Android App Lifecycle Research: Understanding User Operations and State Transitions in Hot vs Cold Starts

User Operation Classification

graph TB A[Android User Operation Scenarios] --> B[Return to Home] A --> C[App Switching] A --> D[App Termination] A --> E[Page Navigation] A --> F[System Functions] B --> B1[Press Home Button] B --> B2[Gesture Swipe to Home] C --> C1[Open Other Apps] C --> C2[Task Manager App Switch] C --> C3[Notification Tap Navigation] C --> C4[Share Function Navigation] D --> D1[Task Manager Swipe Up Close] D --> D2[Task Card Swipe Left/Right Close] D --> D3[Force Stop in Settings] E --> E1[In-App Back Button Exit] E --> E2[In-App Return to Previous Page] E --> E3[Gesture Back Exit App] F --> F1[Pull Down Notification Panel] F --> F2[Screen Rotation] F --> F3[Enter Split Screen Mode] F --> F4[Return from Settings]

Detailed Operations and Lifecycle Impact

A. Return to Home Operations

A1. Press Home Button to Return to Desktop

User Operation: Press Home button to return to desktop

```
Lifecycle Sequence: onPause → onStop
App State: Enter background, remain in memory
Process State: Process continues running
Characteristics: App stays active, ready for quick recovery
Startup Type: Hot start when returning
```

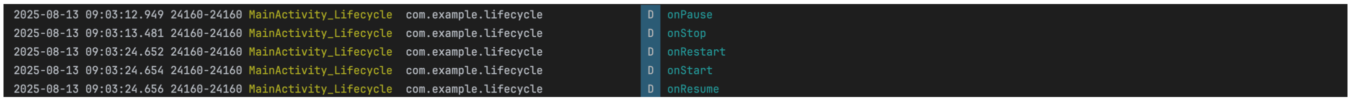
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Press Home] --> C[onPause] --> D[onStop] --> E[Activity background] E --> F[Return to app] --> G[onRestart] --> H[onStart] --> I[onResume] --> J[Activity running] style A fill:#90EE90 style E fill:#FFB6C1 style J fill:#90EE90

Log Output Examples:

```
# Press Home button
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop

# Return to app
MainActivity_Lifecycle: onRestart
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:03:12.949 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onPause
2025-08-13 09:03:13.481 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onStop
2025-08-13 09:03:24.652 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onRestart
2025-08-13 09:03:24.654 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onStart
2025-08-13 09:03:24.656 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onResume
```

A2. Gesture Swipe to Return to Desktop

User Operation: Bottom short swipe to return to desktop

```
Lifecycle Sequence: onPause → onStop
App State: Enter background, remain in memory
Process State: Process continues running
Characteristics: Identical effect to Home button
Startup Type: Hot start when returning
```

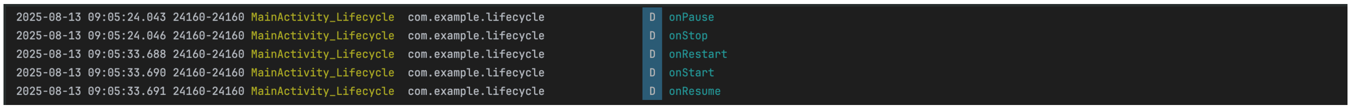
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Gesture swipe] --> C[onPause] --> D[onStop] --> E[Activity background] E --> F[Tap icon] --> G[onRestart] --> H[onStart] --> I[onResume] --> J[Activity running] style A fill:#90EE90 style E fill:#FFB6C1 style J fill:#90EE90

Log Output Examples:

```
# Gesture swipe to home
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop

# Return to app
MainActivity_Lifecycle: onRestart
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:05:24.043 24160-24160 MainActivity_Lifecycle  com.example.lifecycle               D  onPause
2025-08-13 09:05:24.046 24160-24160 MainActivity_Lifecycle  com.example.lifecycle               D  onStop
2025-08-13 09:05:33.688 24160-24160 MainActivity_Lifecycle  com.example.lifecycle               D  onRestart
2025-08-13 09:05:33.690 24160-24160 MainActivity_Lifecycle  com.example.lifecycle               D  onStart
2025-08-13 09:05:33.691 24160-24160 MainActivity_Lifecycle  com.example.lifecycle               D  onResume
```

B. App Switching Operations

B1. Open Other Apps

User Operation: Tap other app icons on desktop

```
Lifecycle Sequence: onPause → onStop
App State: Current app to background, new app to foreground
Process State: Two processes running simultaneously
Characteristics: Current app paused, state preserved
Startup Type: New app may be cold or hot start
```

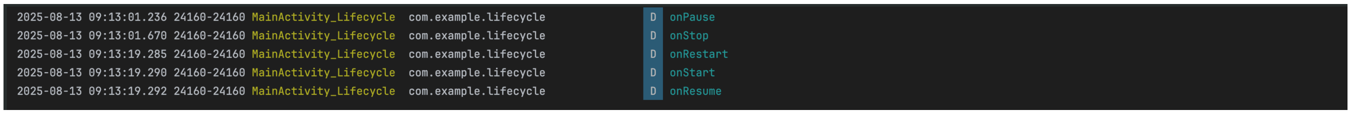
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Tap other App] --> C[onPause] --> D[onStop] --> E[Background state] E --> F[Return] --> G[onRestart] --> H[onStart] --> I[onResume] --> J[Activity running] style A fill:#90EE90 style E fill:#FFB6C1 style J fill:#90EE90

Log Output Examples:

```
# Open other app
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop

# Return to app
MainActivity_Lifecycle: onRestart
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:13:01.236 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onPause
2025-08-13 09:13:01.670 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onStop
2025-08-13 09:13:19.285 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onRestart
2025-08-13 09:13:19.290 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onStart
2025-08-13 09:13:19.292 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onResume
```

B2. Task Manager App Switch

User Operation: Tap app card in task manager

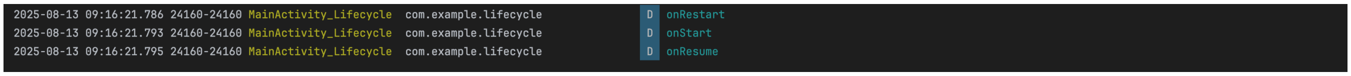
```
Lifecycle Sequence: onRestart → onStart → onResume
App State: Resume from background to foreground
Process State: Process remains unchanged
Characteristics: Quick recovery, complete state preservation
Startup Type: Hot start
```

Lifecycle Flow Diagram:

graph LR A[Activity background] --> B[Tap task card] --> C[onRestart] --> D[onStart] --> E[onResume] --> F[Activity running] style A fill:#FFB6C1 style F fill:#90EE90

Log Output Examples:

```
# Return from task manager
MainActivity_Lifecycle: onRestart
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:16:21.786 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onRestart
2025-08-13 09:16:21.793 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onStart
2025-08-13 09:16:21.795 24160-24160 MainActivity_Lifecycle  com.example.lifecycle                D  onResume
```

B3. Notification Tap Navigation

User Operation: Tap app notification in notification panel

```
Lifecycle Sequence:
- If app in background: onRestart → onStart → onResume
- If app closed: onCreate → onStart → onResume
App State: App brought to foreground, may create new Activity
Process State: May reuse process or create new process
Characteristics: May launch specific Activity or deep link
Startup Type: Depends on current app state
```

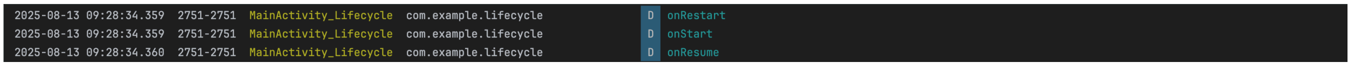
Lifecycle Flow Diagram:

graph LR A[Background state] --> B[Tap notification] --> C[onRestart/onCreate] --> D[onStart] --> E[onResume] --> F[Activity running] style A fill:#FFB6C1 style F fill:#90EE90

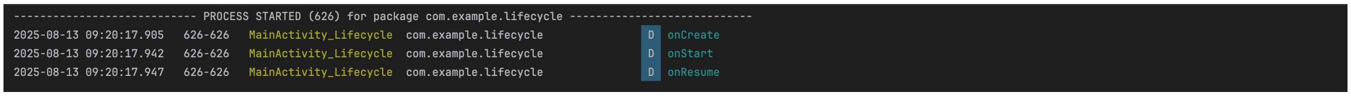
Log Output Examples:

```
# App in background scenario (Press Home and return soon to avoid Activity being killed.)
MainActivity_Lifecycle: onRestart
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume

# App closed scenario
SplashActivity_Lifecycle: onCreate - Start Type: Cold Start
```



```
2025-08-13 09:28:34.359  2751-2751  MainActivity_Lifecycle  com.example.lifecycle              D  onRestart
2025-08-13 09:28:34.359  2751-2751  MainActivity_Lifecycle  com.example.lifecycle              D  onStart
2025-08-13 09:28:34.360  2751-2751  MainActivity_Lifecycle  com.example.lifecycle              D  onResume
```



```
--------------------------- PROCESS STARTED (626) for package com.example.lifecycle ---------------------------
2025-08-13 09:20:17.905   626-626   MainActivity_Lifecycle  com.example.lifecycle              D  onCreate
2025-08-13 09:20:17.942   626-626   MainActivity_Lifecycle  com.example.lifecycle              D  onStart
2025-08-13 09:20:17.947   626-626   MainActivity_Lifecycle  com.example.lifecycle              D  onResume
```

B4. Share Function Navigation

User Operation: Other apps share content to current app

```
Lifecycle Sequence:
- If app in background: onRestart → onStart → onResume
- If app closed: onCreate → onStart → onResume
App State: App awakened, usually launches specific Activity to handle share
Process State: May reuse process or create new process
Characteristics: Usually launches new Activity to handle Intent
Startup Type: Depends on current app state
```

Lifecycle Flow Diagram:

graph LR A[Other App share] --> B[Select target App] --> C[Wake/Start] --> D[ShareActivity onCreate] --> E[onStart] --> F[onResume] --> G[Handle share] style A fill:#e3f2fd style G fill:#90EE90

Log Output Examples:

```
# New Activity launched to handle share
ShareActivity_Lifecycle: onCreate
ShareActivity_Lifecycle: onStart
ShareActivity_Lifecycle: onResume
```

(Debug image unavailable – ShareActivity not yet implemented)

C. App Termination Operations

C1. Task Manager Swipe Up Close

User Operation: Swipe up app card in task manager

```
Lifecycle Sequence: onPause → onStop → onDestroy
App State: App terminated by system
Process State: Process killed
Characteristics: Force terminate, clean all resources
Startup Type: Next startup will be cold start
```

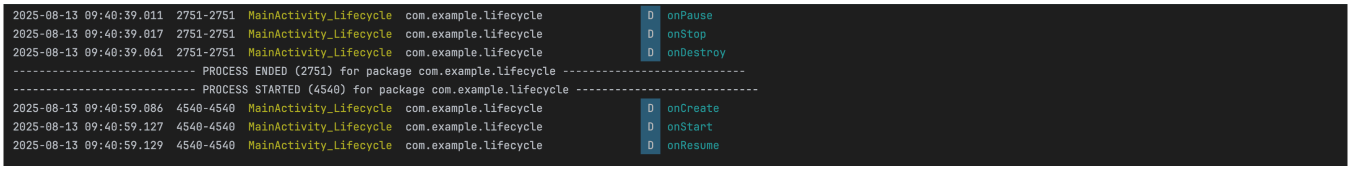
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Swipe up close] --> C[onPause] --> D[onStop] --> E[onDestroy] --> F[Destroyed] F --> G[Restart] --> H[onCreate] --> I[onStart] --> J[onResume] --> K[Activity running] style A fill:#90EE90 style F fill:#FFA07A style K fill:#90EE90

Log Output Examples:

```
# Swipe up close
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop
MainActivity_Lifecycle: onDestroy

# Restart (Cold start)
SplashActivity_Lifecycle: onCreate - Start Type: Cold Start
MainActivity_Lifecycle: onCreate
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:40:39.011  2751-2751  MainActivity_Lifecycle  com.example.lifecycle               D  onPause
2025-08-13 09:40:39.017  2751-2751  MainActivity_Lifecycle  com.example.lifecycle               D  onStop
2025-08-13 09:40:39.061  2751-2751  MainActivity_Lifecycle  com.example.lifecycle               D  onDestroy
--------------------------- PROCESS ENDED (2751) for package com.example.lifecycle ---------------------------
--------------------------- PROCESS STARTED (4540) for package com.example.lifecycle ---------------------------
2025-08-13 09:40:59.086  4540-4540  MainActivity_Lifecycle  com.example.lifecycle               D  onCreate
2025-08-13 09:40:59.127  4540-4540  MainActivity_Lifecycle  com.example.lifecycle               D  onStart
2025-08-13 09:40:59.129  4540-4540  MainActivity_Lifecycle  com.example.lifecycle               D  onResume
```

C2. Task Card Swipe Left/Right Close

User Operation: Swipe left/right app card in task manager

```
Lifecycle Sequence: Immediate onDestroy
App State: App immediately terminated
Process State: Process immediately killed
Characteristics: Fastest force close method
Startup Type: Next startup will be cold start
```

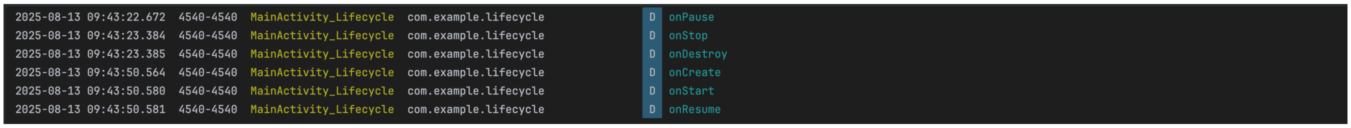
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Swipe left/right] --> C[onDestroy] --> D[Immediate destroy] D --> E[Restart] --> F[onCreate] --> G[onStart] --> H[onResume] --> I[Activity running] style A fill:#90EE90 style D fill:#FFA07A style I fill:#90EE90

Log Output Examples:

```
# Swipe left/right close
MainActivity_Lifecycle: onDestroy

# Restart (Cold start)
SplashActivity_Lifecycle: onCreate - Start Type: Cold Start
MainActivity_Lifecycle: onCreate
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:43:22.672  4540-4540  MainActivity_Lifecycle  com.example.lifecycle            D  onPause
2025-08-13 09:43:23.384  4540-4540  MainActivity_Lifecycle  com.example.lifecycle            D  onStop
2025-08-13 09:43:23.385  4540-4540  MainActivity_Lifecycle  com.example.lifecycle            D  onDestroy
2025-08-13 09:43:50.564  4540-4540  MainActivity_Lifecycle  com.example.lifecycle            D  onCreate
2025-08-13 09:43:50.580  4540-4540  MainActivity_Lifecycle  com.example.lifecycle            D  onStart
2025-08-13 09:43:50.581  4540-4540  MainActivity_Lifecycle  com.example.lifecycle            D  onResume
```

C3. Force Stop in Settings

User Operation: Settings → App Management → Force Stop

```
Lifecycle Sequence: Immediate onDestroy (may skip other lifecycle methods)
App State: App force terminated
Process State: Process immediately terminated, clean all resources
Characteristics: Most thorough close method, cleans all background tasks
Startup Type: Next startup will be cold start
```

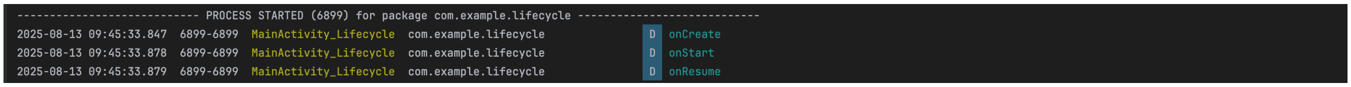
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Settings force stop] --> C[Process killed immediately] --> D[Force terminate] D --> E[Restart] --> F[onCreate] --> G[onStart] --> H[onResume] --> I[Activity running] style A fill:#90EE90 style D fill:#FF6B6B style I fill:#90EE90

Log Output Examples:

```
# Force stop
MainActivity_Lifecycle: (no lifecycle logs - process killed directly)

# Restart (Cold start)
SplashActivity_Lifecycle: onCreate - Start Type: Cold Start
MainActivity_Lifecycle: onCreate
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
--------------------------- PROCESS STARTED (6899) for package com.example.lifecycle ---------------------------
2025-08-13 09:45:33.847  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onCreate
2025-08-13 09:45:33.878  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onStart
2025-08-13 09:45:33.879  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onResume
```

D. Page Navigation Operations

D1. In-App Back Button Exit

User Operation: Press Back button in app's root Activity

```
Lifecycle Sequence: onPause → onStop → onDestroy
App State: App normally exits
Process State: Process terminates
Characteristics: Normal exit flow, saves necessary state
Startup Type: Next startup will be cold start
```

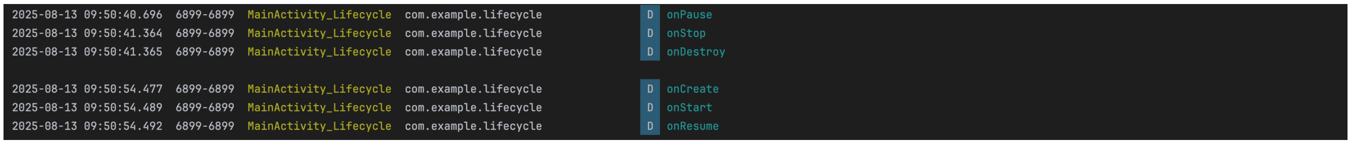
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Press Back] --> C[onPause] --> D[onStop] --> E[onDestroy] --> F[App exit] F --> G[Restart] --> H[onCreate] --> I[onStart] --> J[onResume] --> K[Activity running] style A fill:#90EE90 style F fill:#FFA07A style K fill:#90EE90

Log Output Examples:

```
# Back button exit
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop
MainActivity_Lifecycle: onDestroy

# Restart (Cold start)
SplashActivity_Lifecycle: onCreate - Start Type: Cold Start
MainActivity_Lifecycle: onCreate
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 09:50:40.696  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onPause
2025-08-13 09:50:41.364  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onStop
2025-08-13 09:50:41.365  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onDestroy

2025-08-13 09:50:54.477  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onCreate
2025-08-13 09:50:54.489  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onStart
2025-08-13 09:50:54.492  6899-6899  MainActivity_Lifecycle  com.example.lifecycle              D  onResume
```

D2. In-App Return to Previous Page

User Operation: Press Back button in app's non-root Activity

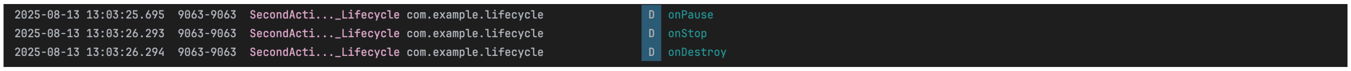
```
Lifecycle Sequence:
- Current Activity: onPause → onStop → onDestroy
- Previous Activity: onRestart → onStart → onResume
App State: Current page destroyed, return to previous page
Process State: Process keeps running
Characteristics: Normal Activity stack pop operation
Startup Type: Not applicable to app startup
```

Lifecycle Flow Diagram:

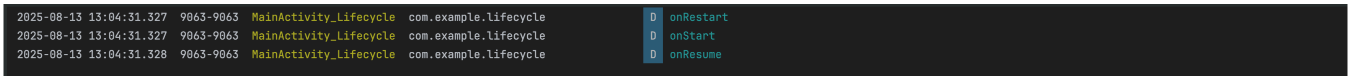
graph LR A[SecondActivity] --> B[Press Back] --> C[SecondActivity onDestroy] --> D[MainActivity onRestart] --> E[onStart] --> F[onResume] style A fill:#90EE90 style F fill:#90EE90

Log Output Examples:

```
# Return to previous page
SecondActivity_Lifecycle: onPause
MainActivity_Lifecycle: onRestart
SecondActivity_Lifecycle: onStop
MainActivity_Lifecycle: onStart
SecondActivity_Lifecycle: onDestroy
MainActivity_Lifecycle: onResume
```



```
2025-08-13 13:03:25.695  9063-9063  SecondActi..._Lifecycle com.example.lifecycle          D  onPause
2025-08-13 13:03:26.293  9063-9063  SecondActi..._Lifecycle com.example.lifecycle          D  onStop
2025-08-13 13:03:26.294  9063-9063  SecondActi..._Lifecycle com.example.lifecycle          D  onDestroy
```



```
2025-08-13 13:04:31.327  9063-9063  MainActivity_Lifecycle  com.example.lifecycle          D  onRestart
2025-08-13 13:04:31.327  9063-9063  MainActivity_Lifecycle  com.example.lifecycle          D  onStart
2025-08-13 13:04:31.328  9063-9063  MainActivity_Lifecycle  com.example.lifecycle          D  onResume
```

D3. Gesture Back Exit App

User Operation: Swipe from screen edge to return, exit app in root Activity

```
Lifecycle Sequence: onPause → onStop → onDestroy
App State: App exits
Process State: Process terminates
Characteristics: Same effect as physical Back button
Startup Type: Next startup will be cold start
```

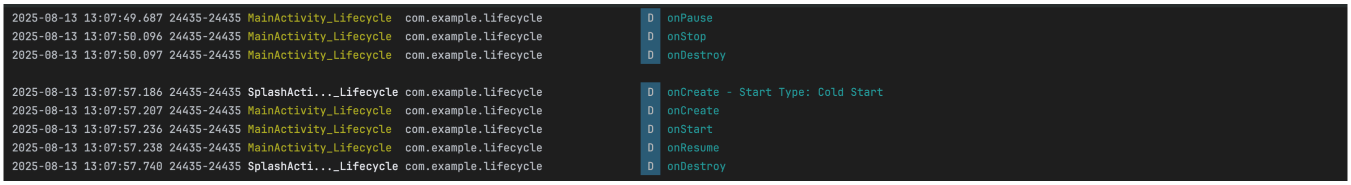
Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Gesture back] --> C[onPause] --> D[onStop] --> E[onDestroy] --> F[App exit] F --> G[Restart] --> H[onCreate] --> I[onStart] --> J[onResume] --> K[Activity running] style A fill:#90EE90 style F fill:#FFA07A style K fill:#90EE90

Log Output Examples:

```
# Gesture back exit
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop
MainActivity_Lifecycle: onDestroy

# Restart (Cold start)
SplashActivity_Lifecycle: onCreate - Start Type: Cold Start
MainActivity_Lifecycle: onCreate
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 13:07:49.687 24435-24435 MainActivity_Lifecycle  com.example.lifecycle              D  onPause
2025-08-13 13:07:50.096 24435-24435 MainActivity_Lifecycle  com.example.lifecycle              D  onStop
2025-08-13 13:07:50.097 24435-24435 MainActivity_Lifecycle  com.example.lifecycle              D  onDestroy

2025-08-13 13:07:57.186 24435-24435 SplashActi..._Lifecycle com.example.lifecycle              D  onCreate - Start Type: Cold Start
2025-08-13 13:07:57.207 24435-24435 MainActivity_Lifecycle  com.example.lifecycle              D  onCreate
2025-08-13 13:07:57.236 24435-24435 MainActivity_Lifecycle  com.example.lifecycle              D  onStart
2025-08-13 13:07:57.238 24435-24435 MainActivity_Lifecycle  com.example.lifecycle              D  onResume
2025-08-13 13:07:57.740 24435-24435 SplashActi..._Lifecycle com.example.lifecycle              D  onDestroy
```

E. System Function Operations

E1. Pull Down Notification Panel

User Operation: Pull down from top to open notification panel

```
Lifecycle Sequence: onPause (app partially obscured)
App State: App paused but still partially visible
Process State: Process continues running
Characteristics: Lightweight pause, quick recovery
Startup Type: Immediate onResume when panel collapsed工
```

App partially obscured

Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Pull notification] --> C[onPause] --> D[Panel open] --> E[Collapse panel] --> F[onResume] --> G[Activity running] style A fill:#90EE90 style D fill:#FFF3E0 style G fill:#90EE90

Log Output Examples:

```
# Pull down notification panel
MainActivity_Lifecycle: onPause

# Collapse notification panel
MainActivity_Lifecycle: onResume
```

E2. Screen Rotation

User Operation: Rotate device to change screen orientation

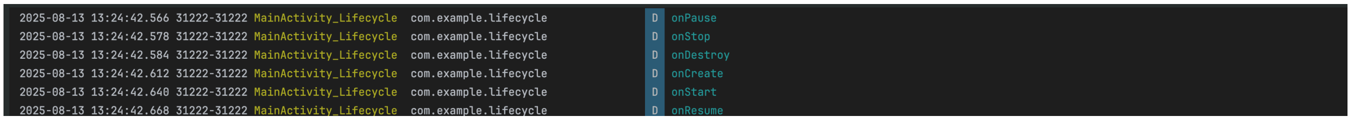
```
Lifecycle Sequence: onSaveInstanceState → onPause → onStop → onDestroy → onCreate → onRes
App State: Activity recreated to adapt to new configuration
Process State: Process remains unchanged
Characteristics: Configuration change, state needs to be saved and restored
Startup Type: Not applicable to app startup, but Activity recreation
```

Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Rotate screen] --> C[Save state] --> D[onDestroy] --> E[onCreate] --> F[Restore state] --> G[onResume] --> H[Activity running] style A fill:#90EE90 style D fill:#FFE0B2 style H fill:#90EE90

Log Output Examples:

```
# Screen rotation
MainActivity_Lifecycle: ========== State Save ==========
MainActivity_Lifecycle: onSaveInstanceState - Save message: 'Updated: 1642134567890'
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: onStop
MainActivity_Lifecycle: onDestroy
MainActivity_Lifecycle: ========== MAIN LIFECYCLE STATE ==========
MainActivity_Lifecycle: onCreate - State Type: State Restore
MainActivity_Lifecycle: ========== State Restore ==========
MainActivity_Lifecycle: onRestoreInstanceState - State restore complete
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 13:24:42.566 31222-31222 MainActivity_Lifecycle  com.example.lifecycle              D  onPause
2025-08-13 13:24:42.578 31222-31222 MainActivity_Lifecycle  com.example.lifecycle              D  onStop
2025-08-13 13:24:42.584 31222-31222 MainActivity_Lifecycle  com.example.lifecycle              D  onDestroy
2025-08-13 13:24:42.612 31222-31222 MainActivity_Lifecycle  com.example.lifecycle              D  onCreate
2025-08-13 13:24:42.640 31222-31222 MainActivity_Lifecycle  com.example.lifecycle              D  onStart
2025-08-13 13:24:42.668 31222-31222 MainActivity_Lifecycle  com.example.lifecycle              D  onResume
```

E3. Enter Split Screen Mode

User Operation: Long press Recent button or gesture to enter split screen

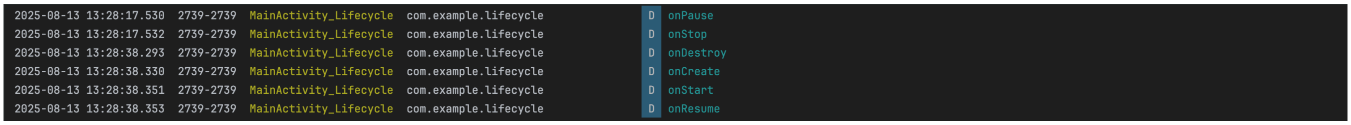
```
Lifecycle Sequence: onPause → Configuration change → onSaveInstanceState → onStop → onDest
App State: Activity recreated to adapt split screen layout
Process State: Process keeps running
Characteristics: Multiple configuration changes, layout adaptation
Startup Type: Not applicable to app startup, but needs layout adaptation
```

Lifecycle Flow Diagram:

graph LR A[Activity running] --> B[Enter split screen] --> C[Config change] --> D[Recreate] --> E[onStart] --> F[onResume] --> G[Split screen ready] style A fill:#90EE90 style D fill:#F3E5F5 style G fill:#90EE90

Log Output Examples:

```
# Enter split screen mode
MainActivity_Lifecycle: onPause
MainActivity_Lifecycle: ========== State Save ==========
MainActivity_Lifecycle: onSaveInstanceState - Save message: 'Welcome'
MainActivity_Lifecycle: onStop
MainActivity_Lifecycle: onDestroy
MainActivity_Lifecycle: onCreate - State Type: State Restore
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
```



```
2025-08-13 13:28:17.530  2739-2739  MainActivity_Lifecycle  com.example.lifecycle          D  onPause
2025-08-13 13:28:17.532  2739-2739  MainActivity_Lifecycle  com.example.lifecycle          D  onStop
2025-08-13 13:28:38.293  2739-2739  MainActivity_Lifecycle  com.example.lifecycle          D  onDestroy
2025-08-13 13:28:38.330  2739-2739  MainActivity_Lifecycle  com.example.lifecycle          D  onCreate
2025-08-13 13:28:38.351  2739-2739  MainActivity_Lifecycle  com.example.lifecycle          D  onStart
2025-08-13 13:28:38.353  2739-2739  MainActivity_Lifecycle  com.example.lifecycle          D  onResume
```

E4. Return from Settings

User Operation: Return to app from notification permission settings page

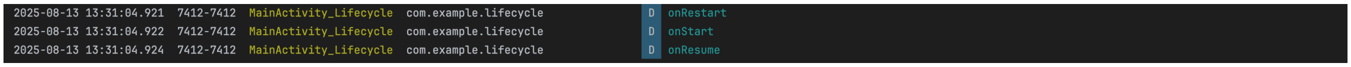
```
Lifecycle Sequence: onRestart → onStart → onResume
App State: Resume from background, may need to check permission state
Process State: Process keeps running
Characteristics: Hot start, but needs to recheck system state
Startup Type: Hot start
```

Lifecycle Flow Diagram:

graph LR A[Settings page] --> B[Return to app] --> C[onRestart] --> D[onStart] --> E[onResume] --> F[Check permissions] --> G[Activity running] style A fill:#E3F2FD style G fill:#90EE90

Log Output Examples:

```
# Return from settings
MainActivity_Lifecycle: onRestart
MainActivity_Lifecycle: onStart
MainActivity_Lifecycle: onResume
MainActivity_Lifecycle: onResume - Permissions granted
```



```
2025-08-13 13:31:04.921  7412-7412  MainActivity_Lifecycle  com.example.lifecycle          D  onRestart
2025-08-13 13:31:04.922  7412-7412  MainActivity_Lifecycle  com.example.lifecycle          D  onStart
2025-08-13 13:31:04.924  7412-7412  MainActivity_Lifecycle  com.example.lifecycle          D  onResume
```

Android Official Lifecycle Diagram with User Operation Mapping

graph TD A[Activity launched] --> B[onCreate] B --> C[onStart] C --> D[onResume] D --> E[Activity running] E --> F[onPause] F --> G{User Operation Type} G -->|Pull notification panel| H[App partially visible] H --> D G -->|Home/Task switch/Open other app| I[onStop] I --> J[Activity not visible] J --> K{System Decision} K -->|User returns to app| L[onRestart] L --> C K -->|Memory pressure/Swipe close/Back exit| M[onDestroy] M --> N[Activity shut down] E -->|Back button exit root Activity| O[Direct onDestroy] O --> M subgraph "User Operation Trigger Points" P1[A1,A2: Home button/gesture to home] --> I P2[B1: Open other apps] --> I P3[B2: Task manager switch] --> L P4[C1,C2: Swipe up/left-right close] --> M P5[D1,D3: Back button exit app] --> O P6[E1: Pull notification panel] --> F P7[E2,E3: Screen rotation/split screen] --> M end style E fill:#90EE90 style J fill:#FFB6C1 style N fill:#FFA07A

User Operations and Lifecycle State Mapping

User Operation Triggered Lifecycle Paths

Operation Category	Specific Operations	Lifecycle Path	Final State	Next Startup Type
A: Return to Home	A1,A2	onPause → onStop	Activity not visible	Hot start
B: App Switching	B1	onPause → onStop	Activity not visible	Hot start
	B2	onRestart → onStart → onResume	Activity running	-
	B3,B4	Depends on current app state	May be running or not visible	Hot/ Cold start
C: App	C1,C2,C3	onPause → onStop →	Activity	Cold

Operation Category	Specific Operations	Lifecycle Path	Final State	Next Startup Type
Termination		onDestroy	destroyed	start
D: Page Navigation	D1,D3	onPause → onStop → onDestroy	Activity destroyed	Cold start
	D2	Current Activity destroyed, previous Activity resumed	Activity running	-
E: System Functions	E1	onPause → onResume	Activity running	-
	E2,E3	Complete recreation flow	Activity running	-
	E4	onRestart → onStart → onResume	Activity running	-

Key State Transition Analysis

graph LR subgraph "From Activity running" A[Activity running] --> B[onPause trigger point] end B --> C{User Operation Decision} C -->|E1:Notification panel| D[onPause only] D --> E[Quick recovery onResume] C -->|A1,A2,B1:Go to background| F[onPause → onStop] F --> G[Enter background state] C -->|D1,D3,C1,C2,C3:Exit/Close| H[onPause → onStop → onDestroy] H --> I[Activity destroyed] C -->|E2,E3:Configuration change| J[Save state → Destroy → Recreate] J --> K[Activity recreation complete] style D fill:#e8f5e8 style F fill:#fff3e0 style H fill:#ffebee style J fill:#f3e5f5

State Persistence Analysis

graph TB A[User Data] --> B{Operation Type} B -->|A: Return to Home| C[Fully Preserved ✓] B -->|B: App Switching| D[Fully Preserved ✓] B -->|C: App Termination| E[Data Lost ✗] B -->|D: Page Navigation| F[Data Lost ✗] B -->|E: System Functions| G[State Save/Restore ⚠] style C fill:#c8e6c9 style D fill:#c8e6c9 style E fill:#ffcdd2 style F fill:#ffcdd2 style G fill:#fff9c4