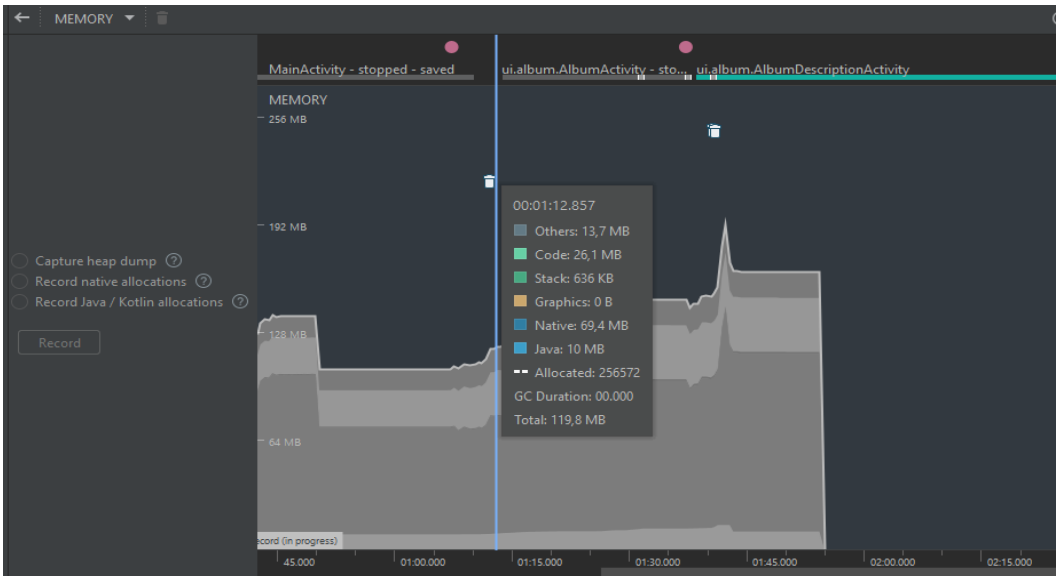
Consumo de Memoria Album List Activity.



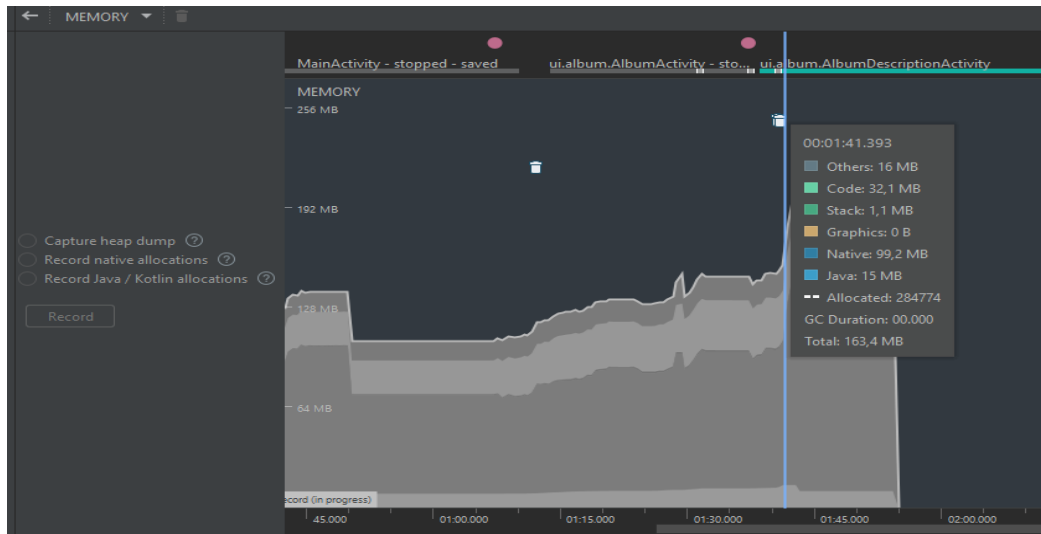
Consumo de Memoria Album Detail Activity.



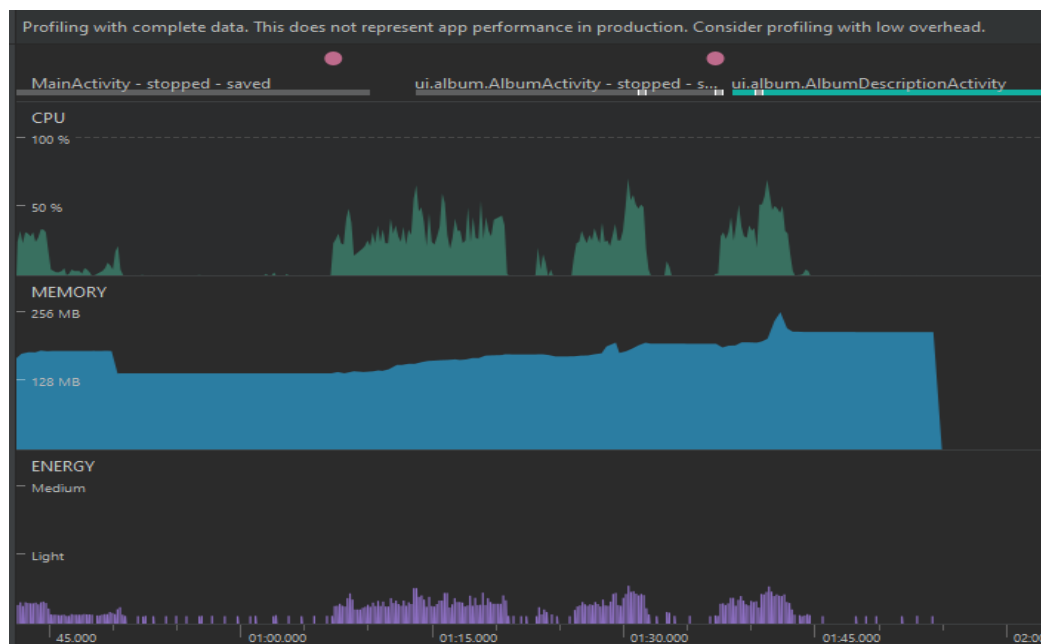
Porcentajes Consumo de Memoria – Inicio Album Activity.



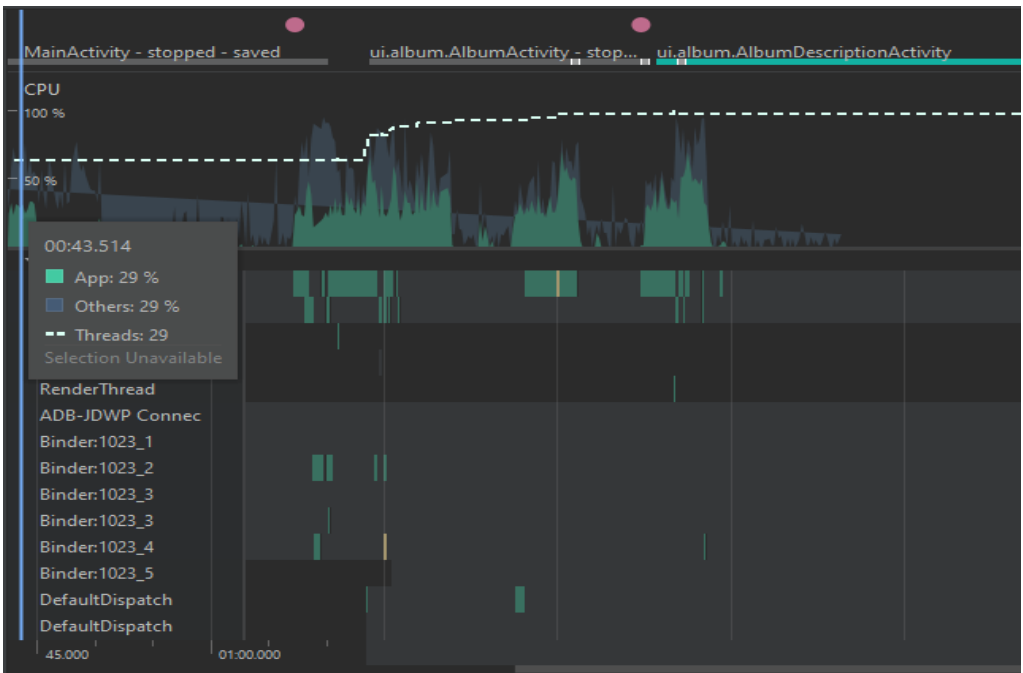
Porcentajes Consumo de Memoria – Inicio Album Detail Activity.



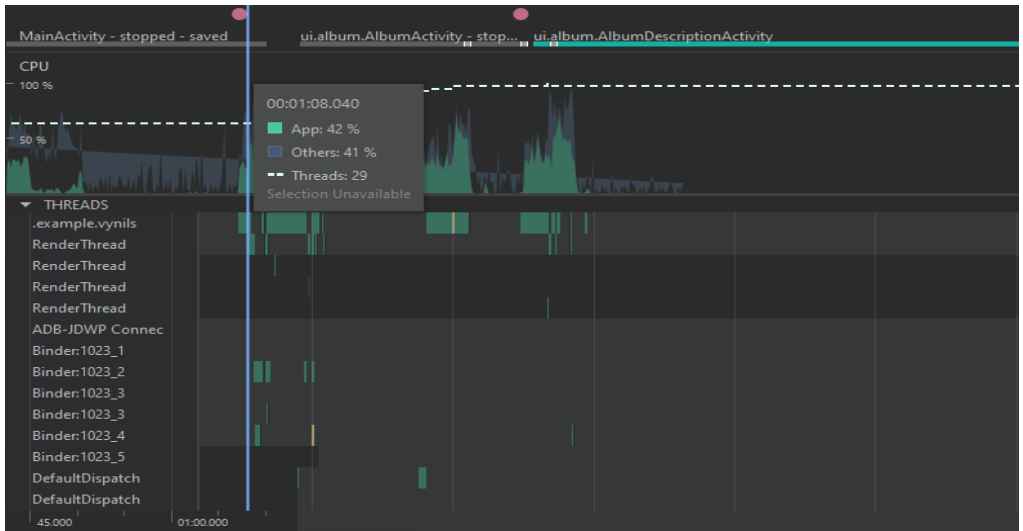
Data Completa Profiler.



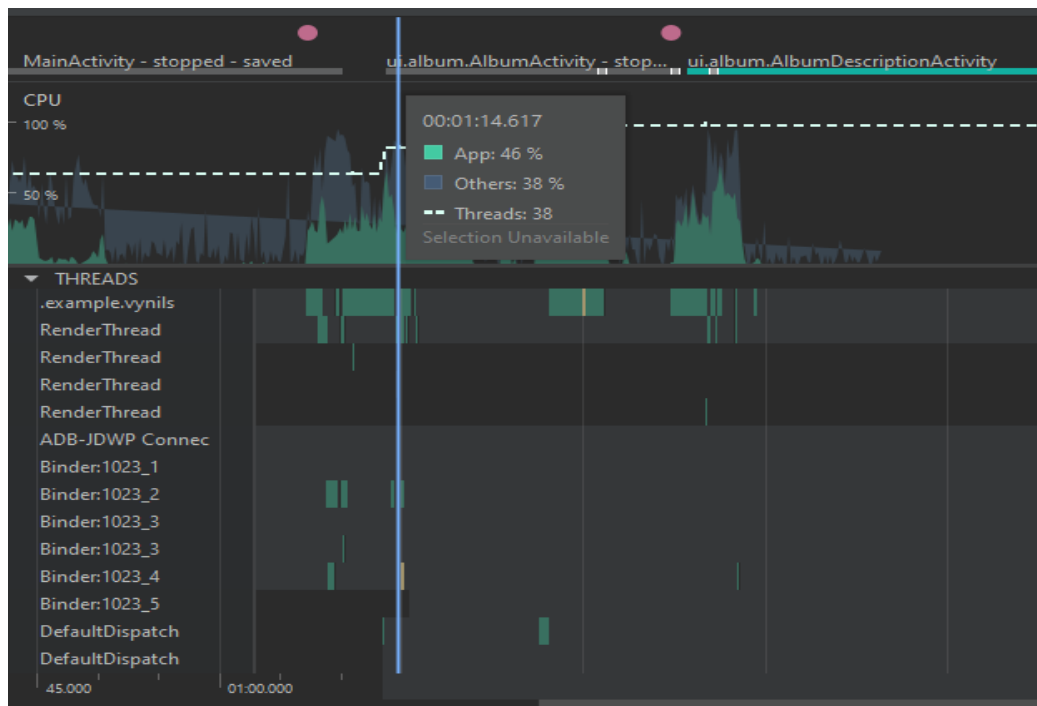
Consumo CPU –Inicio Main Activity.



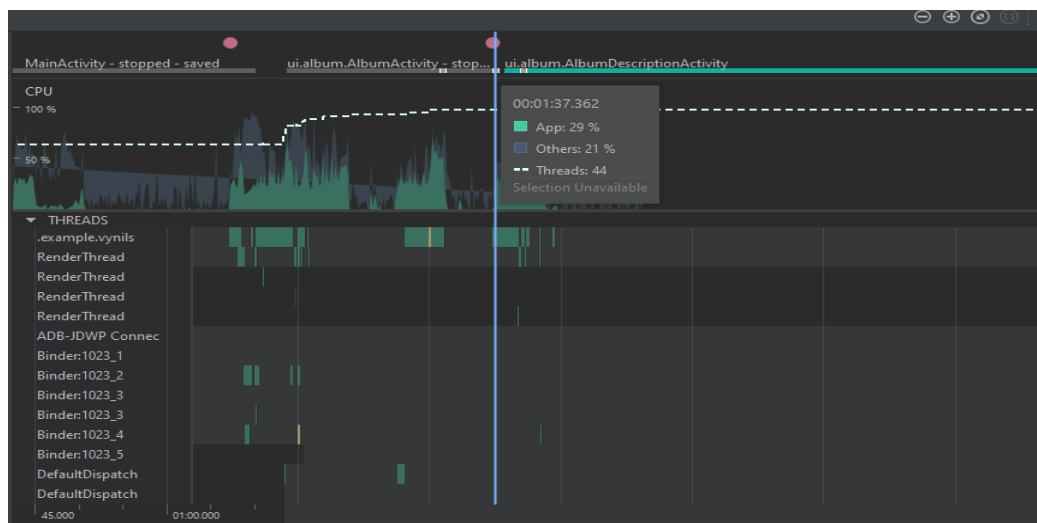
Consumo CPU –Finalización Main Activity.



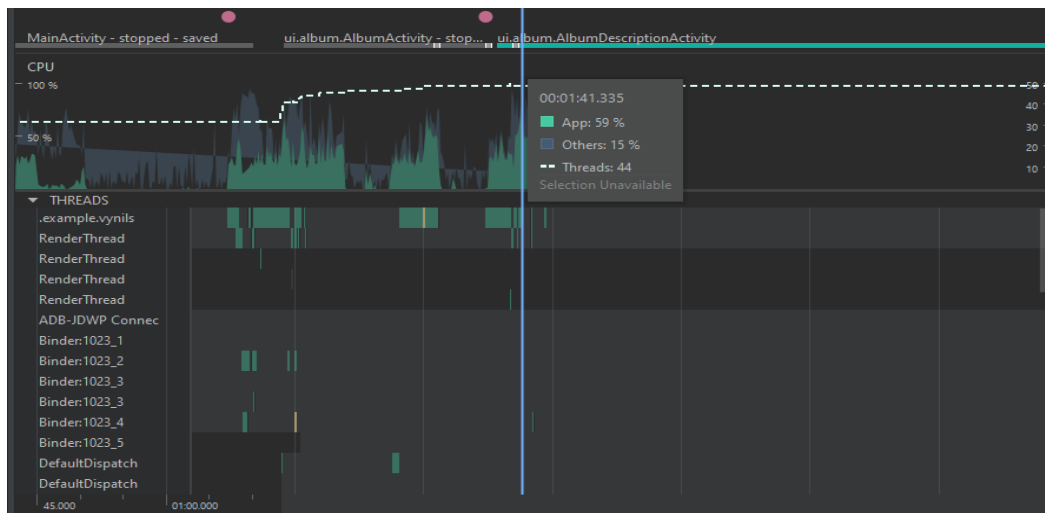
Consumo CPU –Inicio Album Activity.



Consumo CPU –Finalización Album Activity.



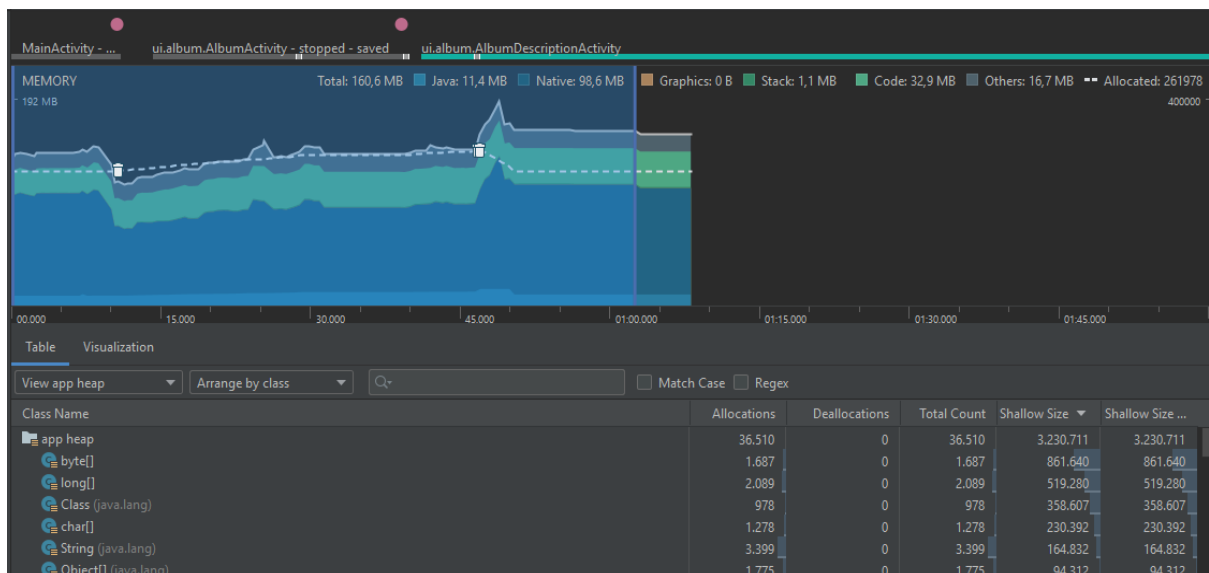
Consumo CPU –Inicio Album Detail Activity.



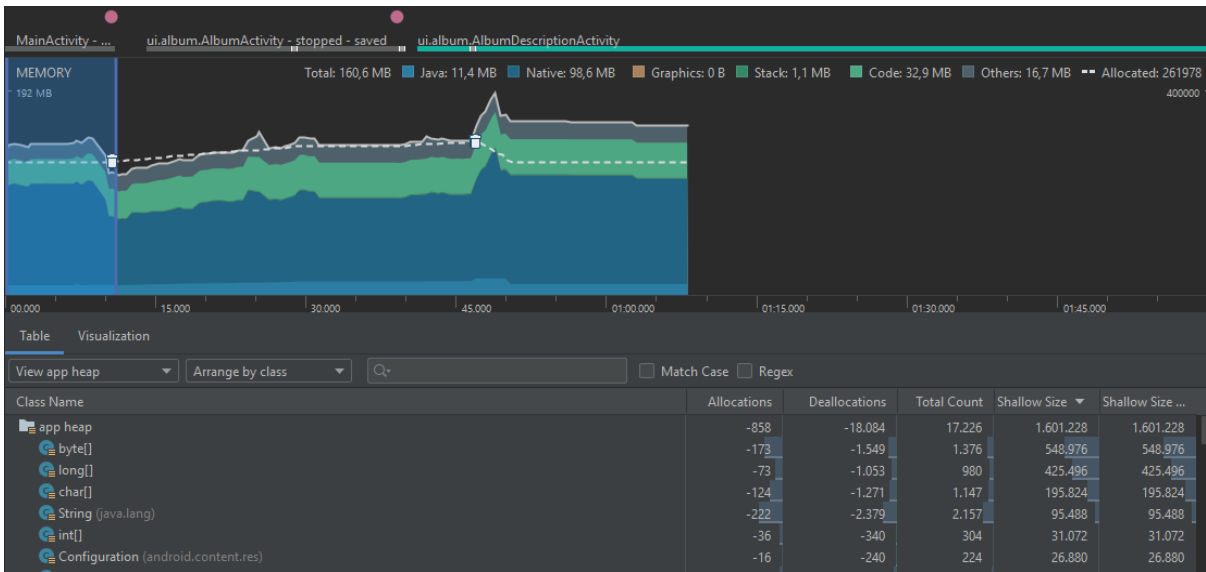
➤ Perfilar Rendimiento de la App – Post Micro Optimizaciones

La versión de la aplicación despues de realizar las micro-optimizaciones está disponible en la rama develop. Desde esa rama se corrió nuevamente el perfil de rendimiento de la aplicación y se obtuvieron los siguientes resultados:

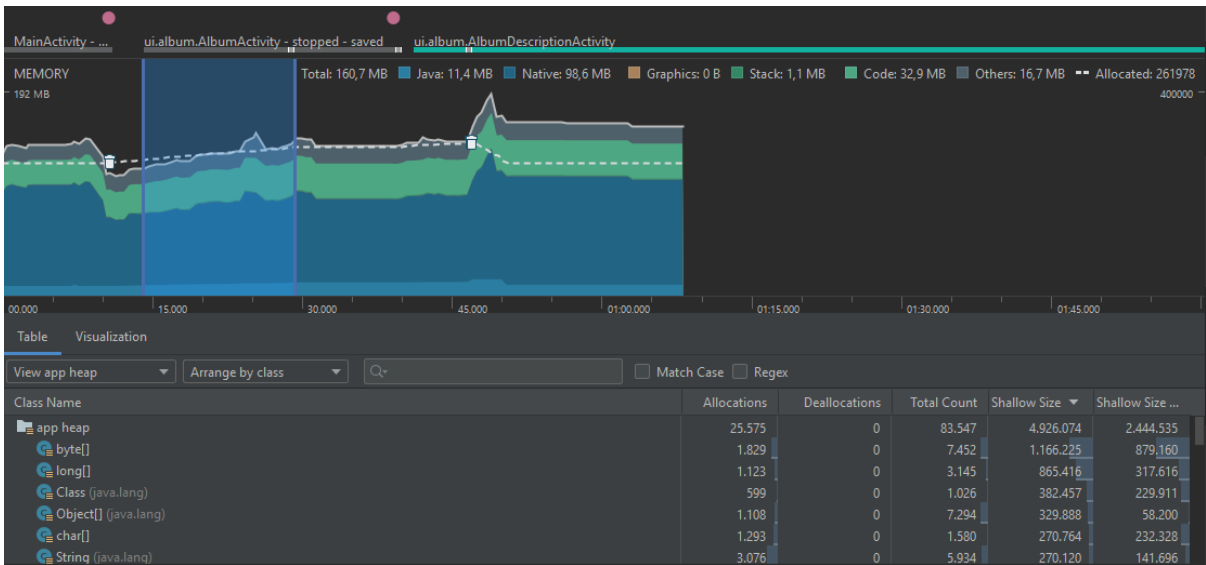
Consumo de Memoria durante todo el test.



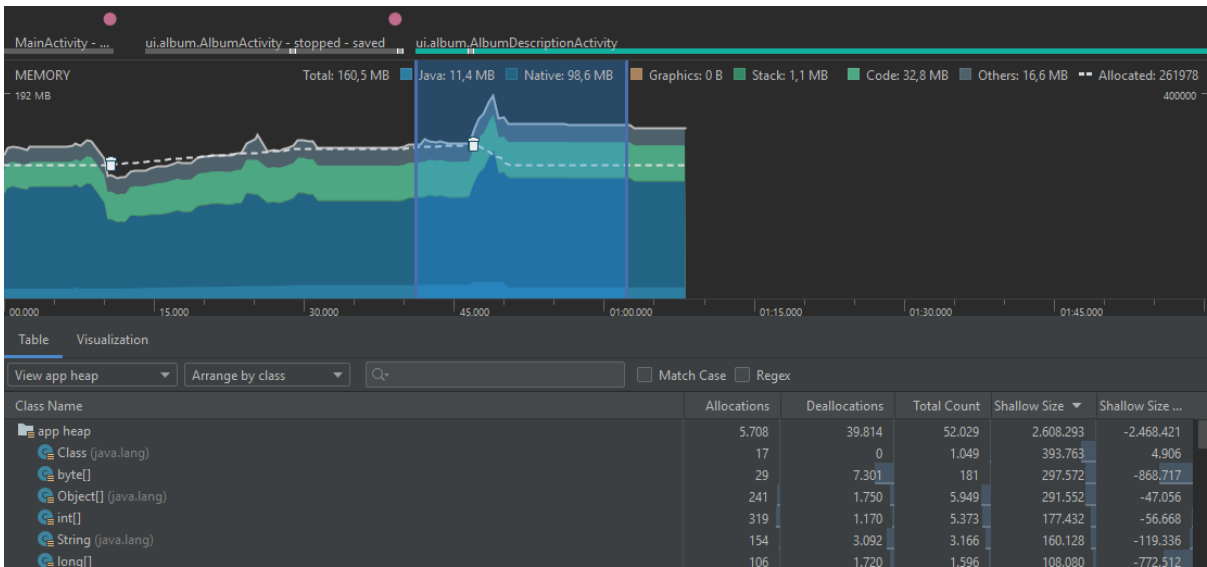
Consumo de Memoria Main Activity.



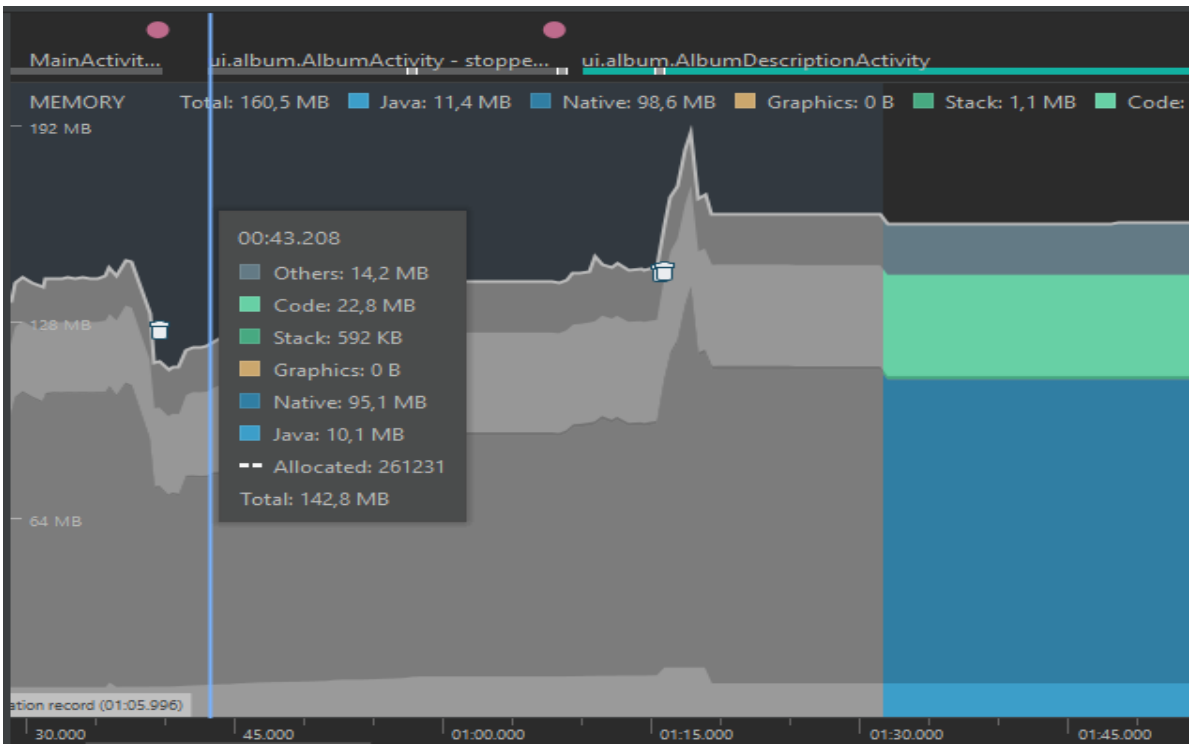
Consumo de Memoria Album List Activity.



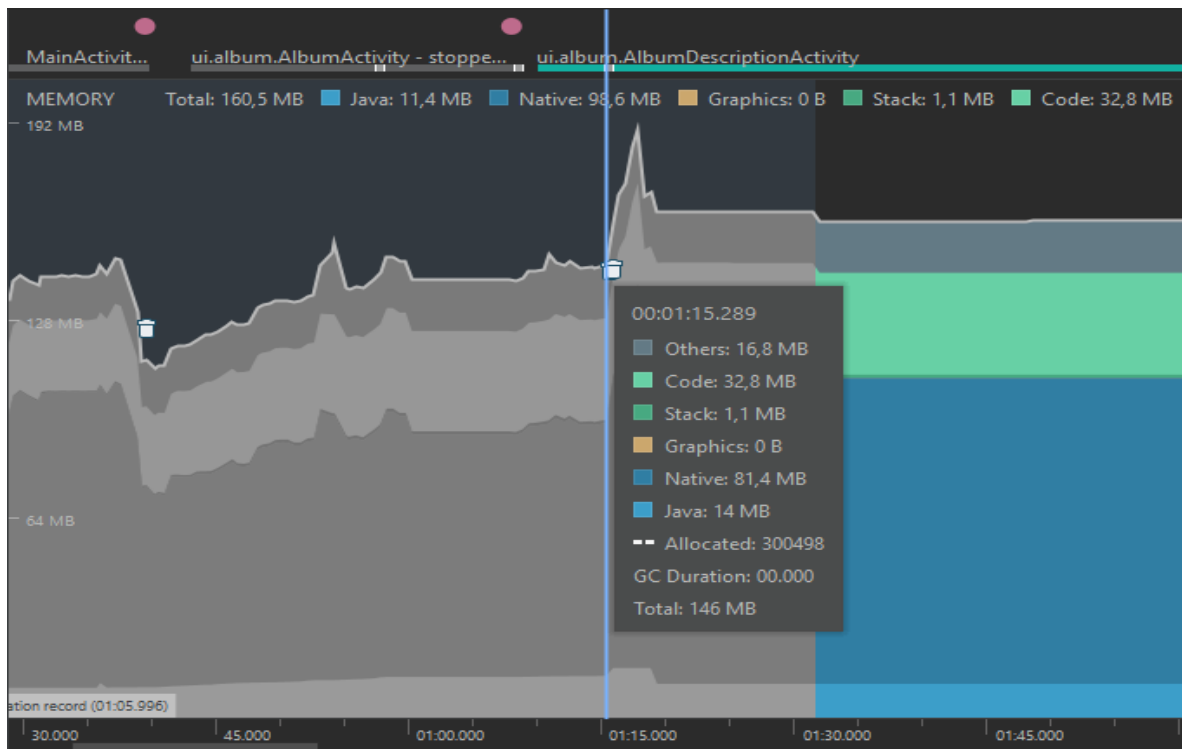
Consumo de Memoria Album Detail Activity.



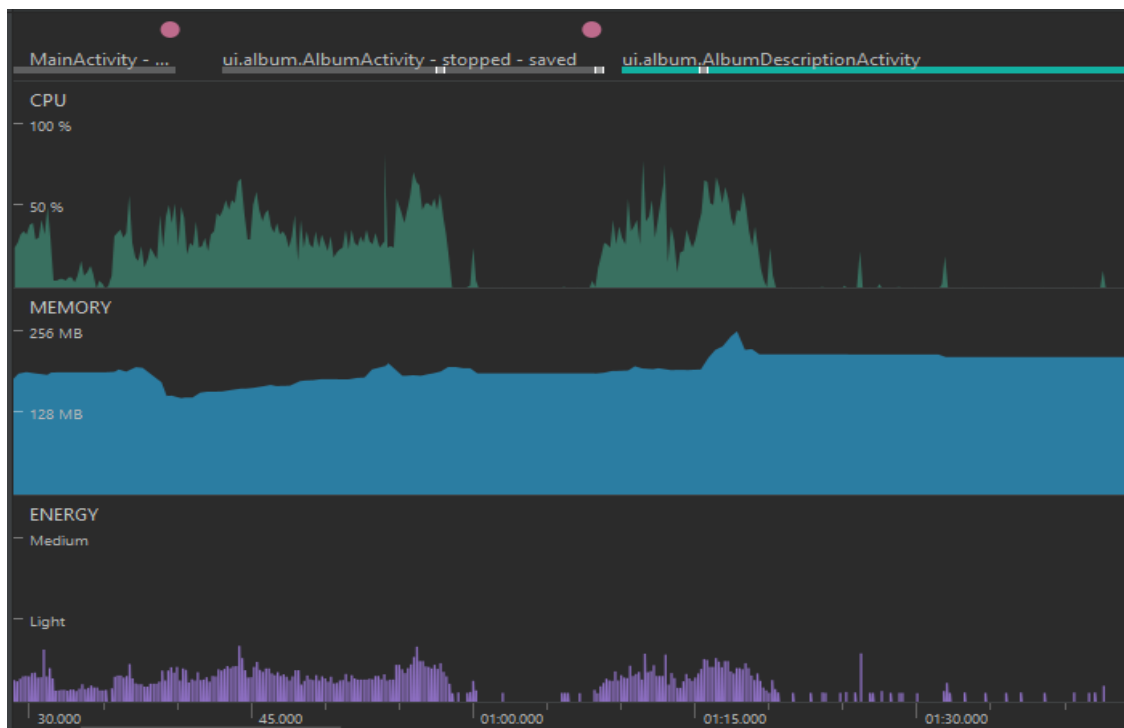
Porcentajes Consumo de Memoria – Inicio Album Activity.



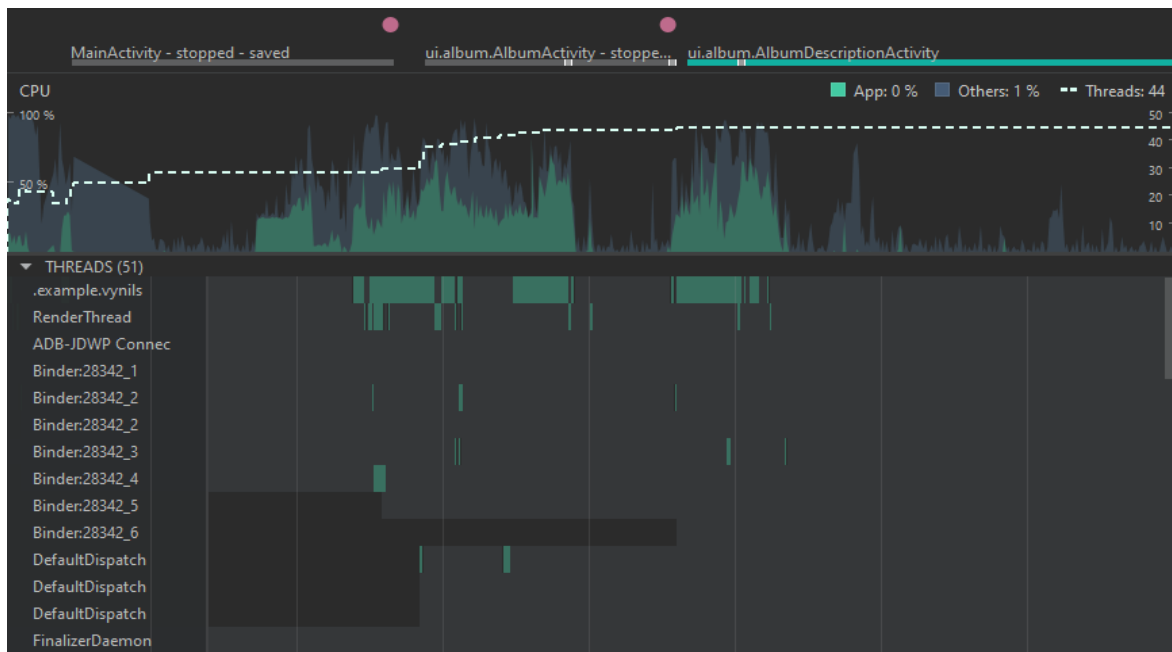
Porcentajes Consumo de Memoria – Inicio Album Detail Activity.



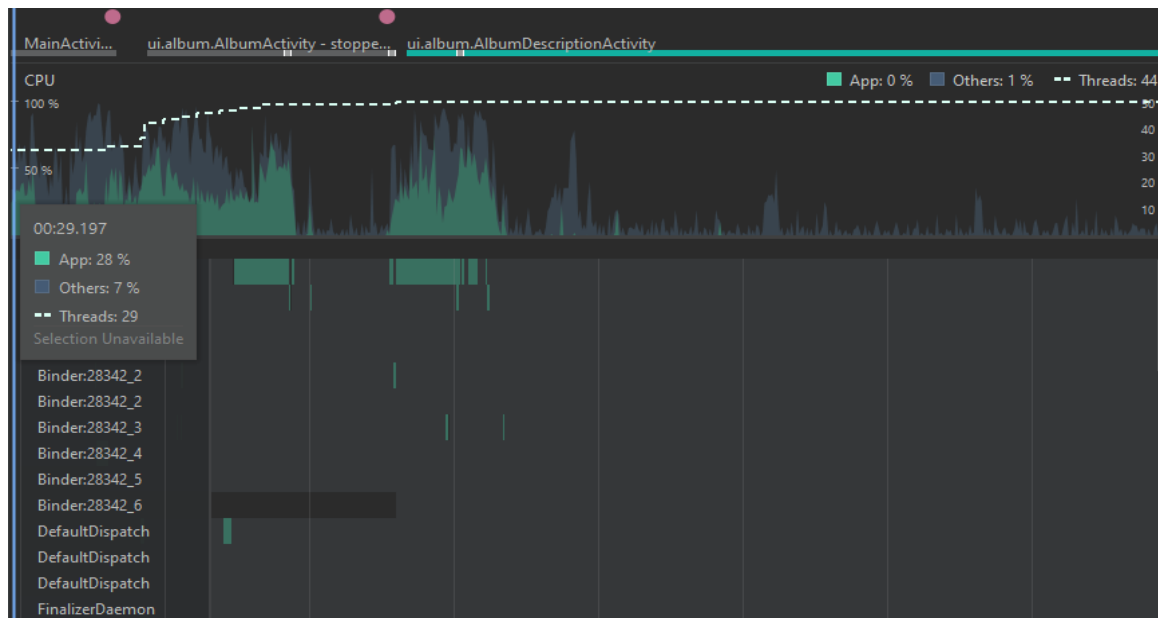
Data Completa Profiler.



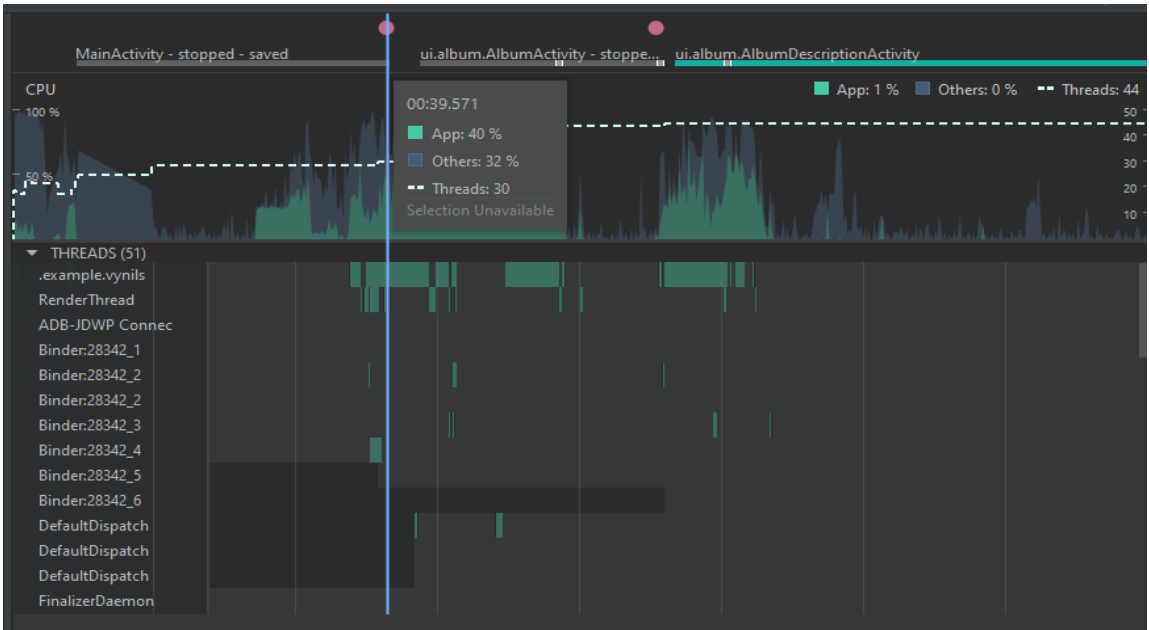
Consumo de CPU durante todo el test.



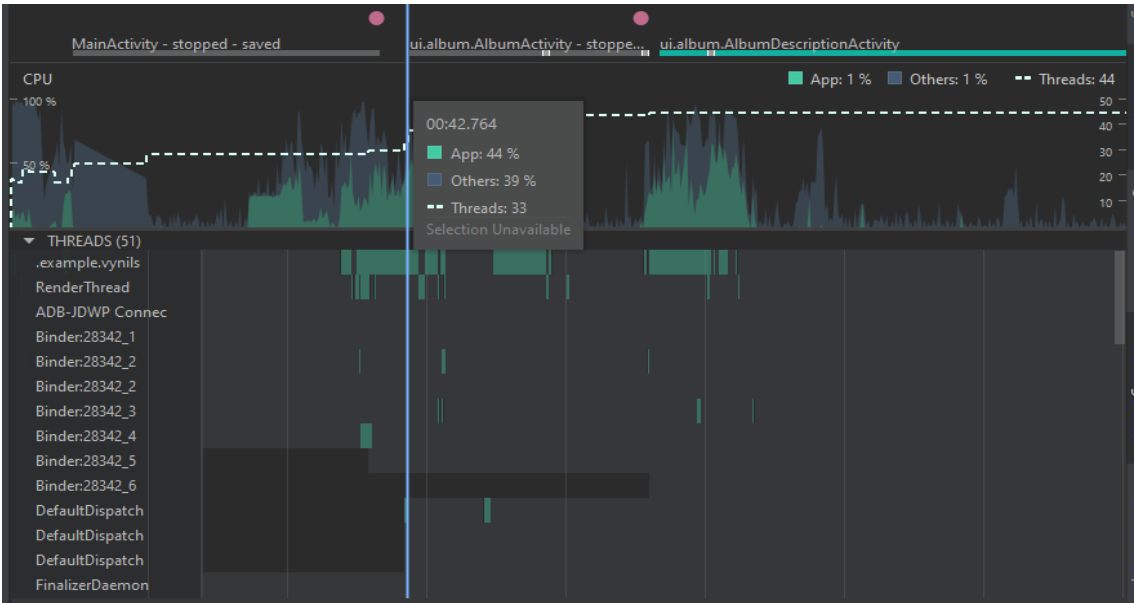
Consumo CPU –Inicio Main Activity.



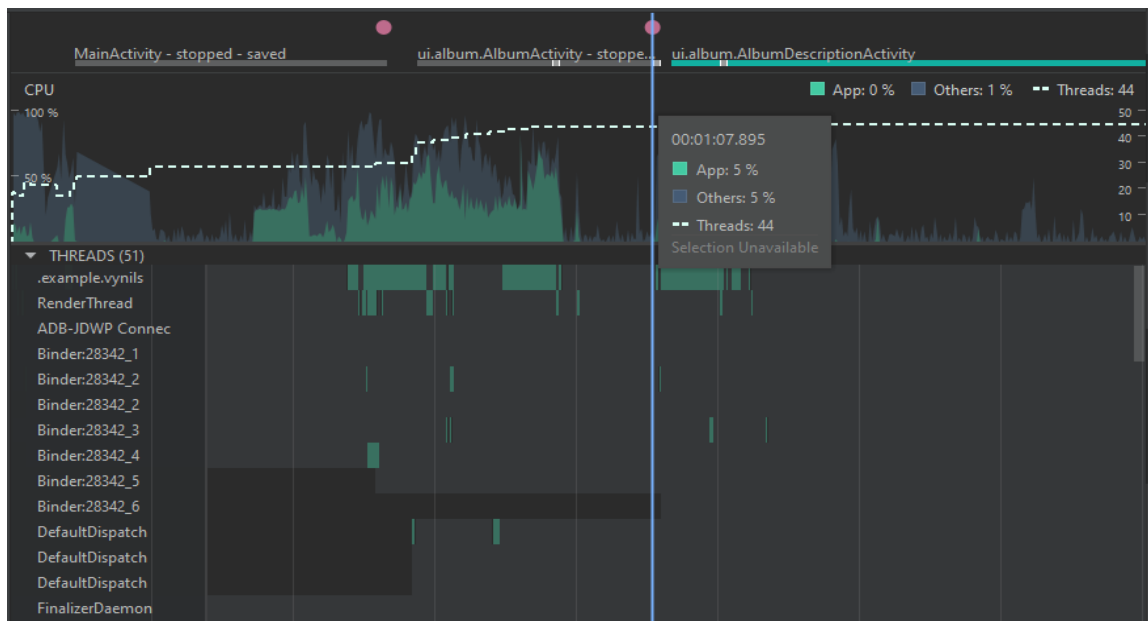
Consumo CPU –Finalización Main Activity.



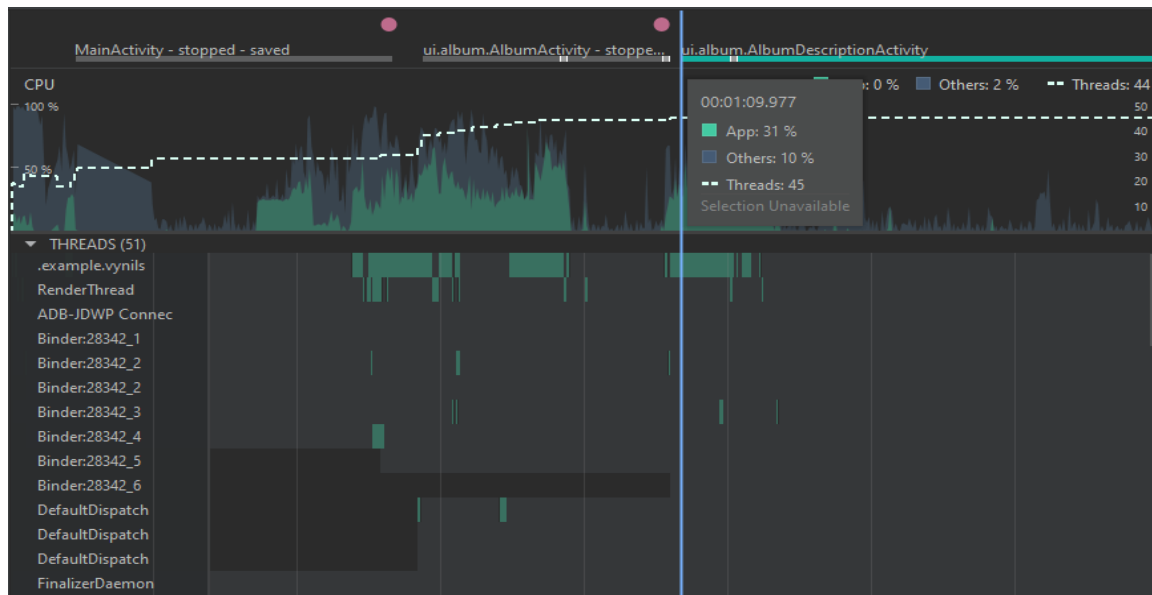
Consumo CPU –Inicio Album Activity.



Consumo CPU –Finalización Album Activity.



Consumo CPU –Inicio Album Detail Activity.



2. Análisis de Resultados

Los resultados obtenidos en tabla de App Heap muestran que después de realizar las micro-optimizaciones a la aplicación el número de Allocations registradas se redujo en 4.179. Esta disminución en número de Allocations mejora el desempeño de la aplicación ya que se reduce el trabajo realizado por el sistema en el proceso de creación de nuevos objetos, así como el esfuerzo y tiempo requerido para la liberación de dichos objetos más tarde (garbage collectors).

El comparativo del App Heap Total Count previo y post micro-optimizaciones permitió conocer que este parámetro se redujo en 23.349. La disminución en el App Heap permite reducir la probabilidad de ocurrencia de OutOfMemoryErrors mejorando de esta forma el desempeño de la aplicación y la experiencia de usuario.

Finalmente, la revisión de los resultados del consumo de CPU muestran una reducción del 2% al inicializar la Actividad Listar Album y una reducción del consumo de CPU de 28% durante la inicialización de la Actividad Detalle Album. Esta disminución en el consumo de recursos favorece el desempeño de la aplicación.