ANDROID DEVICE AND METADATA ANALYSIS

ET401

By

Prakriti Negi
BT22GCY126

NIIT UNIVERSITY
THE UNIVERSITY OF THE FUTURE

Table of Contents

S. No.	Content
1.	Executive Summary
2..	Introduction
3.	Objectives
4.	Scope & Limitations
5.	Background on ADB
6.	Educational value of ADB
7.	Importance of Device Profiling in Mobile Analysis
8.	Importance of Metadata in Social Apps
9.	Prerequisites & Environment Setup
10.	Ethical Considerations
11.	Planned Methodology
12.	Planned Exercises with Explanation
13.	Expected Outcomes
14.	Deliverables
15.	Risks, Issues & Troubleshooting Plan
16.	Timeline
17.	Success Criteria & Evaluation
18.	Planned Commands Reference
19.	Risks of Misuse and Mitigation Measures
20.	Potential Extensions for Future Work
21.	Conclusion of Work Plan

Executive Summary

This project focuses on using **Android Debug Bridge (ADB)** as a command-line tools to analyze mobile devices and their installed applications. The project will combine two key areas:

(i) Mobile device profiling

(ii) Application metadata analysis of common social apps like WhatsApp, Instagram etc.

The work plan outlines the steps required to set up the environment, connect a device, execute ADB commands and document results. While the project is technical in nature, it emphasizes step-by-step execution so that even without prior deep knowledge of ADB, one can replicate and understand the process. The aim is not just to execute commands but to learn the significance of each piece of metadata obtained and how it contributes to device analysis.

Introduction

The mobile phone has become an essential device in everyday life, containing vast amounts of data. For developers, testers, and analysts, tools like ADB provide ways to interact with a device without relying on its touchscreen interface. This work plan aims to use ADB as a practical diagnostic tool. The project focuses on two tasks: (1) extracting general information about the device, such as model, operating system version, battery health, and storage; and (2) examining the metadata of installed social media applications. Metadata refers to data that provides information about other data—for example, an app's version number, installation date, and permissions. These details are not personal content but technical information that helps understand how apps function on a device.

Objectives

- O1 – **Device Profiling:** Gather general system-level information including model, OS version, battery status, resolution, uptime, and storage.
- O2 - **App Metadata Analysis:** Identify popular social apps installed and extract their package metadata including permissions, version information, and timestamps.
- O3 – **Visual Proof of Execution:** Capture screenshots and optional screen recordings using ADB to demonstrate practical usage.
- O4 – **Evidence Management:** Save all results in a structured folder for easy reporting and demonstration.
- O5 - **Skill Development:** Gain hands-on knowledge of ADB and command-line tools, and learn how they can be used for basic device analysis.

Scope & Limitations

Planned Scope:

- Use ADB for gathering information available through system-level commands.
- Focus on diagnostic and educational aspects only.
- Work only on personal or test devices with explicit permission.

Planned Limitations:

- No attempt will be made to access personal content like chats, photos, or media files.
- No rooting or advanced forensic techniques will be performed.
- Some metadata may not be visible depending on Android security restrictions.

Background on ADB

ADB, or Android Debug Bridge, is part of the Android SDK Platform-Tools. It enables developers and testers to communicate with Android devices from a computer via a command-line interface. ADB can be used to install/uninstall apps, transfer files, access system logs, and execute commands directly on the device. For this project, ADB will be used primarily in **diagnostic mode** to collect device information and metadata. Since ADB is widely used in testing, debugging, and mobile forensics, learning it provides valuable practical skills.

Key features of ADB include:

- Device connection management.
- Shell access to run Linux-based commands.
- App installation, removal, and metadata queries.
- File transfer between PC and Android.
- Capture of screenshots and recordings.

By focusing on these capabilities, the project will highlight ADB’s usefulness without diving into unnecessary complexity.

Educational Value of ADB

ADB is not just useful for developers—it is also an essential diagnostic tool for students, researchers, and professionals. Using ADB helps learners to:

- Interact with the Android system beyond the touchscreen.
- Understand how operating systems manage apps, resources, and logs.
- Practice using command-line tools, which are common in industry.
- Appreciate the importance of structured evidence collection.

This project, therefore, is not just a technical exercise but also a training activity in discipline, evidence handling, and systematic analysis.

Importance of Device Profiling in Mobile Analysis

Device profiling is one of the first steps in any mobile analysis process. It involves gathering baseline information about the device so that later observations can be compared against it. Profiling answers questions such as:

- What model of phone is being tested?
- What version of Android does it run?
- How much battery power is available?
- What is the resolution and density of the screen?
- How long has the device been running since its last reboot?

All of these questions are essential to set the context for app analysis. For example, if a certain app behaves differently on two phones, profiling might show that the difference is caused by different Android versions or screen densities. Including this step ensures that analysis is not done in isolation but with awareness of the device environment.

Importance of Metadata Analysis in Social Apps

Metadata is “data about data.” In the context of mobile apps, metadata includes the version of the app, the date when it was first installed, the most recent update date, and the permissions it requests. While metadata is not sensitive like personal messages or photos, it is still highly informative. For example:

- Version numbers can indicate whether an app is outdated or vulnerable.
- Permissions requested by an app can show whether the app is accessing only necessary functions (e.g., camera for Instagram) or requesting unrelated access (e.g., microphone for an app that should not need it).
- Installation and update times help in understanding user behavior and app maintenance cycles.

In practice, when we run the command *adb shell dumpsys package com.whatsapp*, the output includes lines for *versionName*, *versionCode*, *firstInstallTime* and *lastUpdateTime*. By noting these values, one can comment on how actively the app is maintained and what level of permissions it has.

Prerequisites & Environment Setup

Planned Tools

- Android SDK Platform-Tools (ADB) installed on a PC.
- A computer with Windows/Linux/macOS.
- A USB data cable for connecting the device.

Device Preparation

- Enable Developer Options (via tapping Build Number multiple times).
- Enable USB Debugging from Developer Options.
- Connect the phone to the computer and allow RSA key authorization.

Connectivity Verification

- Use *adb devices* command to ensure that the device is recognized.
- Plan to troubleshoot common issues like unauthorized or offline states.

Ethical Considerations

- The project will be conducted only on authorized devices.
- No personal user data will be accessed or stored.
- All findings will be restricted to technical metadata.
- The project will clearly separate educational demonstration from forensic-level investigation.

Planned Methodology

Phase A – Environment Setup

1. Install ADB tools.
2. Verify installation using *adb version.*
3. Connect the device and run *adb devices* to ensure proper setup.

Phase B – Device Profiling

Commands to be executed:

1. Model: *adb shell getprop ro.product.model*
2. Android Version: *adb shell getprop ro.build.version.release*
3. Battery: *adb shell dumpsys battery | grep level*
4. Resolution: *adb shell wm size*
5. Density (DPI): *adb shell wm density*
6. Uptime: *adb shell uptime*
7. Storage: *adb shell df -h /*

Each command will be documented with its purpose and expected type of output.

Phase C – Social App Metadata Analysis

Steps:

1. List all installed packages using *adb shell pm list packages.*
2. Identify popular social apps by filtering names.
3. Run *adb shell dumpsys package <package>* for each app.
4. Extract version, permissions, installation time, and update time.

Phase D – Visual Capture

Steps:

1. Capture screenshot using *adb shell screencap* and transfer to PC.
2. Optionally, record 10 seconds of screen activity using *adb shell screenrecord*.
3. Store all visuals with proper naming conventions.

Phase E – Evidence Organization

1. Create folders for device info, app metadata and screenshots.
2. Save all text outputs and media files in designated folders.
3. Maintain clear filenames and logs for each activity.

Planned Exercises with Explanations

Exercise 1: Connect Device
Purpose: To verify that ADB can communicate with the device.
Expected Outcome: Device listed under *adb devices*.

Exercise 2: Device Profiling
Purpose: To extract hardware and software specifications of the device.
Explanation: These details help identify the device and provide baseline information.

Exercise 3: Social App Metadata
Purpose: To examine technical details of one or more installed social apps.
Explanation: Metadata such as permissions and version are important for understanding app behavior.

Exercise 4: Screenshot Capture
Purpose: To visually document the device state at a specific time.
Explanation: Screenshots serve as proof of execution and add credibility to the project.

Exercise 5: Optional Screen Recording
Purpose: To practice capturing dynamic activity on the device.
Explanation: Demonstrates advanced but accessible ADB capabilities.

Expected Outcomes

- Successfully connected device recognized by ADB.
- A structured report of device profiling information.
- Metadata records of selected social media apps.
- Screenshots captured and stored in the project folder.
- Optional video recording as an additional deliverable.

Deliverables

- Evidence folder with organized outputs (device info, metadata, screenshots).
- Final report containing methodology, commands, and explanations.
- Demo plan for a short live demonstration during project evaluation.

Risks, Issues & Troubleshooting Plan

- **Connection Failures:** Check drivers, cable, and USB debugging settings.
- **ADB Unauthorized:** Accept RSA prompt on device.
- **Permission Restrictions:** Document limitations clearly in final report.
- **Time Constraints:** Focus on core objectives first (device profiling and one app metadata).

Timeline (25 Aug – 30 Sept 2025)

Period	Planned Tasks
25-31 Aug	Install ADB, prepare environment
1-7 Aug	Test connection and basic device profiling commands
8-11 Sept	Work on app metadata commands
12-14 Sept	Practice screenshot and screen recording
15-20 Sept	Organize sample outputs, document methodology
21-25 Sept	Draft report with explanations and placeholders for output
26-27 Sept	Finalize documentation
29-30 Sept	Submit report

Success Criteria & Evaluation

- Device connection is demonstrated successfully.
- At least one app's metadata is extracted and documented.
- A screenshot is captured using ADB.
- Evidence is well organized and presented.
- Report is detailed, covering 10+ pages with thorough explanations.

Planned Commands Reference

- adb devices
- adb shell getprop ro.product.model
- adb shell getprop ro.build.version.release
- adb shell dumpsys battery | grep level
- adb shell wm size
- adb shell wm density
- adb shell uptime
- adb shell df -h /
- adb shell pm list packages
- adb shell dumpsys package <package>
- adb shell screencap -p /sdcard/screen.png && adb pull/sdcard/screen.png
- adb shell screenrecord /sdcard/demo.mp4 –time-limit=10

Risks of Misuse and Mitigation Measures

While ADB is a powerful tool, it can be misused if not handled responsibly. Some of the risks include:

- Accidentally uninstalling or modifying apps if destructive commands are run.
- Accessing sensitive information if used improperly on someone else's device.
- Damaging device settings by running commands without understanding them.

To mitigate these risks, the project work plan includes:

- Limiting commands only to safe, diagnostic ones.
- Working on personal or authorized devices only.
- Avoiding commands that modify apps or system files.
- Double-checking every command before running it.

By establishing these boundaries, the project stays safe and ethical while still being educational.

Potential Extensions for Future Work

Although this project is focused on device profiling and metadata collection, it opens the door for future extensions such as:

- **Automation with Scripts:** Wrapping ADB commands into shell or batch scripts to automate repetitive tasks.
- **Comparison Across Devices:** Running the same commands on multiple phones to compare results.
- **Historical Tracking:** Running the same profiling at different times to observe changes (e.g., battery health trends).
- **Integration with Forensic Tools:** Using ADB as a starting point before moving into more advanced mobile forensic analysis (with proper authorization).

Including these possibilities in the work plan shows awareness of the broader applications of ADB beyond this project.

Conclusion of Work Plan

This work plan provides a structured roadmap for conducting a project on Mobile and App Metadata Analysis using ADB. By dividing the work into setup, profiling, metadata analysis, visual capture, and documentation phases, the plan ensures clarity and direction. The expected outcomes are not only technical outputs (like device info and screenshots) but also personal learning outcomes such as improved command-line skills, better understanding of metadata, and awareness of ethical practices. The project is modest in scale but rich in educational value, making it achievable within the timeline and requirements while still meeting academic standards for a work plan report.