

EMULATORS vs Real Devices

USER EXPERIENCE AND USABILITY

- User Interactions
- Real occurring events
- Overall performance
- Consistency in results

SPECTRUM OF DEVICE CONFIGURATIONS (HARDWARE)

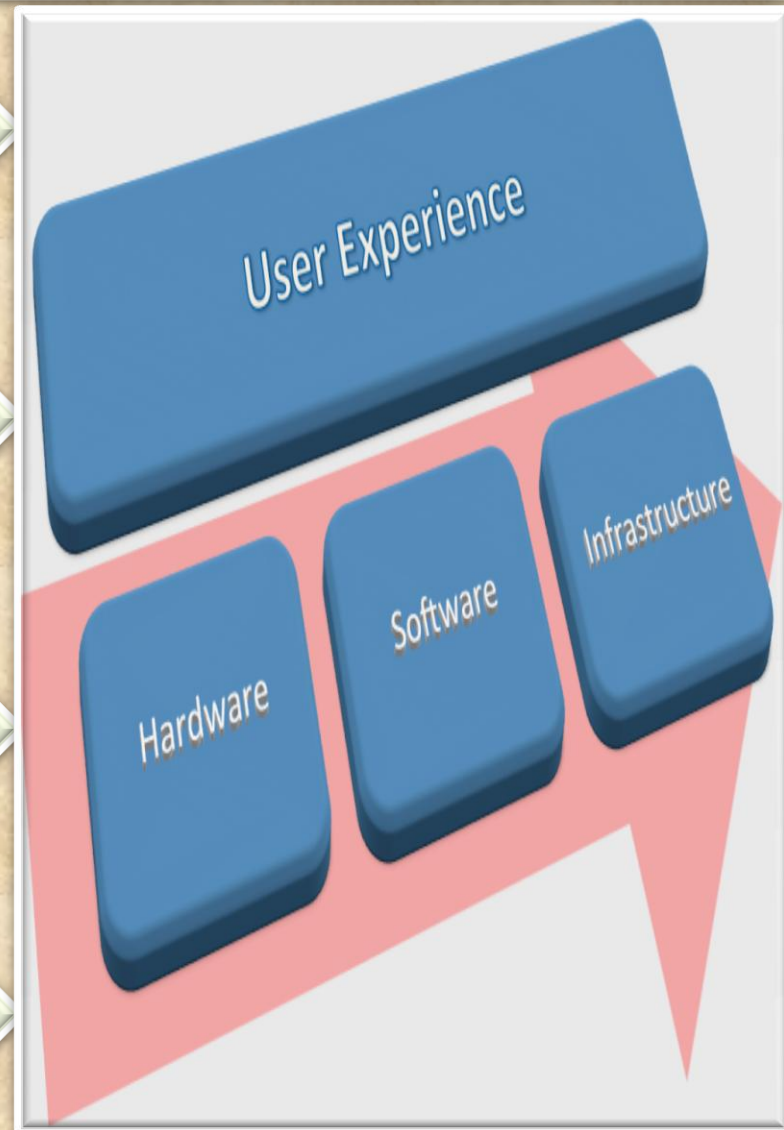
- Chipset
- Display
- Memory
- Sensors

PLATFORM + CUSTOMIZATIONS (SOFTWARE)

- Platform/OS version
- OEM and Carrier customizations
- *Dependencies to other software/app*

INFRASTRUCTURE (NETWORK)

- Network/WiFi



EMULATORS vs Real Devices Summary

Testing types	Device	Emulator
Unit Testing	●	●
Sanity & Acceptance	●	●
Functionality Testing	●	●
Interruption	●	●
Regression	●	●
Localization	●	●
Compatibility	●	●
Negative	●	●
User Acceptance Testing	●	●
Performance and security	●	●

Legend



Support - No



Support - Yes



Support - Partial

REAL DEVICE

EMULATOR

When there is a necessity to observe the behaviors of the software product under various adverse conditions such as memory shortage of the device;

In the process of finding obvious issues on several platforms, performing functional testing.

You need to test your app on as many devices as possible to ensure the maximal coverage either geographically or globally.

One of the main tasks of the mobile applications testing is usability testing, which is impossible without having the real device at hand;

Real mobile devices mean real user environment. Some definite actions like scrolling and zooming are to a great extent different on the touchscreen;

Only actual mobile device has the platform that is suitable for testing experiences of the end-users;

Usage of real devices is much needed in the process of testing the hardware characteristics, such as quality of the display. Besides, the best way of testing memory consumption is testing it on the whole range of actual devices:

It is impossible to monitor possible network issues with the help of an emulator. In this case you will definitely need an actual mobile device

EMULATORS : Mobile Automation- Desktop Based

Mobile test automation tools are installed on a desktop locally
Interacts with the mobile devices through USB connection or Wi-Fi
Mobile Apps can be installed & tested on both Emulators as well as real devices



A Typical set up for Mobile Test Automation

Overview: Mobile APPS

➤ Categories

➤ Types

➤ Distribution/Installation/Logs

➤ Mobile Test Industry Standards

➤ Remote Device Access (RDA)

➤ Emulators

➤ **Simulators**

➤ Troubleshooting Guide

➤ App Risk Analysis

Simulators : MOBILE

Mobile Simulator is a software application for a personal computer which creates a virtual machine version of a mobile device, such as a mobile phone, iPhone, other smartphone, or calculator, on the computer.

The mobile simulator allows the user to use features and run applications on the virtual mobile on their computer as though it was the actual mobile device.

A mobile simulator lets you test a website and determine how well it performs on various types of mobile devices.

A good simulator tests mobile content quickly on multiple browsers and emulates several device profiles simultaneously.

This allows analysis of mobile content in real-time, locate errors in code, view rendering in an environment that simulates the mobile browser, and optimize the site for performance.

Mobile simulators may be developed using programming languages such as Java and .NET



A **SIMULATOR** sets up a similar environment to the original device's OS, but doesn't attempt to simulate the real device's hardware.

Some programs may run a little differently, and it may require other changes (like that the program be compiled for the computer's CPU instead of the device's), but it's a close enough match that you can do most of your development against the simulator.

Simulators Pros and Cons



1. Study the behavior of a system without building it.

2. Results are accurate in general, compared to analytical model.

3. Help to find un-expected phenomenon, behavior of the system.

4. Easy to perform 'What-If' analysis.

1. No support for placing or receiving actual phone calls.

You can simulate phone calls through the emulator console, however.

2. No support for USB connections

3. No support for camera/video capture (input).

4. No support for determining connected state

5. No support for Bluetooth

6. No support for actual GPS

7. No support for Accelerometer feature used in Gaming applications

SIMULATORS vs EMULATOR

Functionality	Emulator	Simulator
What it mimics ?	Mobile device software Mobile device hardware Mobile operating system	Internal behavior of the device. It does not mimic hardware.
How to get it ?	It is generally provided by the device manufacturer.	It is generally provided by the device manufacturer or some other company.
Internal structure	It is written in machine-level assembly language.	It is written in high-level language.
Debugging	It is more suitable for debugging.	It is not suitable for debugging purpose.
Performance	Emulators are really slow. Emulating the actual hardware usually makes the software run slower than it would natively.	Faster than emulators.
Example	Google's Android SDK	Apple's iOS Simulator

SIMULATORS vs EMULATOR vs Real Device SUMMARY

Functionality	Real Device	Emulator / Simulator
Price	Getting real devices will cost you a lot.	It is almost free, we just need to download and install them
Processing Speed	It has faster processing; however network latency may be normal.	It is slower as compared to actual devices. It has observed less latency than real devices connected to the local network or in the cloud.
Debugging	Debugging is not that easy.	It provides step-by-step debugging of an application. Also, it provides an efficient way for capturing screenshots.
Web-app Testing	Web applications can be tested in a normal way.	Testing a web application is much easier.
Reliability	Testing on a real device has a major advantage that it always gives accurate results.	It cannot simulate all types of user interactions; hence it may lead to false results sometimes. So it scores low when it comes to reliability.

SIMULATORS vs EMULATOR vs Real Device

SUMMARY EXTRA “ADVANTAGES”

Issue	Emulator /Simulator Testing	Real Device Testing
Situation-based application	There are specific situations where the deadline to produce text execution results are short and purchasing the required mobile devices may be not possible.	The real device allows the testers to test almost all the real time scenarios which can be tested for the mobile applications. These devices are operated using fingers and simulate real-life usage.
Feeling of closeness towards the real handheld devices	The wide gamut of mobile devices creates the problems, whereby the testers are not confident about which mobile devices to invest in.	The real device allows the testers to test even usability
Ease of availability	Emulator/simulator(s) are in most cases open and free software	The real devices allow stringent performance testing issues like working with a real time transport application for 15 hours continuously which cannot be successfully simulated by the emulators.
Ease of opening an Web application through URL	It is easier to do web application testing when it comes to opening the web application. The user just needs to copy and paste the application URL.	Testing on real devices provides more in terms of reliability.

SIMULATORS vs EMULATOR vs Real Device

SUMMARY EXTRA "ADVANTAGES"

Issue	Emulator Testing	Real Device Testing
Capturing screenshots of the situations where defects appear	Capturing issue of screenshots over simulator is very easy with the simulator since we just need to use Microsoft office facilities.	Testing with real world devices is very helpful in terms of interoperability testing.
Simulation of validation of battery scenarios	The emulator/simulators are not able to simulate the battery issues.	Real world devices can easily perform the same.
Validation of incoming interrupts	The emulator/simulators are not able to simulate the incoming interrupts for SMS as well as the incoming calls.	Real world devices can easily simulates incoming interrupts.
Validation of exact color displays	The emulator/simulator is not able to properly emulate/simulate the exact color display of the devices when the real device is in sunlight or in black.	Real world devices can easily simulates the exact color displays.
Validation of the performance	The performance of the emulator/simulator tends to be slower than the original devices at times.	The original devices tend to perform faster than the emulator or the simulators.
Simulating memory related issues	The memory available at the emulator/simulator tends to be far more than in the real devices	The memory storage level of the devices tend to be far less than the emulators thus it may

SIMULATORS vs EMULATOR vs Real Device

SUMMARY EXTRA “DISADVANTAGES”

Emulator/Simulator Testing	Real Device Testing
<p>The emulator/simulator is not always the best type of solution for scenarios such as the ones whereby the testing team needs to validate the performance of the application for a longer period of time.</p>	<p>The real devices are costly compared to the emulator/simulators. Thereby projects under budget and timeline constraints may risk profitability as well as the viability of the overall project.</p>
<p>The emulator/simulator is suitable mostly for certain types of functional test case executions.</p>	<p>There is a very wide variety of mobile devices from apple to Samsung to android and to Symbian and so on.</p>
<p>The emulator/simulator can some time not be supportive of certain types of application</p>	<p>Real Mobile devices when used in the developing stage for unit testing and similar purposes could turn out to be harder to connect to the IDE than the emulators</p>
<p>Not all the emulator/simulator supports the complete gamut of mobile applications.</p>	<p>In order to test with the real world devices, the devices need to be always connected to the USB port of the machines.</p>

Simulators : iPhone

iPhone and iPad are popular devices from Apple. Its has its own proprietary operating system, iOS. Its well known for its design and performance.

Prerequisites:

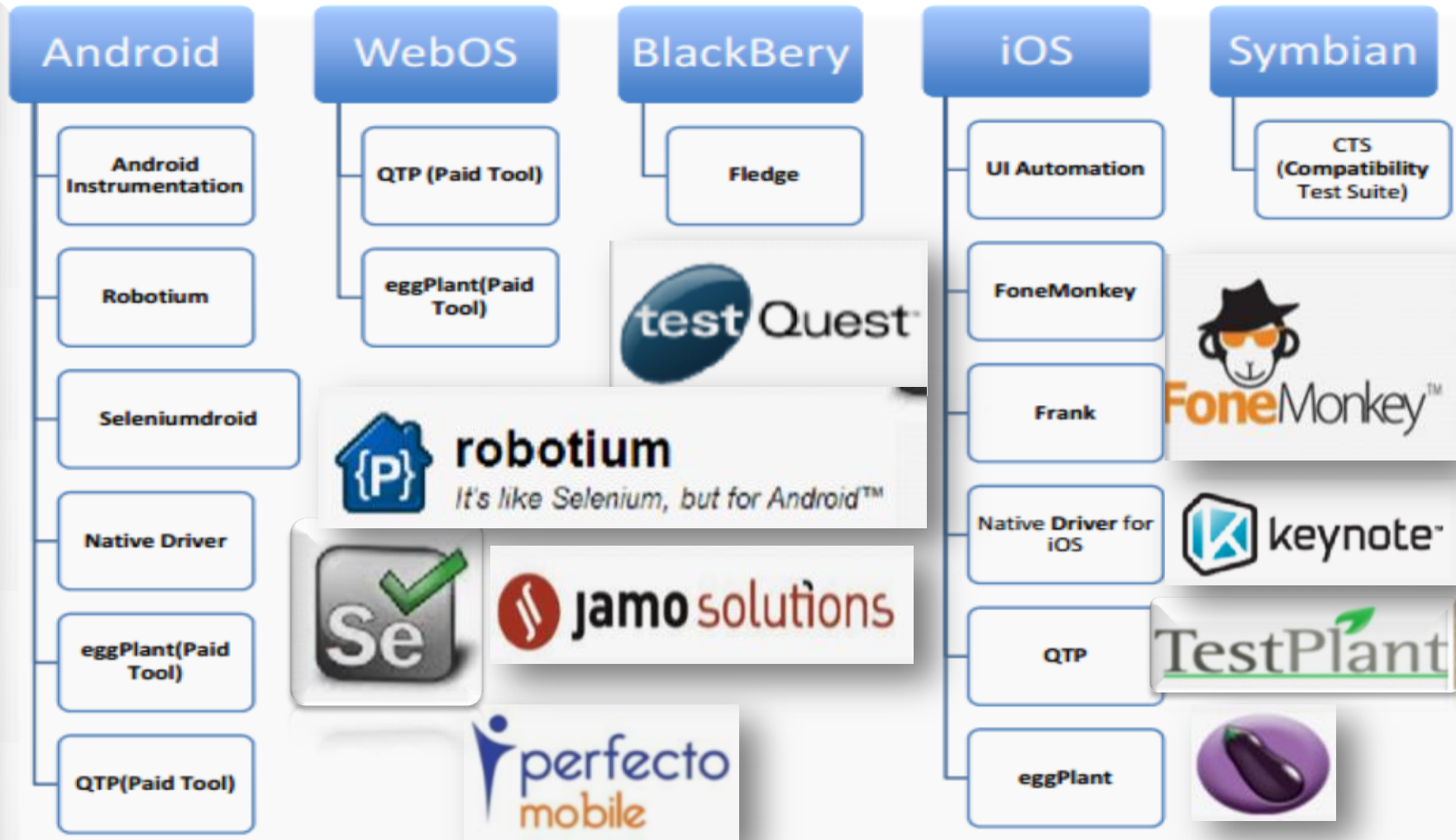
MAC Environment

Getting Started:

- Deploy Native app into iPhone Simulator using Xcode.
- Xcode - an integrated development environment that manages your application projects and lets to edit, compile, run, and debug code
- Once the native app is deployed, the app icon appears on simulator, thus aids to access the app directly from simulator
- Testing in UIAutomation can be done by both developer and tester, it requires knowledge of iPhone development.
- UIAutomation is the automation technique for iPhone.
It uses JavaScript as test script to perform automation. This can be done both in device and simulator
- iPhone Simulator does not emulate an iPhone processor, disk drive, memory constraints

Most Popular Mobile App Testing Tools

Existing Mobile Platforms And Available Test Automation Tools



Overview



iOS

Android

ANDROID



WHAT'S NEW in Oreo 8.0 (released Aug 21, 2017)

ANDROID OREO 8.0



Zippy performance



Picture-in-picture is a great add-on



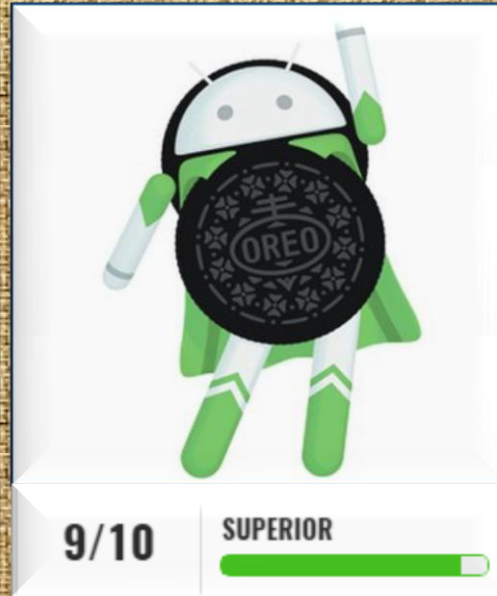
Smarter notifications



Cleaner settings menu



Performance improvements : much faster



Notification dots are annoying



Fragmentation remains an issue



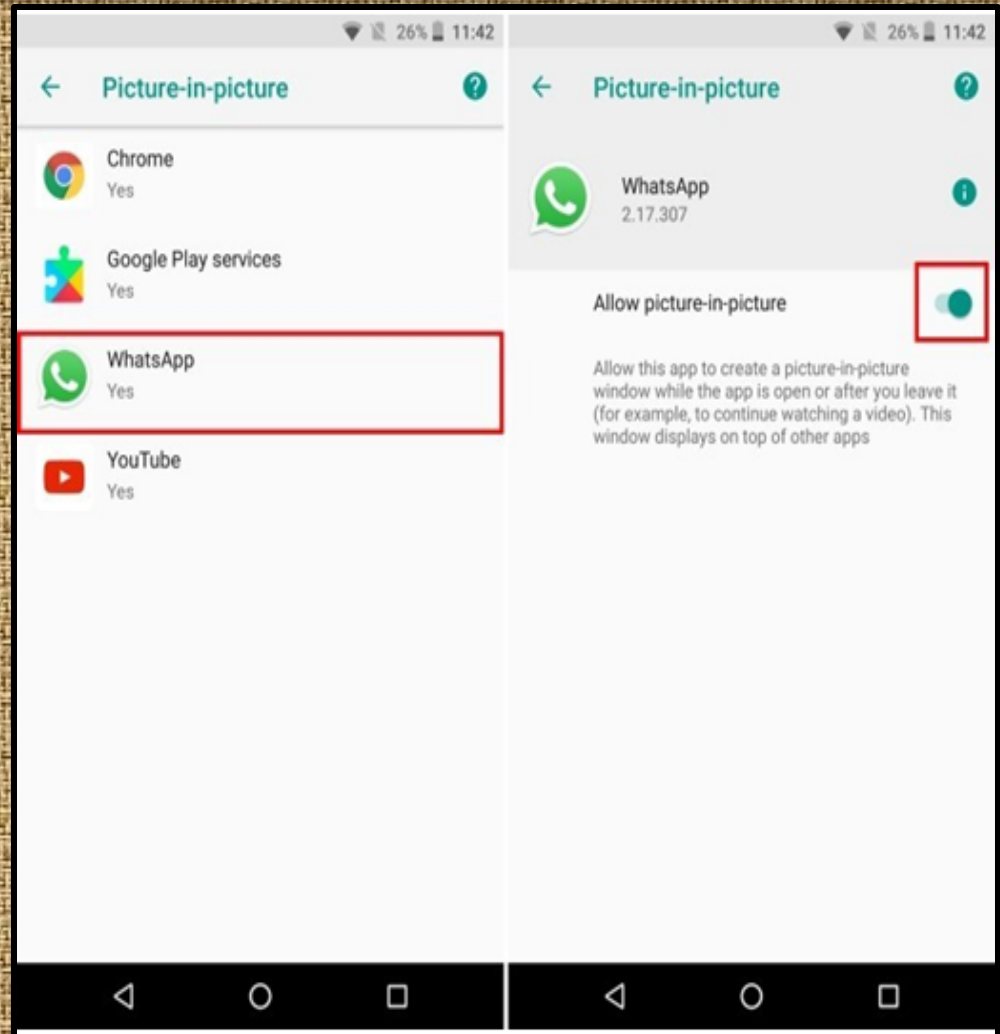
ANDROID OREO 8.0

1. Picture in picture

You can move the video around the screen as you wish and then send the video away by pushing it to the bottom of the screen.

This feature will be especially welcome on tablets, so you can better use that extra screen real estate.

You can now use the **Picture-in-Picture** mode for the apps that you wish to



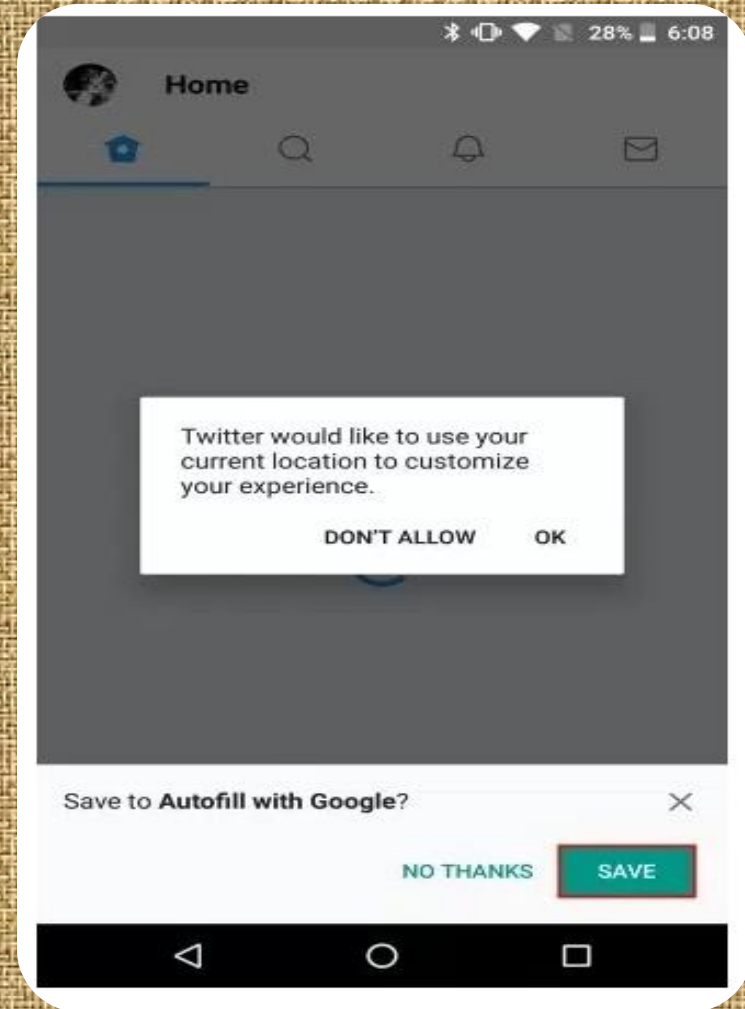
ANDROID OREO 8.0

2. AutoFill Inside Apps

You may be familiar with Google Smart Lock, which saves your passwords when you log-in for the first time into a website.

It then presents the same credentials to you the next time that you wish to log-in to the website. This feature has been extended to work within apps as well.

Google will basically save your passwords and offer to easily input them directly the next time you try to log into an app.

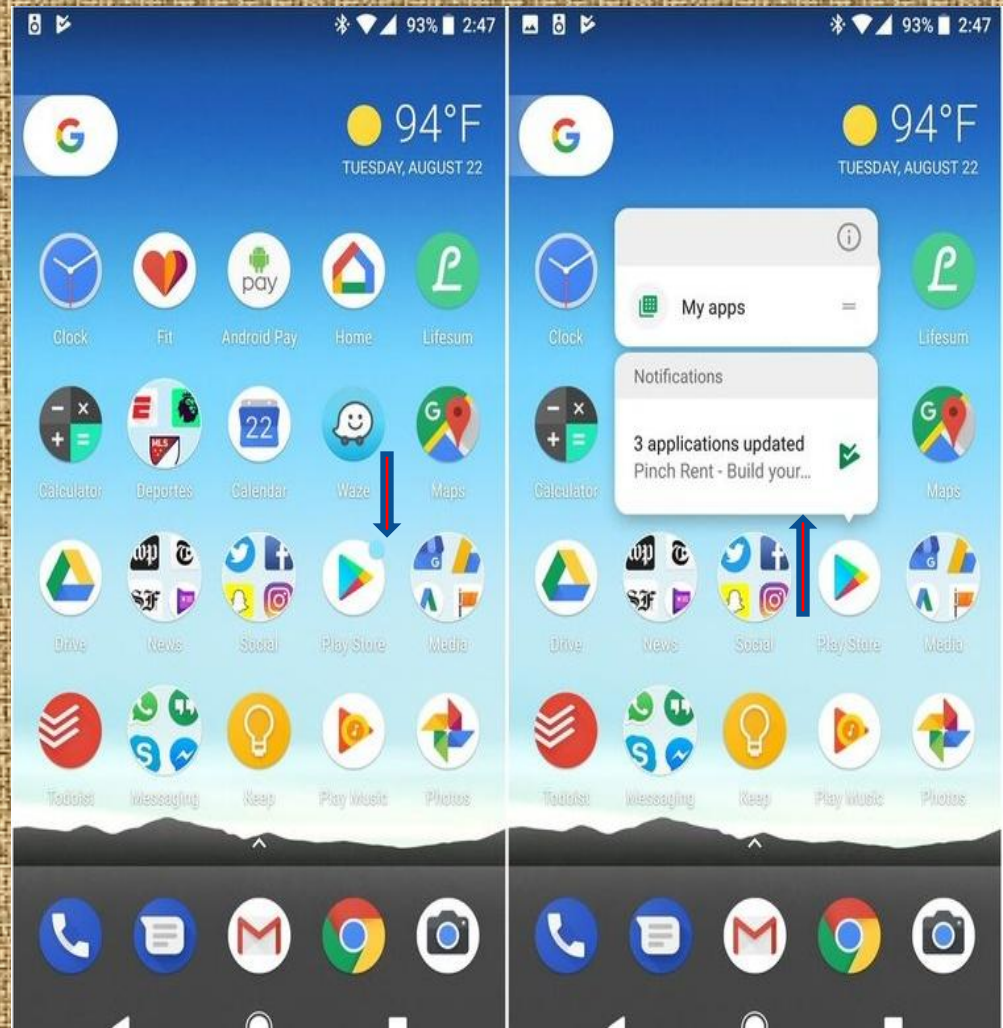


ANDROID OREO 8.0

3. Notification Badges

When you have an app with a notification, you'll now see a colored dot at the top of the icon.

When you touch and hold the icon, you'll see the contents of the notification and be able to swipe it away.

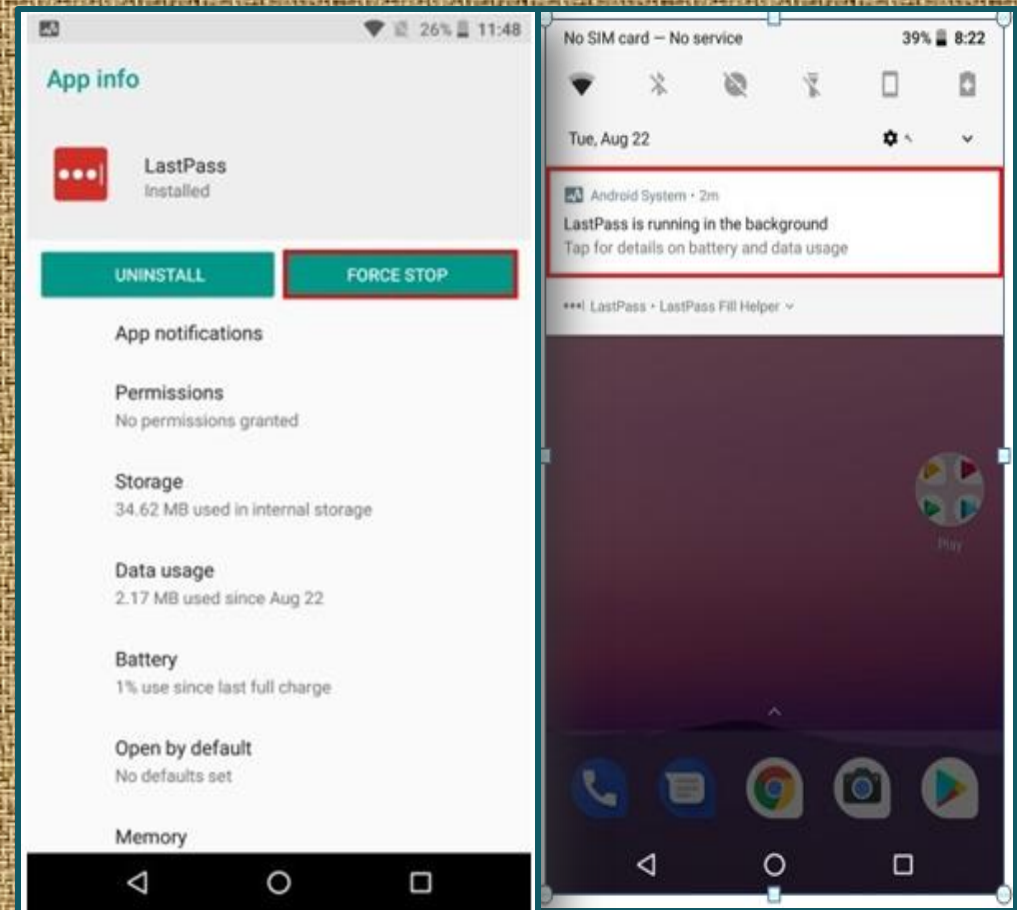


ANDROID OREO 8.0

4. Control of APPs running in Background

While Android Oreo features tons of battery optimizations, it also gives the user control over what apps should one allow to run in the background.

Android Oreo shows the user what apps are currently running in the background, and allows force close them.



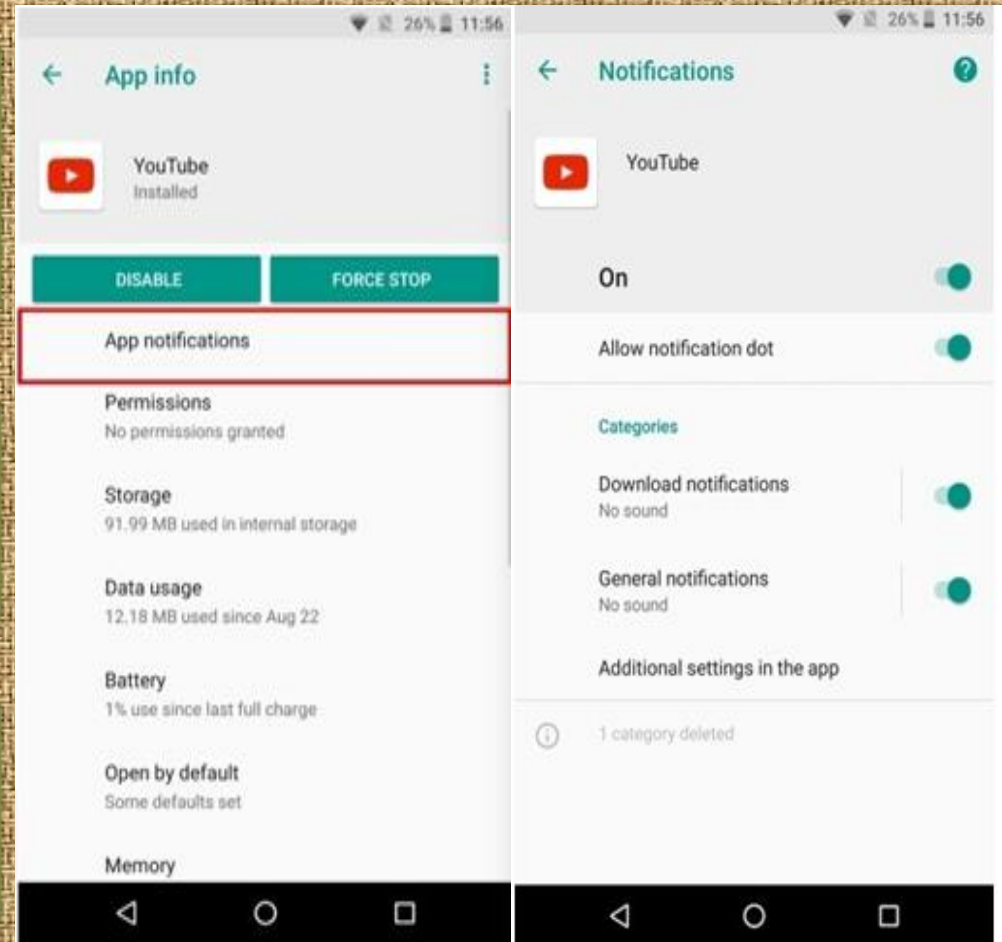
ANDROID OREO 8.0

5. Filter Notifications

New feature, that is Notification Channels. Prior, notifications were set as one collective set, and the user had the option to either disable them entirely or bear them altogether.

In Oreo, the notifications are now divided into categories, based on their priority or functioning.

User can now easily disable one category of app notifications while keeping the other one enabled.

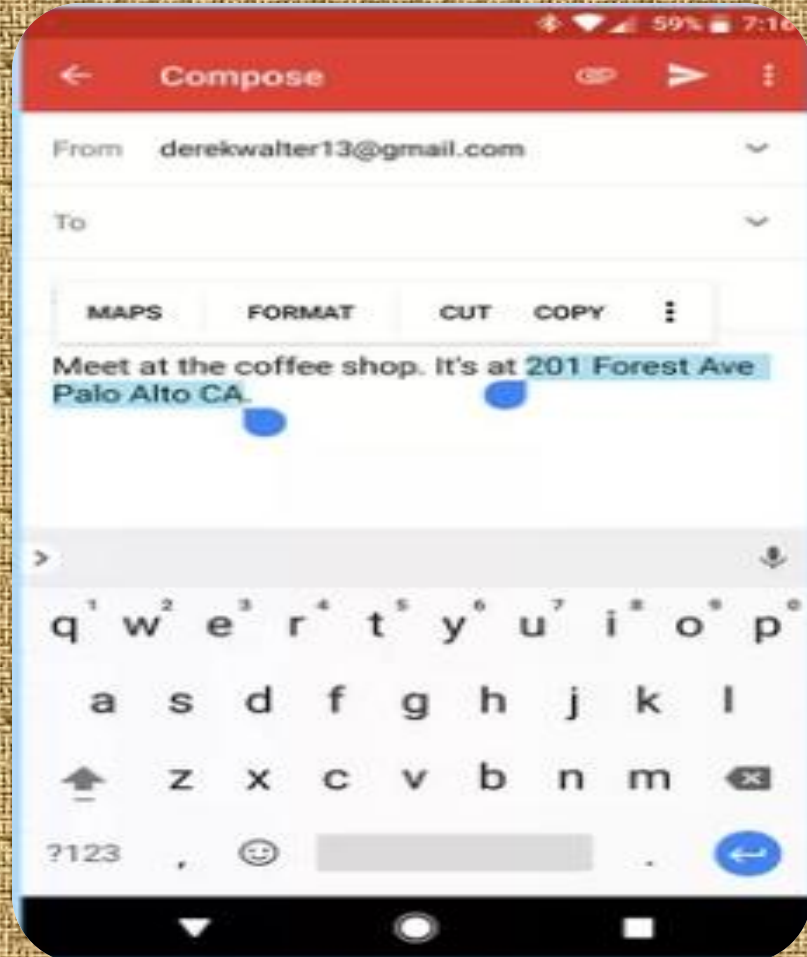


ANDROID OREO 8.0

6. Smart Text Selection

When you highlight a piece of text, the device will offer specific actions based on the content.

Example: if you highlight an address you'll see a choice to open that location in Google Maps.



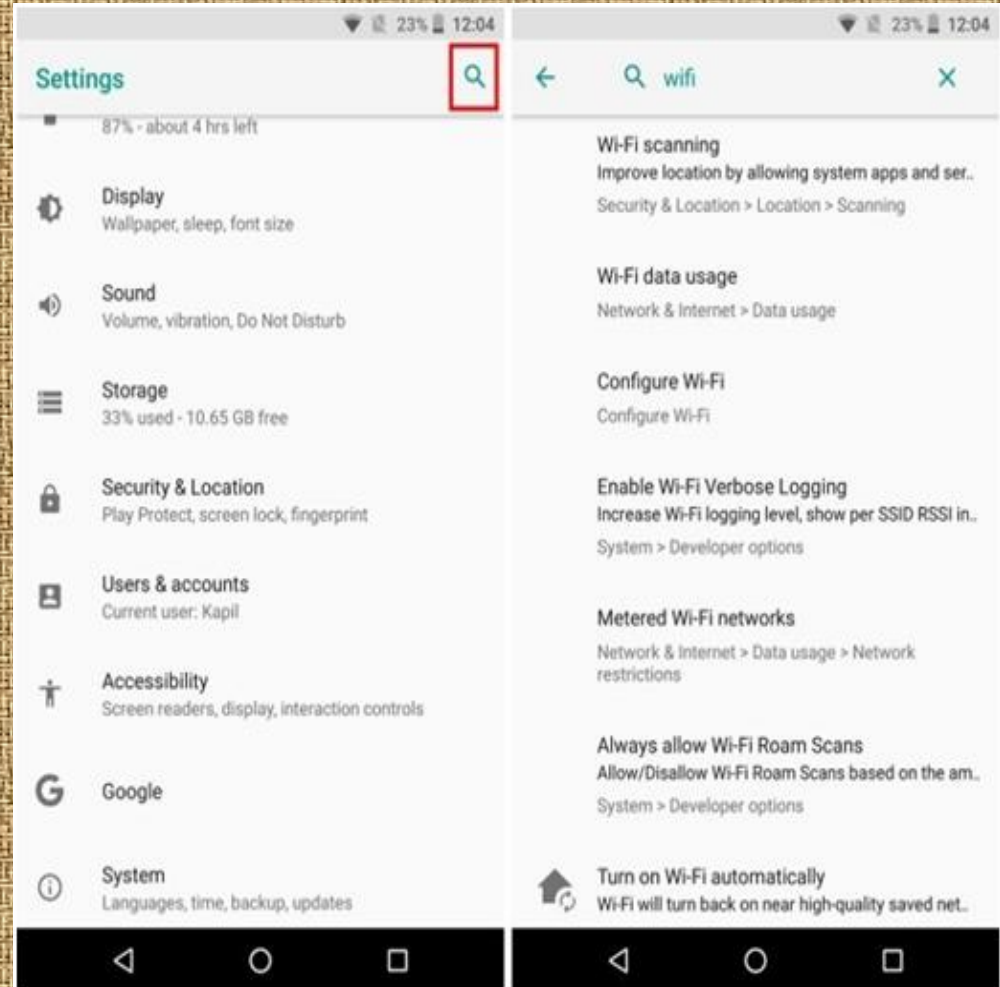
ANDROID OREO 8.0

7. Settings Navigation

When Android Nougat debuted last year, it brought a redesigned settings menu that made it much easier to find your way around.

Different sections are categorized in a more straightforward manner.

There's also a search feature, which is often the best way to find what you need.



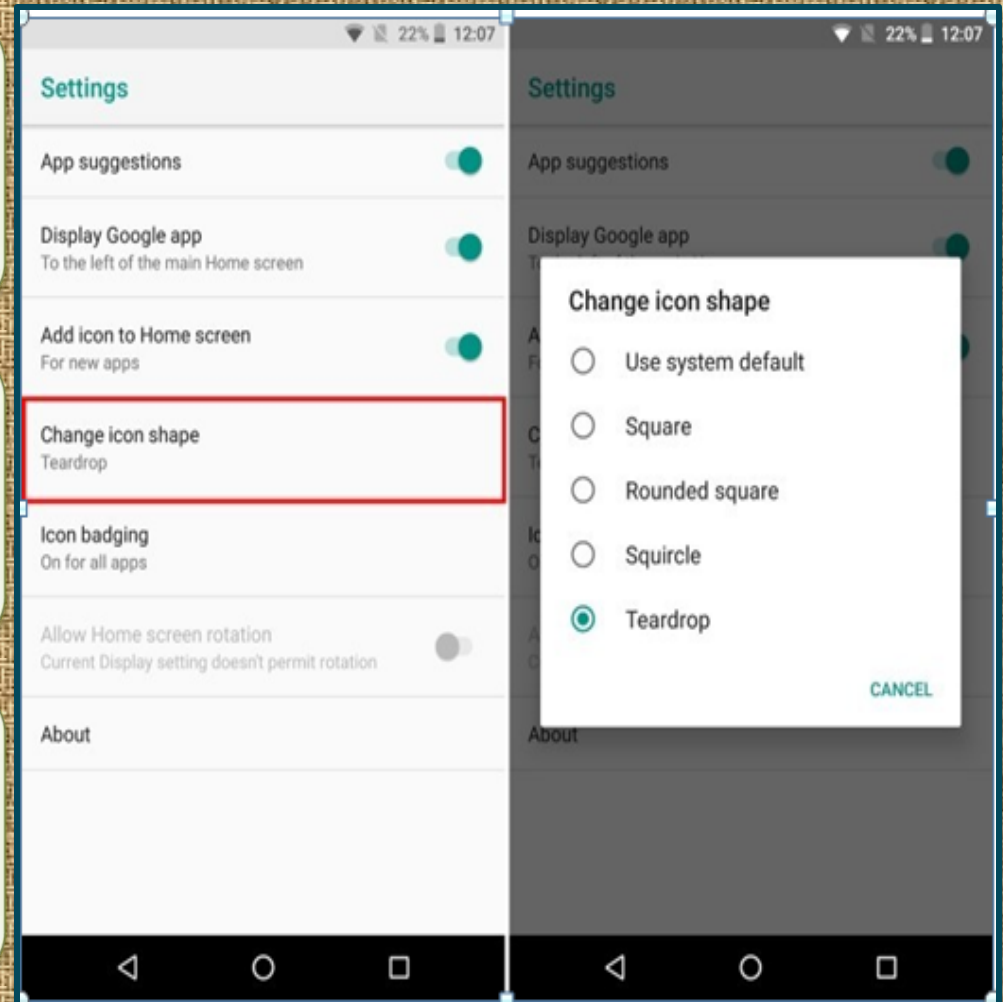
ANDROID OREO 8.0

8. Icons Shape

While the Pixel Launcher is highly regarded as the stock and basic Android Launcher, when coupled with Android Oreo, it also allows the user to change the overall look of the app icons.

Tap on **“Change icon shape”** to continue.

Choose from **System default**, **Square**, **Rounded square**, **Squircle**, and **Teardrop**.



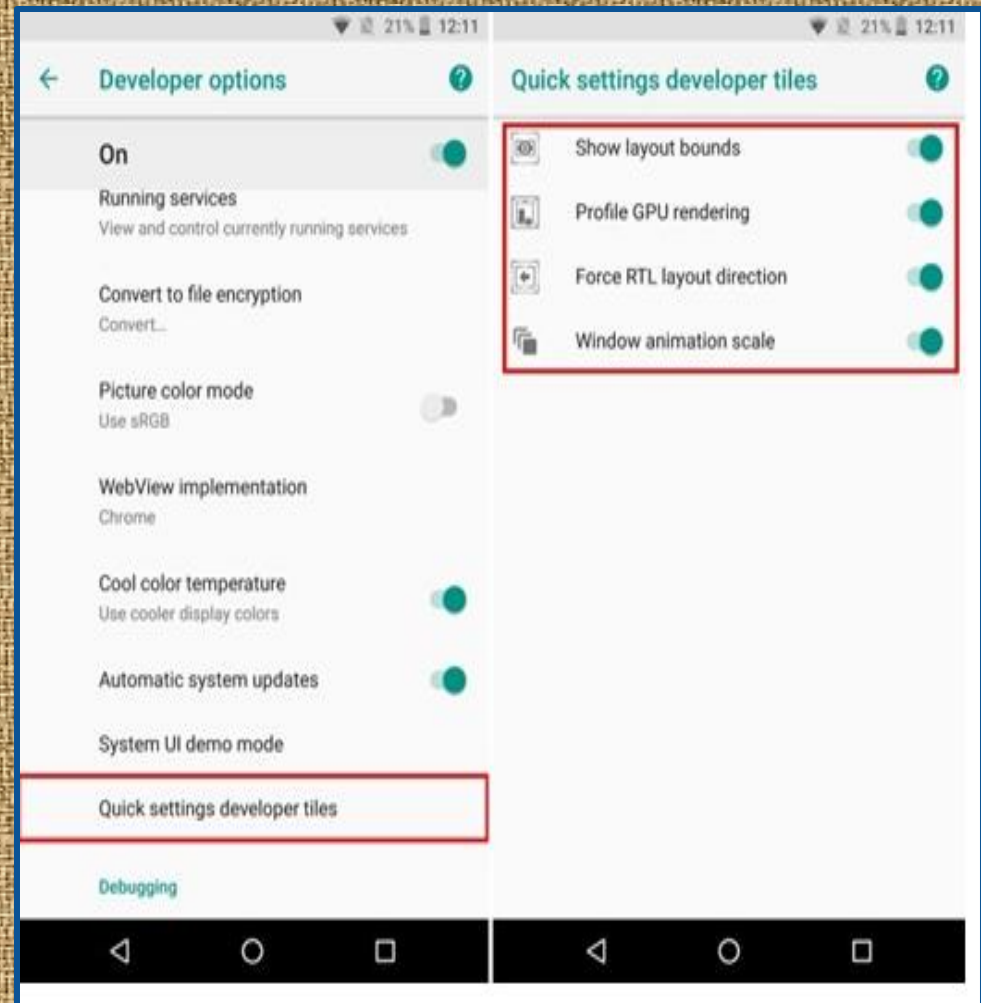
ANDROID OREO 8.0

9. Developer Quick Settings

Quick Settings on Android allow for easy toggling and/or access to various functions of the system.

Previously it was disabled by default but now you may enjoy it.

Now, you can enable or disable various developer related tiles such as **“Show layout bounds, Profile GPU Rendering, Force RTL layout direction, and Window animation scale”**.



ANDROID OREO 8.0

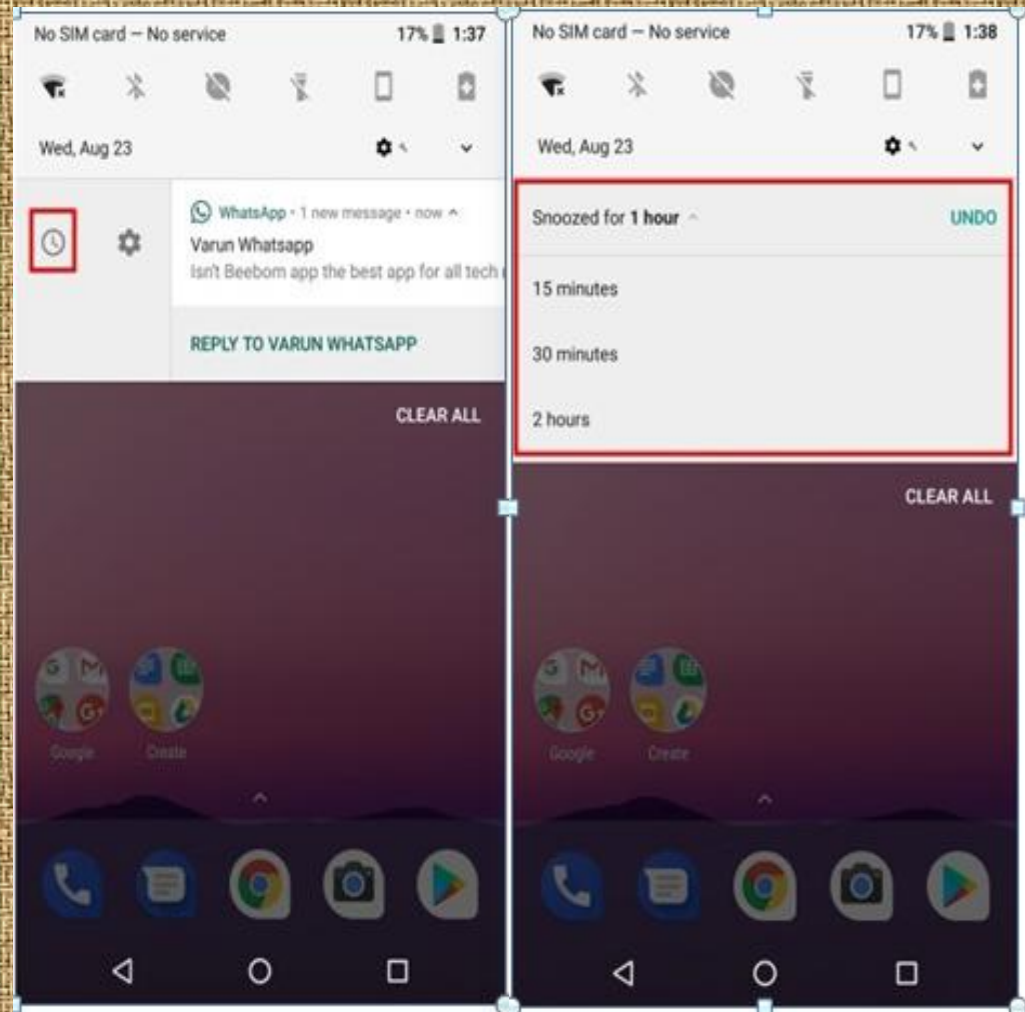
10. Snooze Notifications

Android notifications are great but there are times that it might come in the way of what you're currently doing.

You can simply snooze your notifications continue doing your work.

Once the timer runs out, you'll be presented with the same notification again.

Set it up to **15 min**, **30 min**, **1 hour** or **don't snooze**.



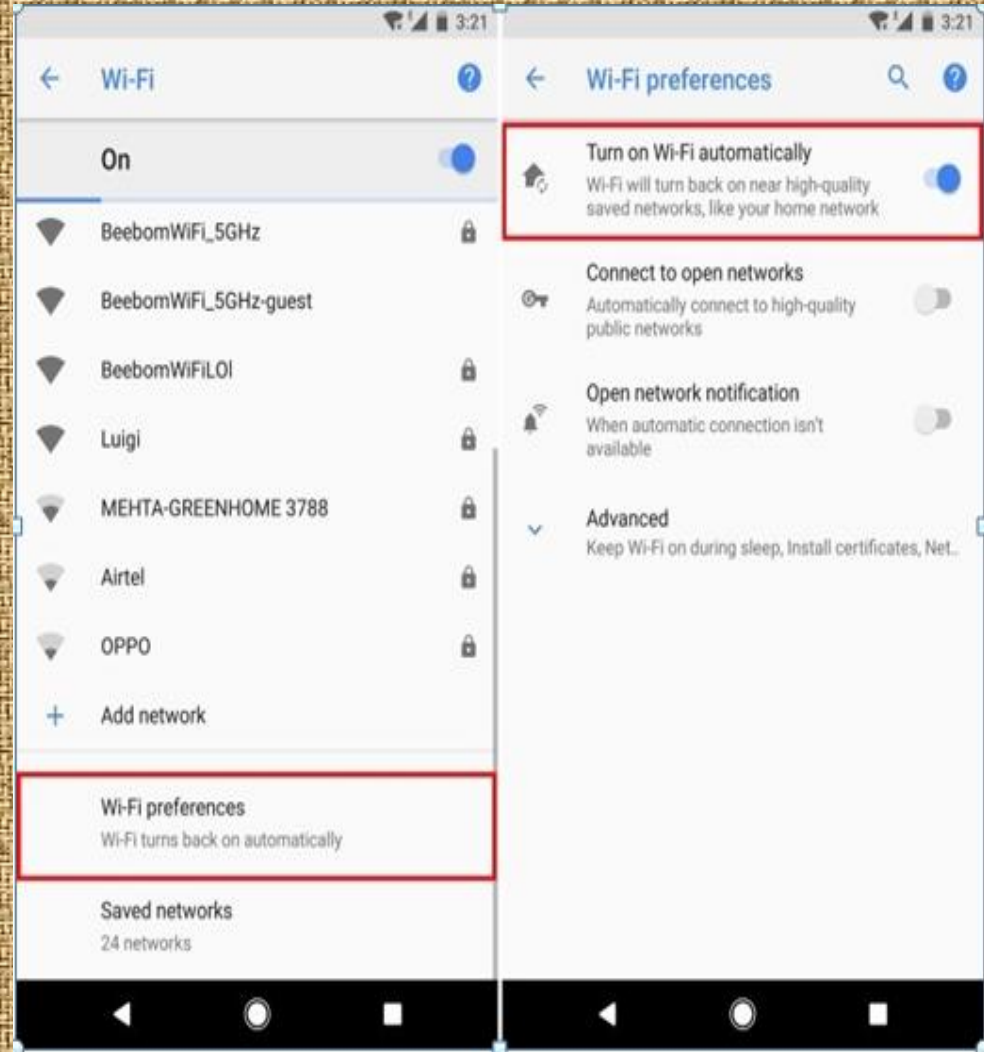
ANDROID OREO 8.0

11. Smarter Wi-Fi

Android Oreo comes with the ability to automatically turn on **Wi-Fi** when you're in home or office, where there is a high-quality network available.

It comes as a nifty little feature that while conserving battery, also allows you to automatically switch to **high-speed networks**.

This is a smart feature that should save you some battery life and keep you from having to constantly turn on and off **Wi-Fi** on your phone.



ANDROID OREO 8.0

12.Nighttime Mode

The Night Light mode debuted in Nougat, but now it's better in Oreo, with a slider bar so you can change the hue of the light.

Just as with Nougat, you can schedule times for the Night Light to pop up so you don't keep yourself up for all hours of the night.

