Performance Analysis of Efficient Power Management Controls in Android Device

Chirag Shah[1], Sumit Chaudhary[2], Priya Agrawal[3]

Student, Computer Engineering, IIST, Cadila Group, Ahmedabad. India[1]

Head of Department, Computer Engineering, IIST, Cadila Group, Ahmedabad. India[2]

Lecturer, Computer Engineering, IIST, Cadila Group, Ahmedabad. India[3]

Abstract: - Now a days too much research work is carried out currently in Android, but one of the biggest issue that faced by any customer as well as developer is that, Android device consumes too much power that leads to battery and heating problems which degrades the performance. This paper depicts the significance of various ways how one can reduce the power consumption of android device in terms of Linux power management, Battery Enhancement, Wake Lock management etc. and provide the efficient result in growth of performance of device.

Keyword:-Android power consumption, Android performance management, Battery Enhancement, Power Saving Apps, Power efficiency

I. INTRODUCTION

The rendition history of the Android versatile working O.S. started with the arrival of the basic or trial Android version alpha in November 5, 2007[3]. The main business variant, Android 1.0, was comes in September 2008. Android is constantly created by Google and the Open Handset Alliance, and it has seen various updates to its base working framework since the underlying discharge. Adaptations 1.0 and 1.1 were not discharged under particular code names, but rather since 2009's Android 1.5 Cupcake, Android renditions have had candy store themed code names. Each is in sequential order arrange, with the latest significant variant being Android 7.0 Nougat, discharged in August 2016. Control administration is the most basic thing in any working framework because of the power request of top of the line cell phones [7]. Control administration is a system to ration control in the cost of idleness/execution. Android Power administration based on top of Linux Power Management. Linux Power administration is sufficiently bad for a Mobile gadget. [1] So android keeps the low level power administration execution on Linux and assembled Android Power administration stack on top of Linux PM.As Android platform has become more and more popular, large amount of Android applications have been developed. When developers design and implement Android applications, power consumption management is an important factor to consider since it improves the usability of the applications. Research mainly includes the three different avenue for the better battery life of Android devices, power saving Apps survey and how one can improve performance management of device. Research mainly includes the three different avenue for the better battery life of Android devices, power saving Apps survey and how one can improve performance management of device. The first section of paper include the Power Management Controls method for android. It consist following ,Power Management in Linux, Battery Enhancement to improve battery life ,Power driver, Wake lock management .Rest of section of paper includes the Survey on Power saving Apps, performance management survey.

II. LITERATURE RIEVIEW

Today, Android is not only limited for the mobile gadgets. It has also expanded for Smart TV, Laptop/Computer OS. Android is a most quick adaptable and grow able versatile working Operating System created by Google, in view of the Linux bit and planned essentially for touchscreen cell phones, for example, cell phones and tablets. Android's UI is primarily in light of direct control, utilizing touch motions that freely relate to certifiable activities, for example, swiping, tapping and squeezing, and controlling display objects, alongside a virtual console window for content information [5]. Previous work consist the four avenue for controlling the power but in this research I include the few new research based on customer as well as developer's interaction with Android. And also surveying the various ways to improve performance of any android device and Apps that's very useful to improve the battery life as well as less power consumption.

III. MECHANISMS OF POWER MANAGEMENT AND ARCHITECTURES

A) Ways of Battery Performance Enhancements:

Battery life is a perennial user concern. To extend battery life, Android continually adds new features and optimizations to help the platform optimize the off-charger behaviour of applications and devices [1]. Power management aims to improve battery life of equipment by minimizing power consumption while guaranteeing expected system performance .Active power consumption occurs while some processing is on-going Dynamic power consumption (transistor switching) + Leakage consumption.[3]Static (also Standby or Idle) power consumption occurs when limited or no processing is on-going and the system is waiting for a wakeup event.
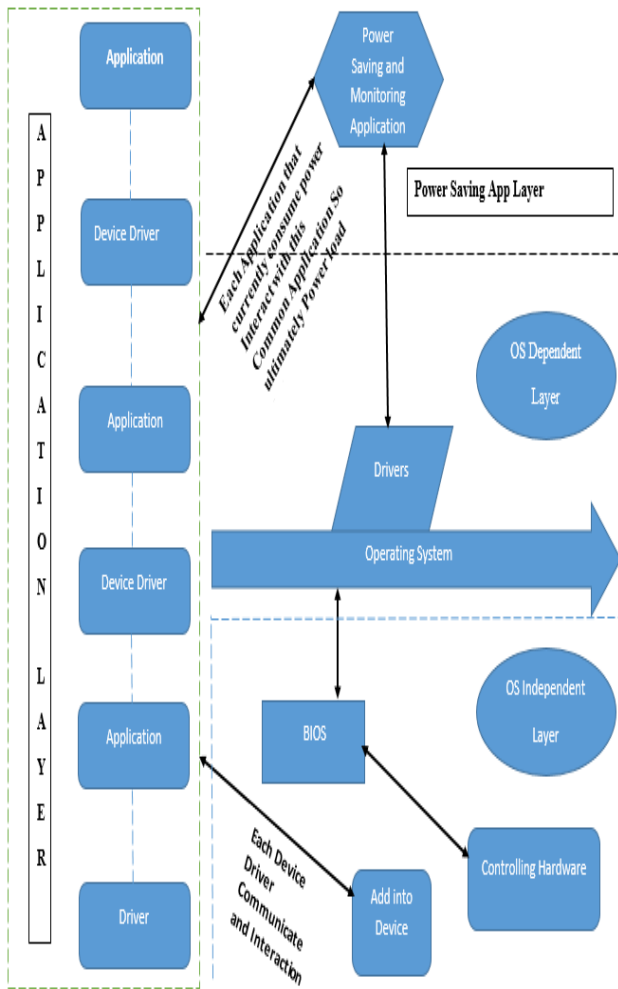
Fig. 1. Power Management Architecture

Fig. 2. Standby Mode Workstation

Very limited dynamic power consumption + Leakage consumption
➢   Managed by Dynamic Voltage & Frequency Scaling (DVFS)
➢   Adaptive Voltage Scaling (AVS)
➢   Dynamic Power Switching (DPS)

There are several ways for enhancements of battery used in android device.

I. APP Standby Mode:-
The stage distinguishes inert applications and spots them in App Standby until the client starts effectively captivating with the application [4].For example considered the standby function which can be invoked or replaced at the time when it is needed. Same as APP Standby means the user can remove or close the current application that being proceed or replaced it with some other Application for use.

Active applications are unaffected by App Standby. An application is active when it has:
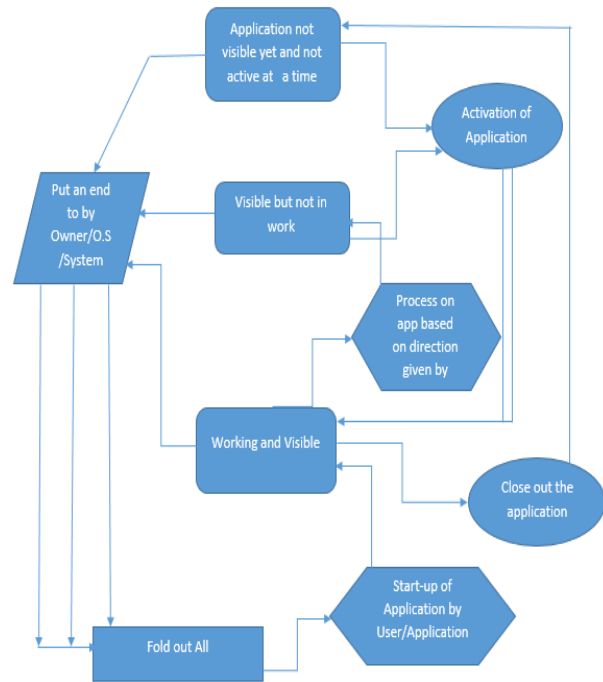
➢   The stage distinguishes inert applications and spots them in App Standby until the client starts effectively captivating with the application.
➢   A warning saw by the client, for example, in the bolt screen or notice plate.
➢   Explicitly been launched by the user.

II. DOZE Mode:-
Doze extends battery life by deferring application background CPU and network activity when a device is unused for long periods. Android 7.0 extends Doze to trigger a lighter set of optimizations every time a user turns off the device screen, even when the user continues to move around, enabling longer lasting battery life. Your smartphone does so much that it feels impossible to get through the day on a full charge. It doesn't have to be that way, though. Android's Doze feature, introduced in Marshmallow and refined in Nougat, keeps battery drain low when you're not using your phone [6]. Here's how it works, and how to make the most of it. In Android, applications can utilize what's known as a "wake lock" to keep your telephone from going into a power-sharing profound rest mode. This profound rest mode generally kicks in when your telephone's screen is off, yet that can hinder how some applications function. For instance, in case you're utilizing a wellness tracker, you don't need your telephone killing GPS or your accelerometer in light of the fact that your telephone is in your pocket with the screen killed. This is a decent idea. Applications keep your telephone conscious and working when they have to, and given it a chance to rest when they don't. This is an issue, however, when each designer thinks their application is sufficiently essential to keep your telephone on constantly.

That is the reason applications like Facebook execute your battery, notwithstanding when you're not utilizing them. System services (such as telephony) may be preloaded and exempted from Doze by default [4]. Users can also exempt specific applications from Doze in the Settings menu. By default, Doze is disabled in the Android Open Source Project (AOSP). For details on enabling Doze, see Integrating Doze.

III. Exemptions:-

You can exempt applications from being subject to Doze or App Standby. Exemptions may be needed in the following use cases:

➢ OEM using non-GCM Cloud Messaging platform
➢ Carrier using non-GCM Cloud Messaging platform
➢ Third-party application using non-GCM Cloud Messaging platform

Developer must be needed provide attention on Do not exempt apps to avoid testing and optimizing. Unnecessary exemptions undermine the benefits of Doze and App Standby and can compromise the user experience, so we strongly suggest minimizing such exemptions as they allow applications to defeat beneficial controls the platform has over power use [3]. If users become unhappy about the power consumption of these apps, it can lead to frustration, bad experiences (and negative user reviews for the app), and customer support questions. For these reasons, we strongly recommend that you do not exempt third-party applications and instead exempt only cloud messaging services or apps with similar functions.

Apps exempted by default are listed in a single view within the Settings > Battery menu. This list is used for exempting the app from both Doze and App Standby modes. To provide transparency to the user, the Settings menu MUST show all exempted applications

B) Linux Power Management Configuration and Working:-

There are two most powerful power management standards

A) APM (Advanced Power Management)
B) ACPI (Advanced Configuration and Power Interface)

I. APM:-

WORKING:-Control lives in BIOS .Uses movement timeouts to decide when to shut down a gadget. BIOS once in a while utilized as a part of inserted frameworks. Makes control administration choices without illuminating OS or individual applications. No knowledge of add-in cards or new devices. Uses layered approach to manage devices. APM-aware applications (including device drivers) talk to an OS-specific APM driver [5]. The driver communicates to the APM-aware BIOS, which controls the hardware. Communication occurs in both directions; power management events are sent from the BIOS to the APM driver, and the APM driver sends information and requests to the BIOS via function calls .In this way the

APM driver is an intermediary between the BIOS and the operating system [4] .power management happens in two ways; through function calls from the APM driver to the BIOS requesting power state changes, and automatically based on device activity

II. ACPI

Propelled Configuration and Power Interface (ACPI), then again, is proposed to be the successor to APM. It replaces APM in capacities past the abilities of APM and is perfect with new equipment. It is best in class and thorough contrasted with APM [2]. Control divided between BIOS and OS Decisions managed through the OS .Enables sophisticated power policies for general-purpose computers with standard usage Patterns and hardware No knowledge of device-specific scenarios (e.g. need to provide predictable response times or to respond to critical events over extended period)

WORKING: - ACPI specification defines the following four Global 'Gx' states and six Sleep 'Sx' states for an ACPI-compliant computer-system:

• G0 (S0) :-
➢ 'Away mode' is a subset of S0, where monitor is off but background tasks are running.

• G1, Sleeping, subdivides into the four states S1 through S4:-
➢ S1: All processor caches are flushed, and the CPU(s) stop executing instructions. Power to the CPU(s) and RAM is maintained; devices that do not indicate they must remain on may be powered down
➢ S2: CPU powered off. Dirty cache is flushed to RAM
➢ S3 (mem): Commonly referred to as Standby, Sleep, or Suspend to RAM. RAM remains powered
➢ S4: Hibernation/Suspend-to-Disk - All content of main memory is saved to non-volatile memory such as a hard drive, and is powered down.

• G2 (S5), Soft Off
• G3, Mechanical Off
• The computer's power has been totally removed via a mechanical switch

• Legacy State:-The state on an operating system which does not support ACPI. In this state, the hardware and power are not managed via ACPI, effectively disabling ACPI.
• Power mode interface is on sysfs
• /sys/power/state
• sysfs is a virtual file system provided by Linux. sysfs exports information about devices and drivers from the kernel device model to user space, and is also used for configuration
• Changing state done by
➢ # echo mem > /sys/power/state
➢ # echo disk > /sys/power/state

➤ # echo standby > /sys/power/state

C) Power Driver:-
Despite the fact that Android acquires the power administration of Linux, the previous added its own particular power driver to the portion 2.6.33[city]. The driver controls the peripherals which incorporate screen show and backdrop illumination, console backdrop illumination and catch backdrop illumination. This driver [5] is added keeping in mind Android devices have limited battery life and the power saving features are different than personal computers.

D) Wake Lock Configuration and Management:-
By default, Android tries to put the system into a sleep or better a suspend mode as soon as possible .Applications running in the Dalvik VM can prevent the system from entering a sleep or suspend state, i.e. applications can assure that the screen stays on or the CPU stays awake to react quickly to interrupts .A dedicated PM API is written in Applications Framework layer. Android apps are required to request CPU resources with wake locks through the application framework and native Linux libraries. The architecture is depicted in Figure 1. If there is no wake lock active, CPU is shut down [6]. Wake locks are used by applications and services to request CPU resources. A locked wake lock, depending on its type, prevents the system from entering suspend or other low-power states. There are two settings for a wake lock. WAKE_LOCK_SUSPEND prevents a full system suspend while WAKE_LOCK_IDLE is a low-power state, which often cause large interrupt latencies or that disable a set of interrupts, will not be entered from idle until the wake locks are released.
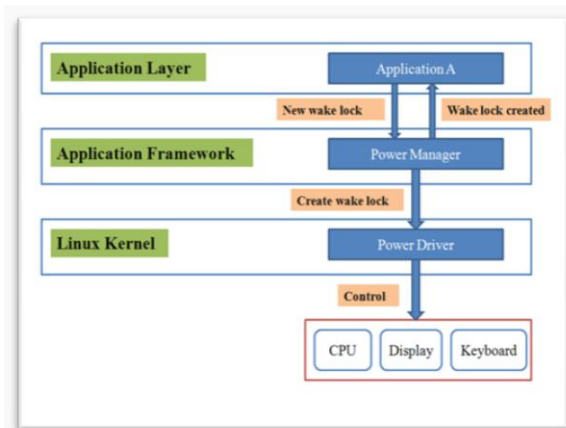

Fig.3. [5] Android Wake lock power management Architecture

In Figure 1, when Application A is launched, it needs to use CPU resources. Thus it sends a request to Power Manager API in applications framework which in turn transfers (using Java Native Interface) the request to the power driver present in the Linux kernel. Power Manager also reports back to the Application A that the wake lock

is created. And depending on the wake lock, resources are consumed [5]. Table 1 lists different wake lock settings. For application which does not interact with the user and run in the background, PARTIAL_WAKE_LOCK is used.

TABLE I
Wake Lock Flag Configuration and Effects

Wake Lock Type	CPU Status	Screen Status	Keyboard Status
PATIAL_ WAKE_LOCK	On	Off	Off
SCREEN_DIM_ WAKE_LOCK	On	Dim	Off
SCREEN_BRIGHT_ WAKE_LOCK	On	Bright	Off
FULL_ WAKE_LOCK	On	Bright	Bright

IV. POWER SAVING APPS SURVEY

With a specific end goal to draw out the battery life of Android controlled gadgets, designers have composed numerous applications accessible in the Google Play Apps store. A few such applications are examined amid the overview to comprehend the power sparing methodologies, how they increment control effectiveness, their working standards and confinements that clear path for changes. It is watched that these power sparing applications have two unmistakable methodologies for controlling force utilization

These approaches are portrayed in Table II. The subsequent sections discuss power efficiency increase, operation and limitations of Set CPU for Root Users [13], CPU tuner (Rooted phones) [12], Juice Defender [7].

These apps are chosen based on their popularity, high user rating and positive user feedback

TABLE II
Power Consumption Approaches

Primary approach	Secondary Approach	Example of apps
CPU frequency scaling	Controlling smartphone features	Set CPU, CPU tuner
Controlling smartphone features	CPU frequency scaling	Juice Defender

a) Power consumption and efficiency Growth:-
Several hardware components like GPS, Wi-Fi of smartphones and tablets consume very high power [7]. Hence power can be spared by turning them off when not being utilized. There are some different components like auto sync, notice recurrence which utilize the availability and other equipment. Bringing down such warning frequencies (for the most part Facebook, Gmail) will

decrease the use of cell phone segments and increment control proficiency. Taking after are the elements which are controlled by these power sparing applications to expand battery life. List that's not exhaustive,

- Flip control on Wi-Fi, Bluetooth, GPS, auto adjust, off-line mode, auto screen bolt, USB mass stockpiling, screen-dependably on, light, 2G, 3G, 4G/WiMAX (if present) and portable information (APN).
- Change brilliance level of show.
- Volume and vibration control.
- Alter screen timeout esteem.
- Scheduling – night, end of the week, crest.
- Setting Wi-Fi timeout.
- Setting dim home screen backdrop for OLED show.

b) Application Solution and Operations:-
It is vital to comprehend the working rule of applications To explore their impediments. We introduce the working outline of Set CPU and CPU tuner in Figure 2.
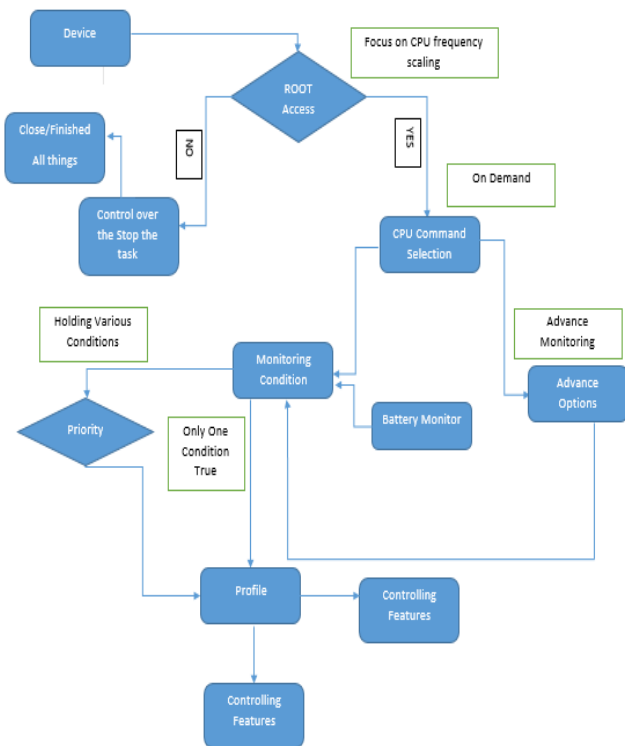

Fig. 4. [Operation of Set CPU and CPU tuner]

Once the app is installed in a rooted phone and root permission is granted, there are sliders that allow controlling the CPU frequency manually. At that point CPU senator must be chosen. It controls how the CPU recurrence ought to be scaled between the most extreme and least set frequencies [10]. A large portion of the bits (thusly cell phones) have "on demand" and "execution". At the point when the CPU stack achieves an edge, on demand scales up the recurrence quickly and downsizes

the recurrence when the heap is lesser. Other accessible CPU governors are recorded in [10]. Some of them have propelled condition checking highlights. The profiles design the application to set the CPU recurrence under specific conditions. There is a "condition screen" which constantly screens the conditions set in profiles. In the event that such a condition is valid, the individual profile is activated. For instance, the profile "Battery <" is set when battery level falls beneath a given edge. The "Time" profile is enacted for a specific length of time. Each profile has a need. On the off chance that states of a few profiles are valid, then the need of the profiles is checked. The profile with most elevated need is enacted. Every profile is additionally ready to give the framework data i.e. the battery level, memory status and so on.

Juice Defender on the other hand focuses on controlling the smartphone features, like disabling connectivity and reducing brightness to reduce power consumption. There are several versions (both free and paid) of Juice Defender [3] available in Google play store. One version of the app requires root permission to scale CPU frequency and toggle 4G/WiMAX. But that is not the prime concern in this power saving app. The operational diagram is given in Figure 3.
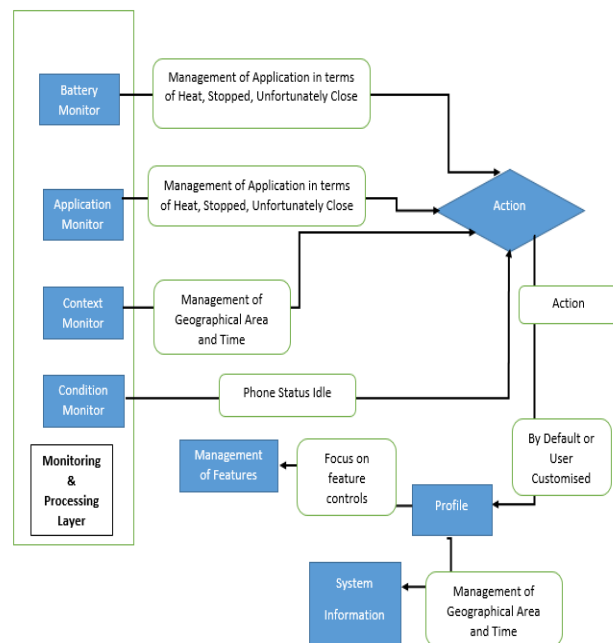

Fig. 5 [Juice Defender Working]

V. GENERAL PERFOMANCE MANAGEMENT SURVEY

This survey is purely based on the user's interaction with mobile device and result are achieved as per as user's requirement, Analysis and other general issues.

1) Device Information:-
It is critical that you find out about the abilities and disadvantages of your telephone. Try not to overburden your telephone with asset hungry applications which

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 3, March 2017

would some way or another debase your telephone's execution to your detriment.

2) Uninstall the Apps that never used:-

Each application you introduce in your telephone consume some storage space and runs some foundation forms. The more storage room possessed or the more foundation procedures running on your telephone, the slower your telephone's execution. Try not to introduce applications you scarcely utilize. Keep just the applications that you require and in the event that you have an application that isn't generally valuable, you can uninstall or incapacitate it.

3) Stop Synchronization:-

Matchup is a decent component which synchronizes your information with Google servers. Keeping sync on, you can get warnings at whatever point new mail comes in or when you get new notices or updates for applications. To do this, it does a revive at each present interim (e.g. it checks your mail at regular intervals for new approaching mail), and this will eat into your android's execution for different exercises. You can keep Sync off for unrequired administrations or just have it on when you have to exchange or transfer materials to Google servers.

4) Update your Apps:-

You ought to consistently refresh your telephone's introduced applications from Google Play. Attempt to get refreshes your applications when they are accessible in Google Play. Engineers settle bugs and include highlights in new discharged forms of applications. Refreshed applications perform better and speedier, and will more improbable crash your telephone.

5) Use High speed memory card:-

The memory card is the storage room of your telephone. Telephones with low interior memory can get a lift from high-limit memory cards for most extreme storage room. Be that as it may, it includes limit as well as speed to the blend. You can get between 2GB to 32GB worth of storage room to bolster rapid read/compose operations. Continuously go for memory cards of Class 6 or Class 10 for your Android telephone in the event that you look for execution for your telephone.

6) Update Your Android Regularly:-

If you haven't refreshed your Android telephone to the most recent firmware, you ought to. Google conveys awesome enhancements to each new arrival of the Android working framework, including refreshes that guarantee steadiness, higher execution speed and availability alongside other easy to understand new elements.

7) Keep less Widgets:-

Gadgets are not applications, which just run once you enact them. No, gadgets are continually running out of sight; they are dynamic constantly. While some Android clients utilize it to monitor the climate, and their imperative dates, others utilize gadgets like 'Developed controls' for snappy access for basic arrangements, for example, exchanging Wi-Fi, Bluetooth or their GPS on or off. They do have their utilizations yet remember that having an excessive number of them running on your home screen will bring about your gadget's execution to back off.

8) Disable App that no longer useful:-

'Execution partner' was acquainted in Android ICS with help you impair applications that you needn't bother with. Handicapping applications is not quite the same as uninstalling applications uninstallation expel application documents from the telephone yet no record is evacuated/erased when you impair it. These debilitated applications are not appeared on the home screen or application plate yet you can re-empower them for utilize at whatever point you require them.

9) Avoid Live theme or Wallpaper:-

You should love Android's live backdrop highlight. Telephones (from various producers) accompany diverse live backdrops however you ought to realize that live backdrops that are gravely made will require more CPU cycles to run, consequently depleting the battery quicker. Every time you actuate the home screen, you run your applications as well as your live
Backdrop.

10) Root Your Device:-

As Android is based on the Linux bit, similar to whatever other OS, it likewise can be established. Establishing your Android gives you manager level access on your telephone, which implies that you can uninstall the applications that accompanied your telephone yet may not be what you need to utilize. You can then glimmer it with custom ROMs that might be a superior fit to your employment prerequisites or identity, or to improve the execution and battery life of your telephone. You can even refresh your telephone to the most recent OS or updates that could some way or another be unsupported for your telephone show. User don't want to root in, there are various elements against establishing your Android gadget; for a certain something, and most makers regard your telephone guarantee voided if your telephone has been established. You likewise need to comprehend what you are doing to abstain from messing everything up, destroyed. This is called "bricking" a.k.a. transforming your telephone into a futile block. So think it through before you utilize this technique to upgrade the execution of your Android.

11) Turn off Animation or any type of service that consume large amount of memory:-

What liveliness? All things considered, on the off chance that you haven't seen, your telephone screen really grows dim and in as you change from one undertaking to the next. To see this in real life Go to Settings > Developer Options > Window Animation Scale > Animation scale 10x. At that point pick Window Animation Scale again to see the blur basically in 'slow mo.' when you pick Animation off, the screen just snaps to position without activity. You can likewise kill Transition Animation Scale similarly. In the event that you are a substantial client, you'd likely notice that your Android performs quicker with all movements off.

12) Use task manager for close the Apps process & Services:-

Download any free Task Killer applications or Task Managers from Google Play. A portion of the great ones are Advanced Task Killer and Super Task Killer Free. Utilize an undertaking executioner to slaughter foundation forms and different applications that you don't need or you don't utilize. Set alternatives to consequently kill pointless applications (in view of your determinations) after each set timeframe. A portion of the cases incorporate Update Manager (refreshes don't come each moment/hour) or any producer introduced application which you don't require.

VI. CONCLUSION

The paper summarize we lead an experimental review on power administration required by Android applications and gadgets. Paper quickly abridge, the need of productive battery administration for Android cell phones and tablets. The new Android control administration design quickly talked about. Distinctive research bearings of cell phone control utilization are discussed. Using power administration engineering talked about in research paper alongside various power sparing application prompts to power loads distribution the outcome in effective development in battery level and less battery consumption. Then four research roads are exhibited to drag out the battery life of Android gadgets. The customer server based approach is roused from in spite of the fact that the disconnected preparing in the remote server could be stretched out all things considered. Comparable utilization examples could be bunched to produce control sparing profiles that considers genuine use design. However, this approach raises protection and security concerns. Another thought is to utilize a learning motor to produce the use design and create control sparing profiles. This approach will limit the protection worries as all the preparing is done inside the App. The review on power sparing application portrays The engineering of Set CPU and Set tuner are given brief data about the execution development because of force sparing application lastly paper talked about survey on various ways utilizing which client oversee android gadget and its application and result in better execution and effective battery level improvements.

REFERENCES

[1] "Power Management Techniques in Smartphones Operating Systems", Ahmed Abdelmotalib, Zhibo Wu, 1694-0814.
[2] "Battery Power Saving Profile with Learning Engine in Android Phones", Rimpy Bala, Anu Garg,   Volume69– No.13, May 2013.
[3] https://en.wikipedia.org/wiki/Android.
[4] https://source.android.com/devices/tech.
[5] "Android Power Management: Current and Future   Trends", Soumya Kanti Datta, Christian Bonnet, Navid Nikaein, 2012 The First IEEE Workshop on Enabling Technologies for Smartphone and Internet of Things (ETSIoT).
[6] https://www.e-consystems.com/blog/linux-android/?p=802.
[7] http://www.juicedefender.com/
[8] "Integration of Smart Phone and IOT for development of Smart Public Transportation System", Shiv. H. Sutar, RohanKoul, Rajani Suryavanshi, 2016 International Conference on Internet of Things and Applications (IOTA),Maharashtra Institute of Technology, Pune, India 22 Jan - 24 Jan, 2016.
[9] "How Android App Developers Manage Power Consumption?" An Empirical Study by Mining Power Management Commit.
[10] http://www.setcpu.com/documentation.html
[11] "Comparing Power Management Strategies of Android and TinyOS", Siquan Hu, Yanchao Yu, Lun Xie.
[12] https://play.google.com/store/apps/details?id=ch.amana.android.cputuner&hl=en
[13] https://play.google.com/store/apps/details?id=com.mhuang.overclocking&hl=en
[14] J.M. Kang, S. Seo and J. Hong. "Usage pattern analysis of smartphones." In 13th Asia-Pacific Network Operations and Management Symposium, 2011, pp. 1-8.

BIOGRAPHIES

Mr. Chirag Shah currently pursuing Bachelor of Computer Engineering from Indrashil Institute of Science & Technology, Cadila Group, Rajpur, Ahmedabad (Gujarat).His area of interest is Web Development and Mobile Development so he is working on Escort Counseling which is web based application developed in Java Technology.

Sumit Chaudhary is working as Head of Department in CSE at Indrashil Institute of Science & Technology, Cadila Group, Rajpur, Ahmedabad (Gujarat). He is pursuing Ph.D. from Uttaranchal University, Dehradun (Uttarakhand). He worked with various institutes like Uttaranchal Institute of Technology (UIT), Dehradun, Shri Ram Group of colleges, Muzaffarnagar (U.P.), IIMT Institute of Engineering & Technology, Meerut (U.P.), INDIA including all that he has more than 7 year experience in teaching. He obtained his M-Tech (Computer Science & Engineering) with Hons from Shobhit University and B-Tech (Computer Science & Engineering) from SCRIET, Meerut (U.P.). During this short period of time, he has been supervised several dissertation of M.Tech students. He has been member of several academic and administrative bodies. During his teaching he has been coordinated many Technical fests and National Conferences at Institute and University Level. He has attended several seminars, workshops and conferences at various levels. His many papers are published in various national, international journals and conferences. His area of research includes Cloud Computing, Wireless Sensor Network (WSN), Network Security, Neural Network, Artificial Intelligence and MANET (Mobile Ad-Hoc network).

Ms. Priya Agrawal is working as a Lecturer in Computer Engineering department at Indrashil Institute of Science & Technology, Rajpur, Kadi (Gujarat). She is Masters of Engineering (Computer Engineering) pursuing from Indrashil Institute of Science & Technology, Rajpur, Kadi (Gujarat). She has attended several seminars, workshops at various levels. Her area of research includes Data Mining.