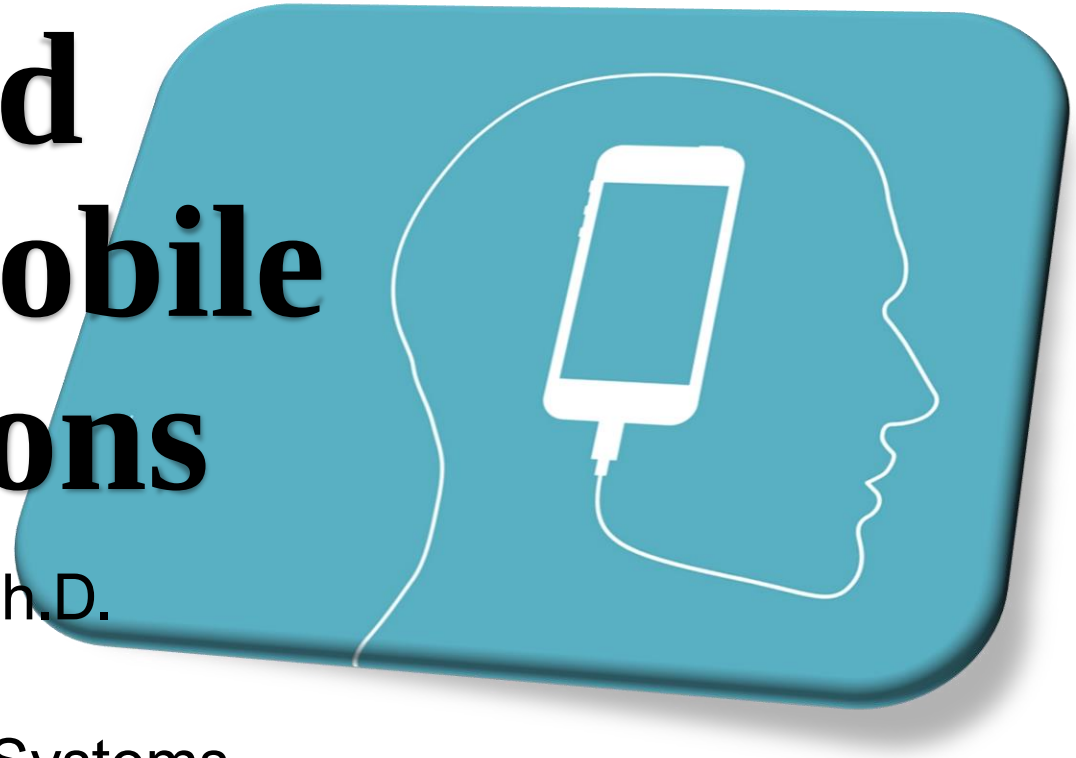
Android not only mobile applications



Florenc Demrozi Ph.D.

28/05/2021

Embedded Operating Systems

IoT4Care

UNIVERSITÀ di VERONA

Dipartimento di INFORMATICA

About Me

1991

2005

Enrollment in the three-year degree in Computer Science

2010

2014

B.Sc. in Computer Science

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

2016

M.Sc. in Computer Science and Engineering

UNIVERSITÀ di VERONA
PHD School in NATURAL SCIENCES AND ENGINEERING

2019

UF UNIVERSITY of FLORIDA

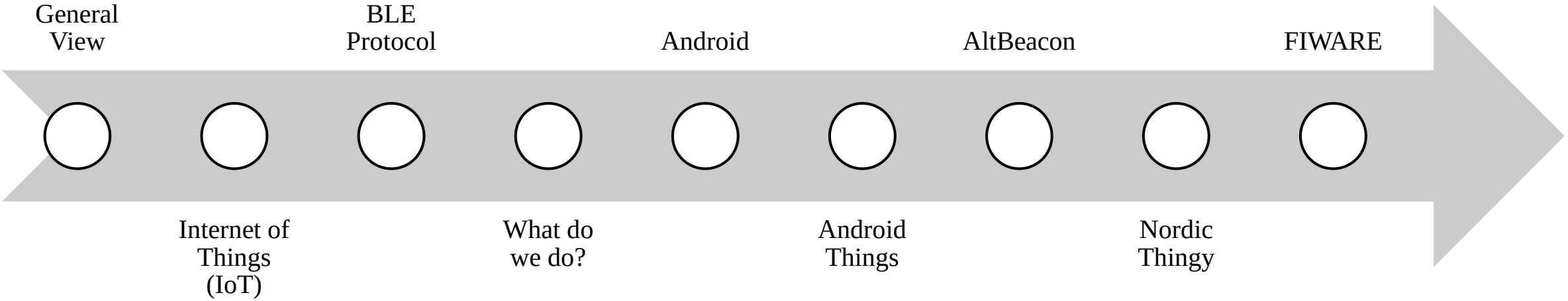
Visiting Scholar
Department of Biomedical Engineering (BME), i-Heal Lab

2020

Ph.D. in Computer Science

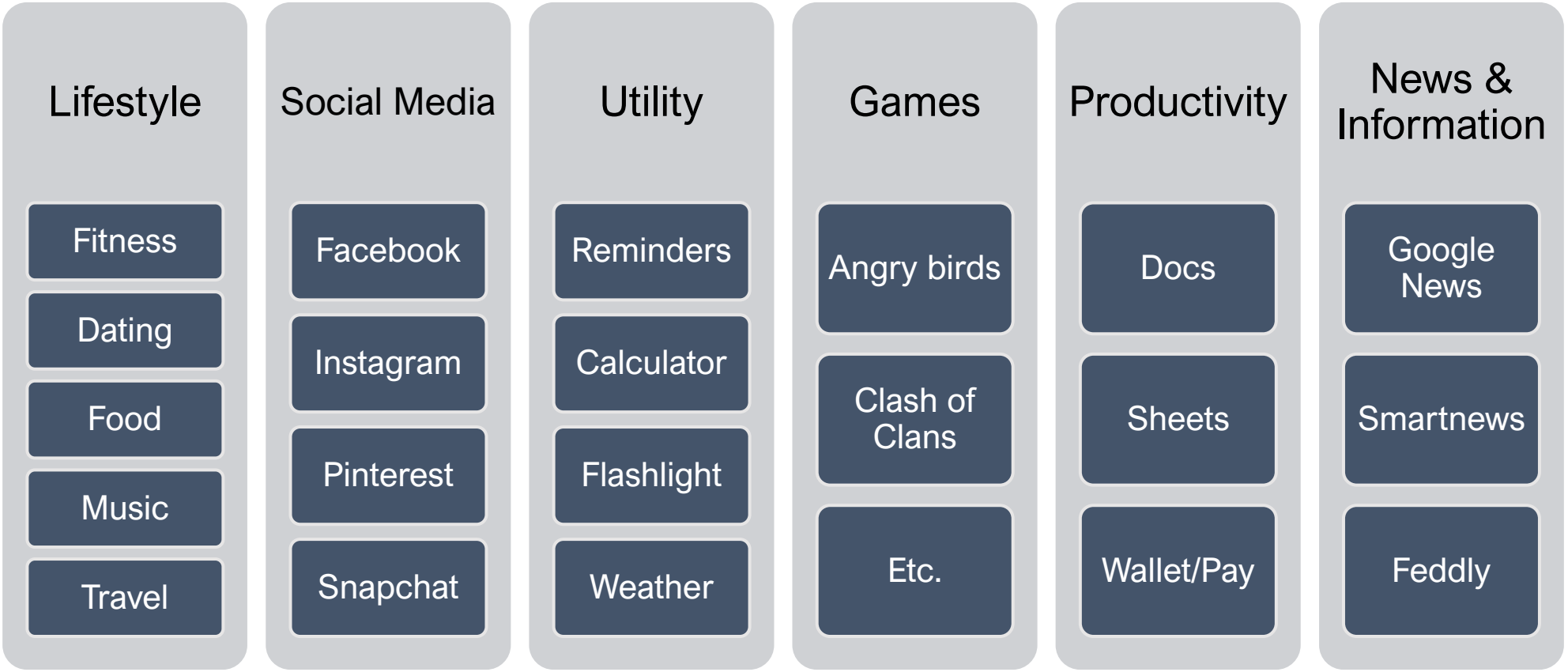
An IoT based Virtual Coaching System (VSC) for Assisting Activities of Daily Life

Outline

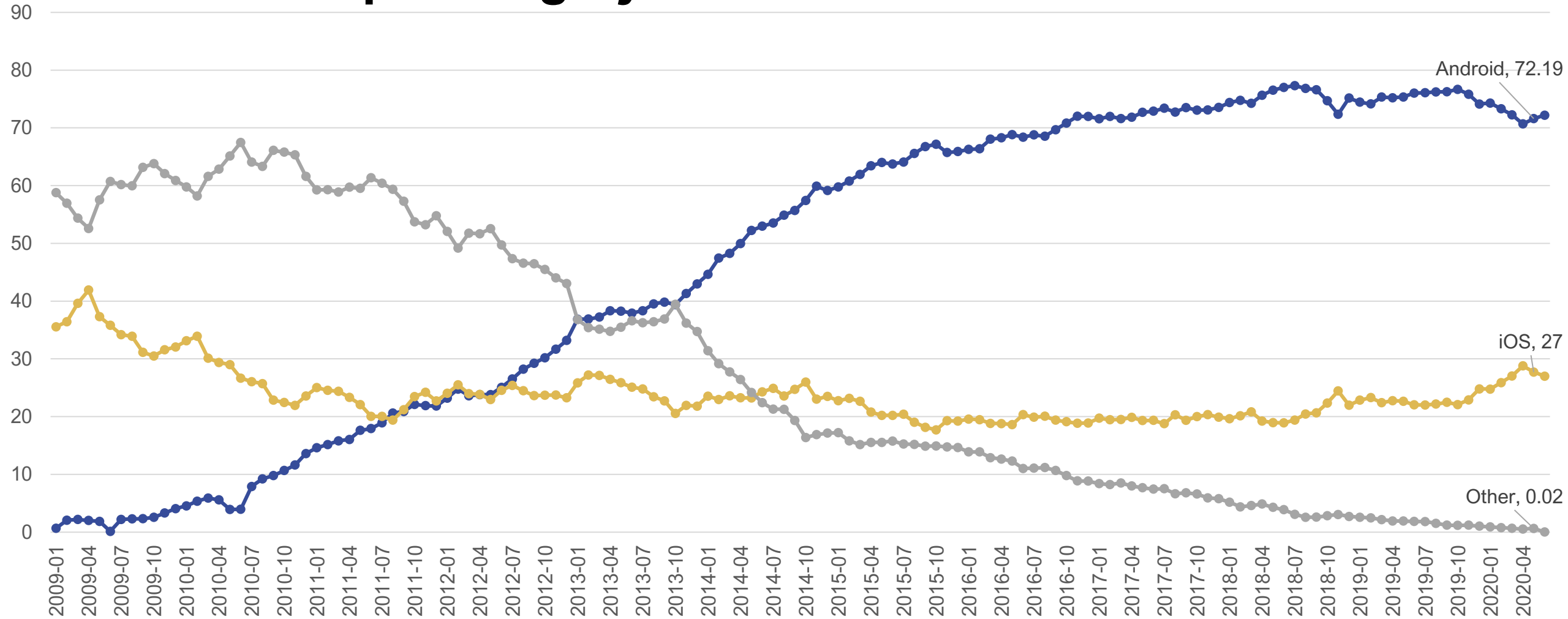

General View
Internet of Things (IoT)
BLE Protocol
What do we do?
Android
Android Things
AltBeacon
Nordic Thingy
FIWARE

Tell me some Android applications categories example.

Mobile applications categories

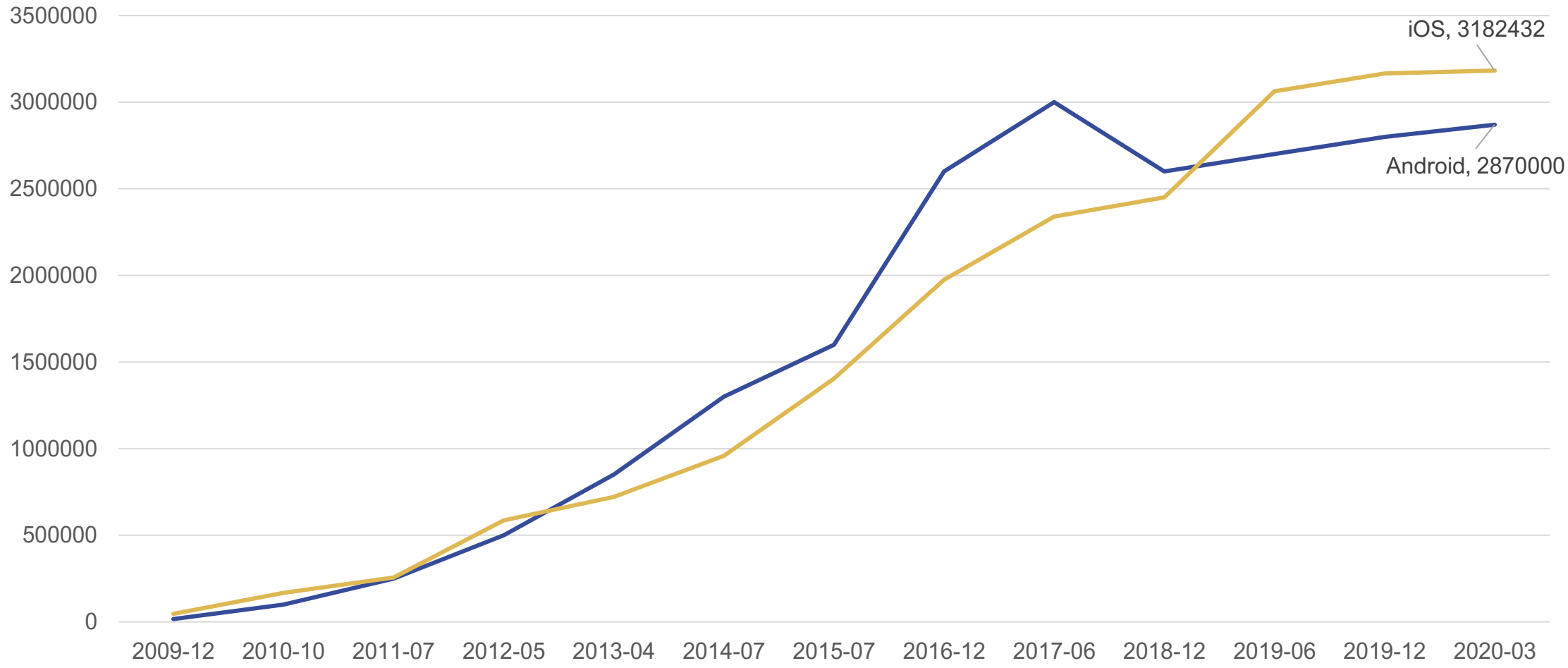


Mobile Operating System Market Share Worldwide[1]



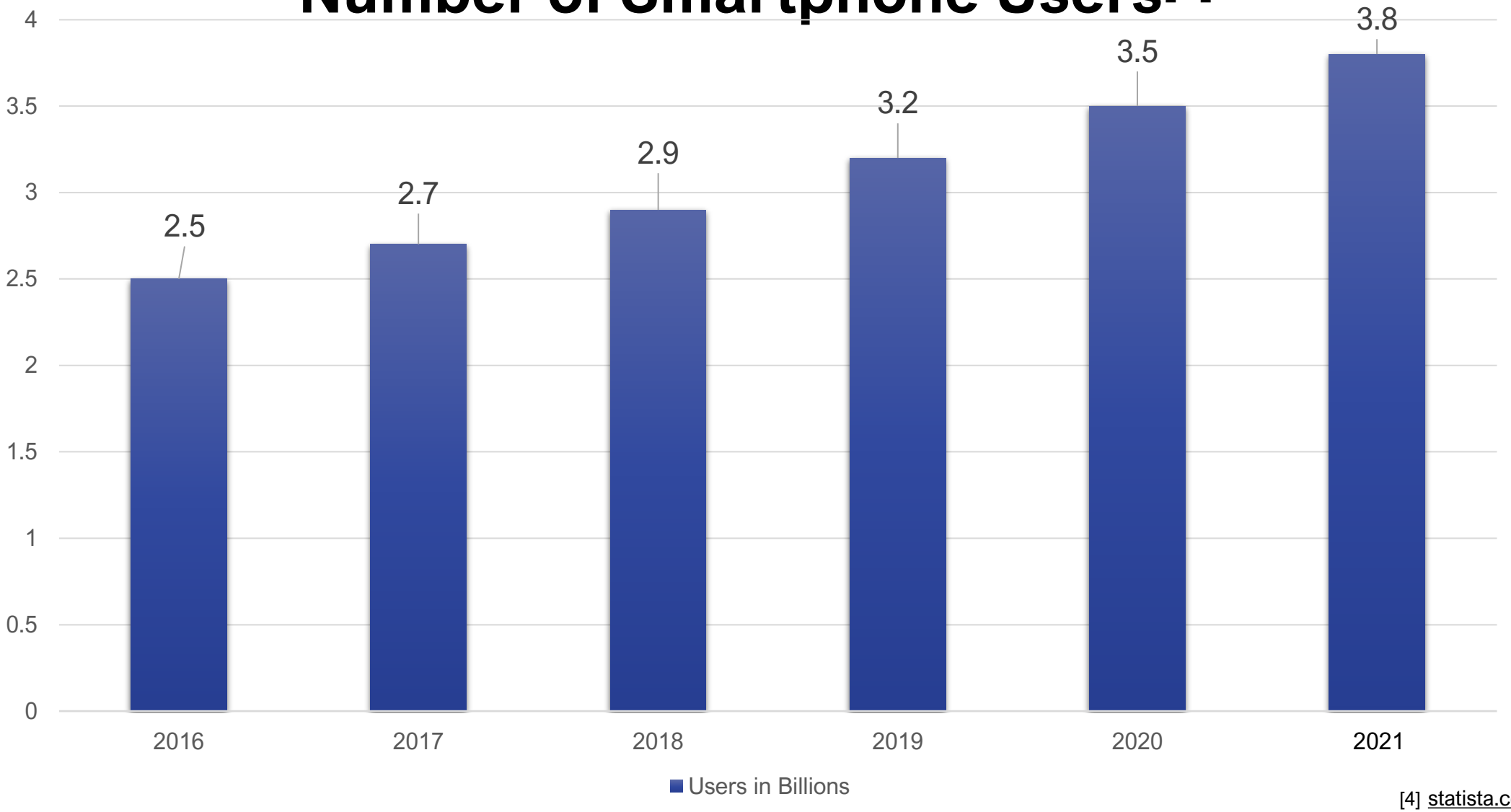
[1] gs.statcounter.com

Number of available applications: Dec. 2009 – Mar. 2020[2,3]



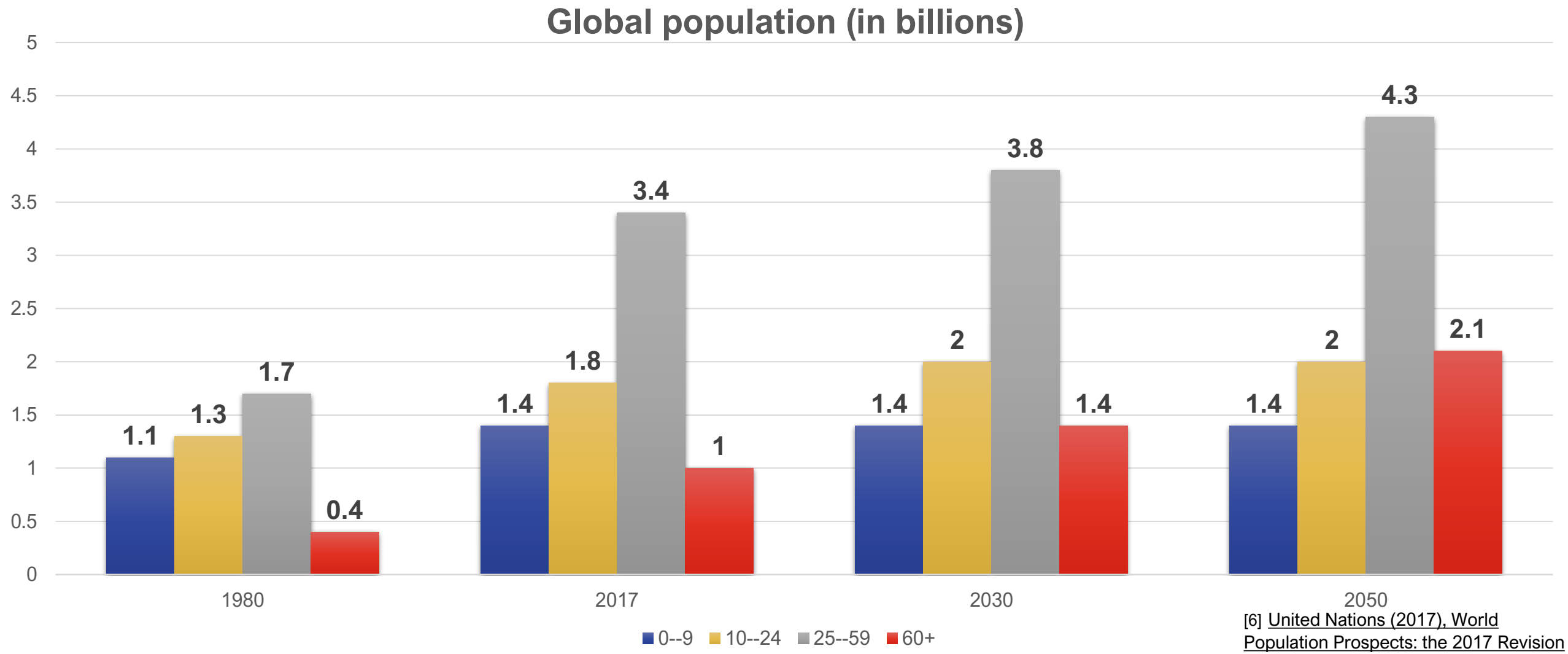
[2] statista.com
[3] statista.com

Number of Smartphone Users[4]



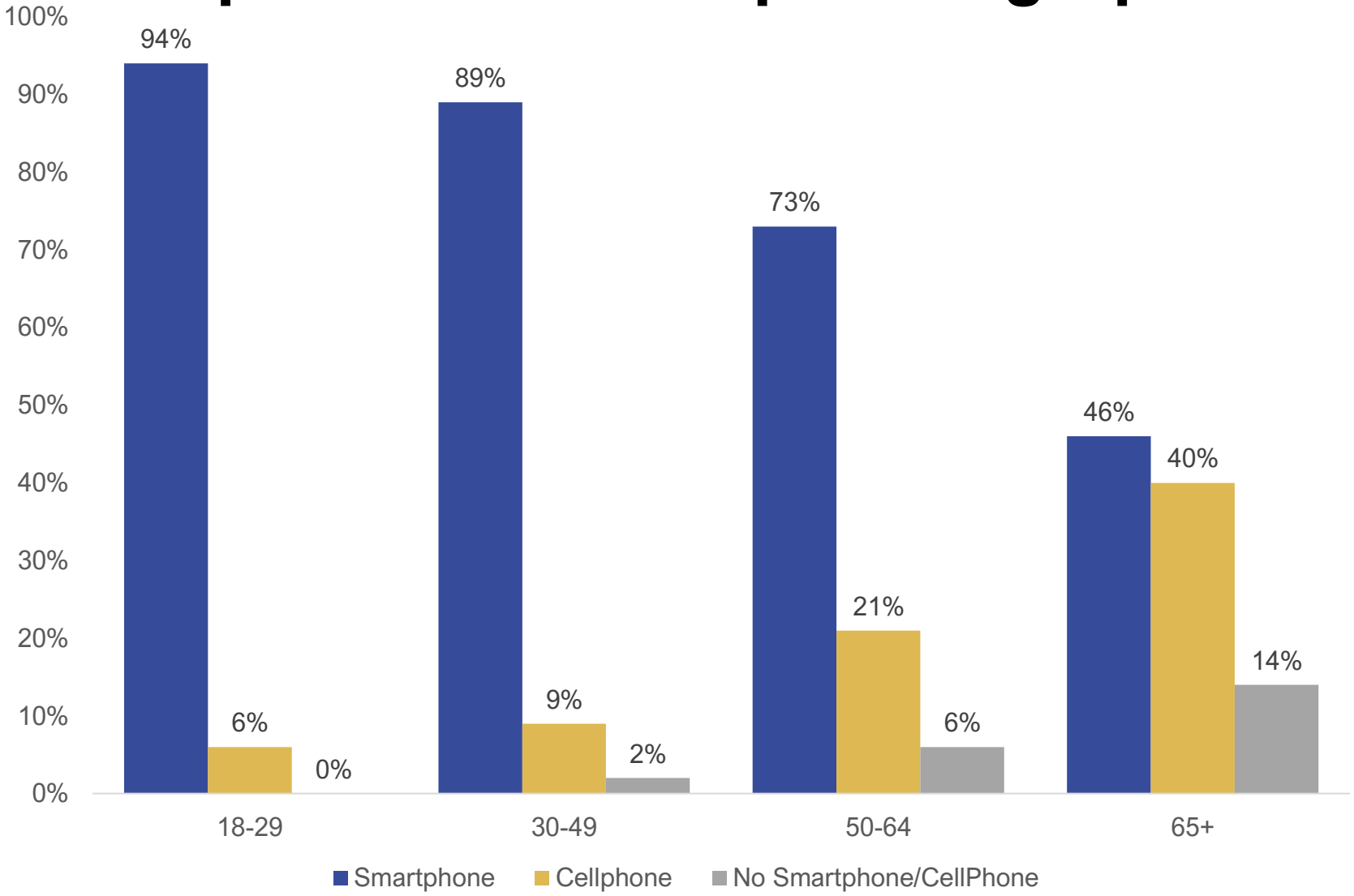
[4] statista.com

Aging of population[6]



[6] United Nations (2017), World Population Prospects: the 2017 Revision

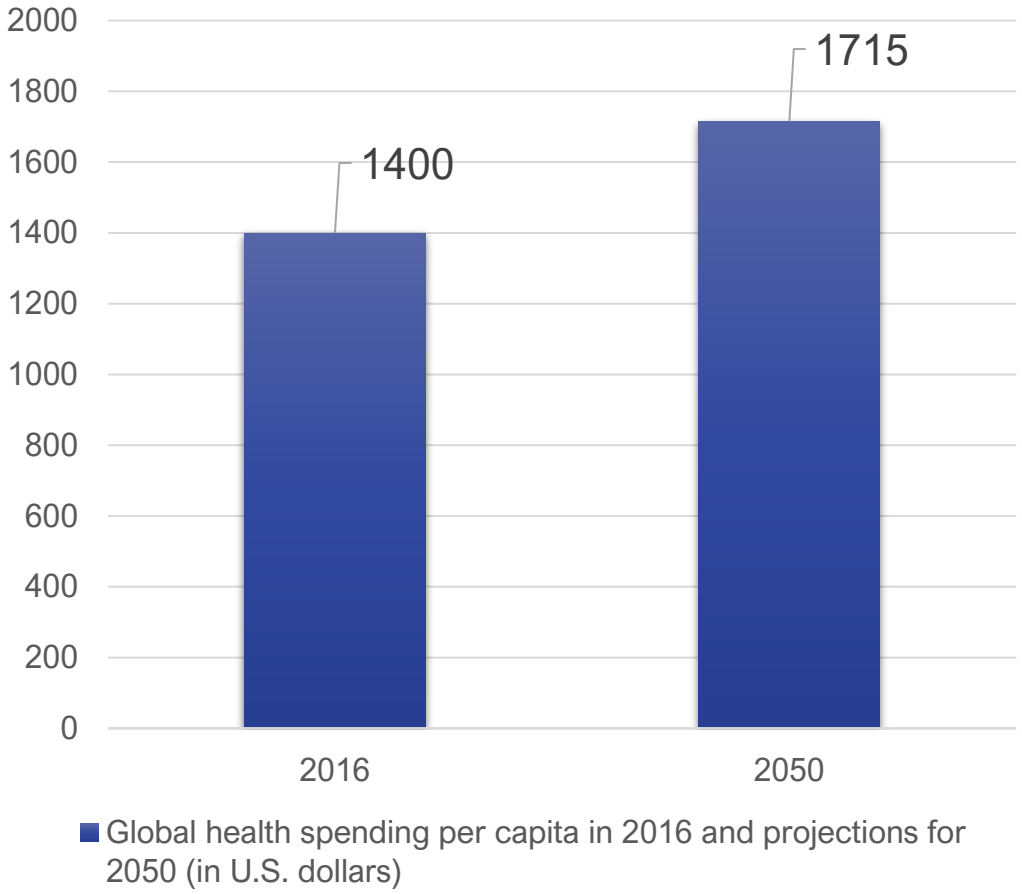
Smartphone Ownership Demographics[5]



[5] bankmycell.com

Main implications

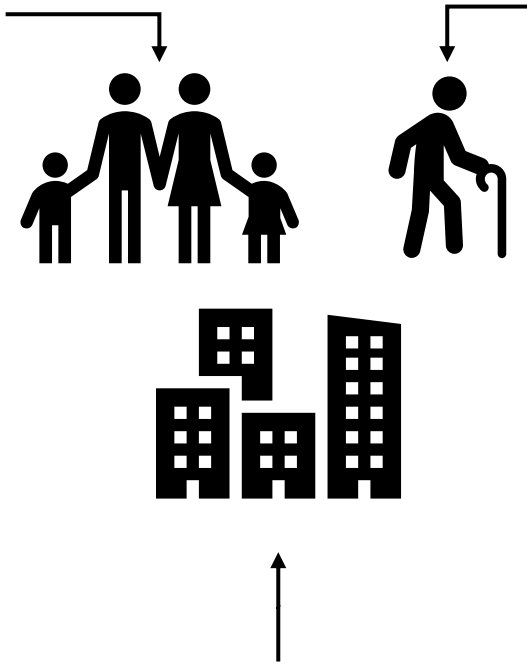
Health-Care System Cost[7]



Social System

Family members of elder people

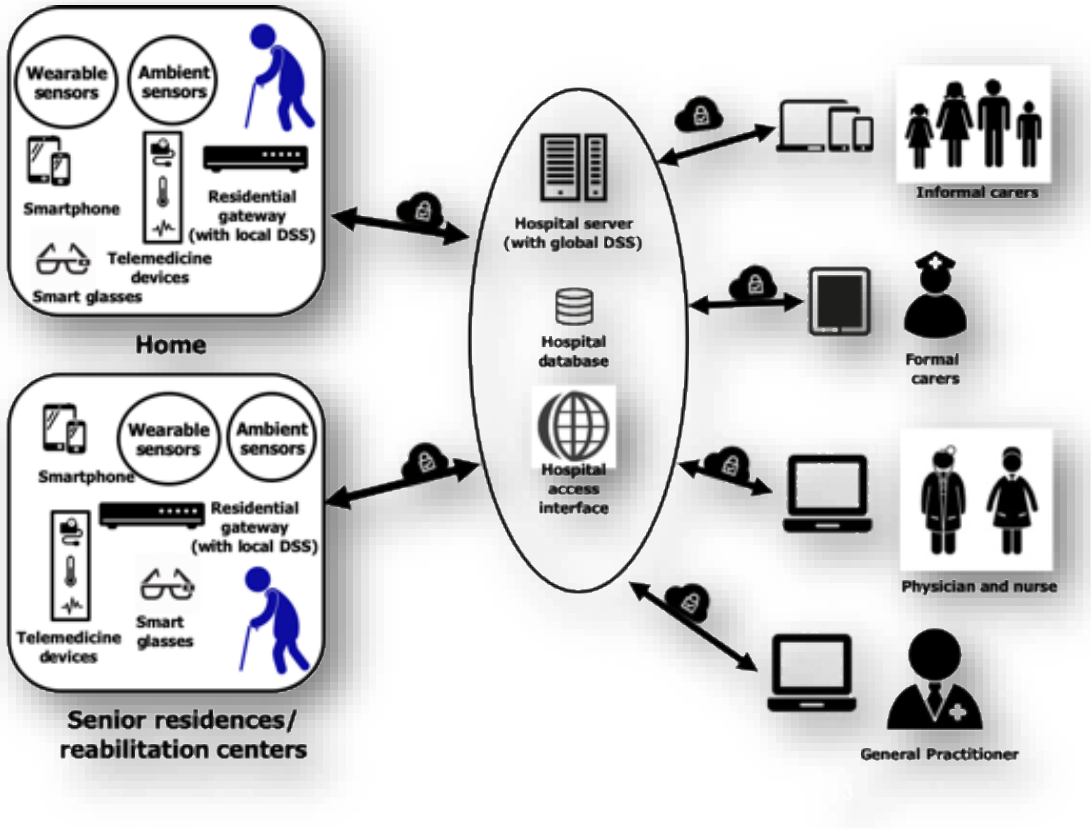
Elder needs will double in the next 20 years



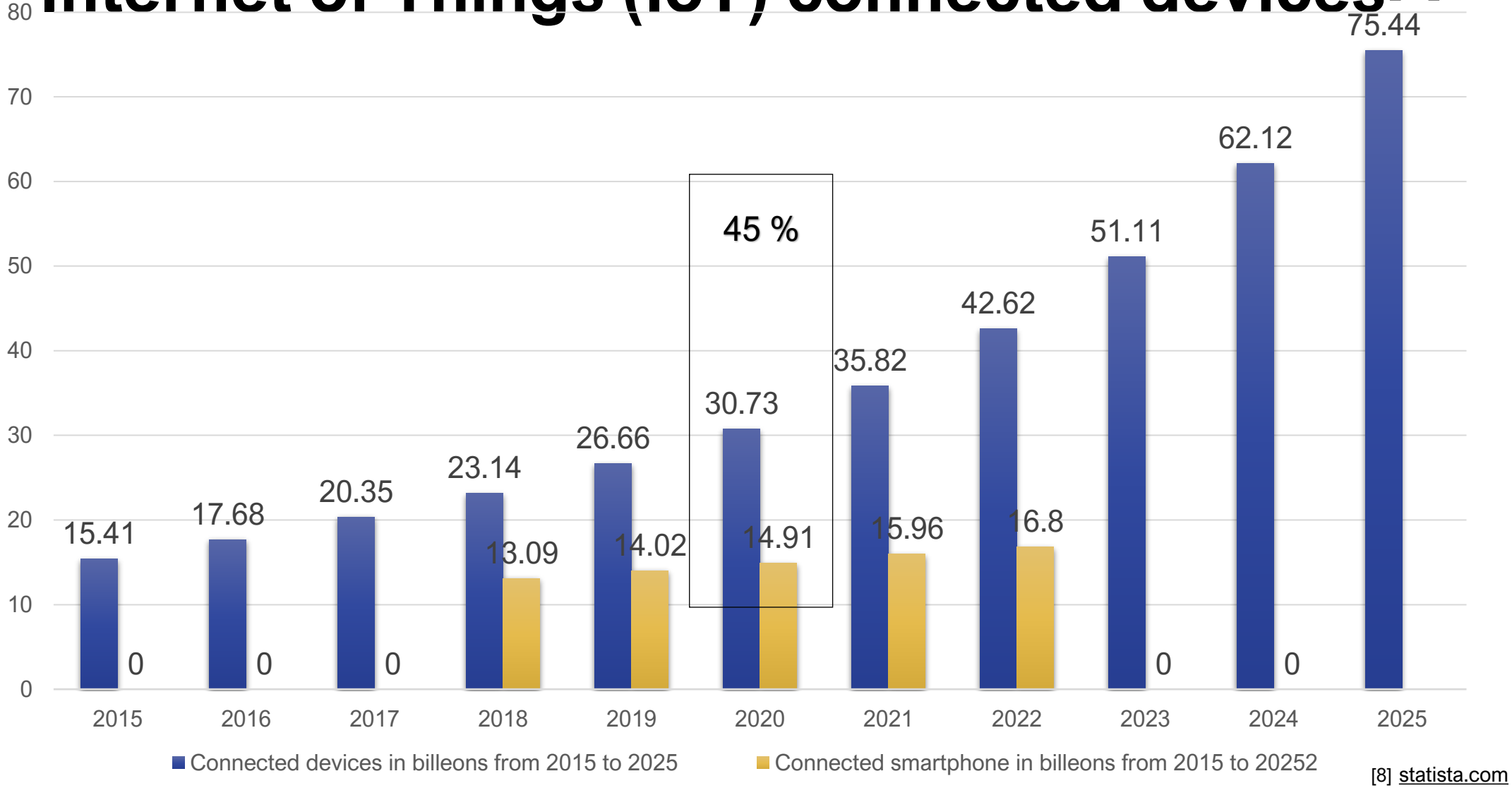
Dedicated health-care structures

[7] statista.com

Technology can Help

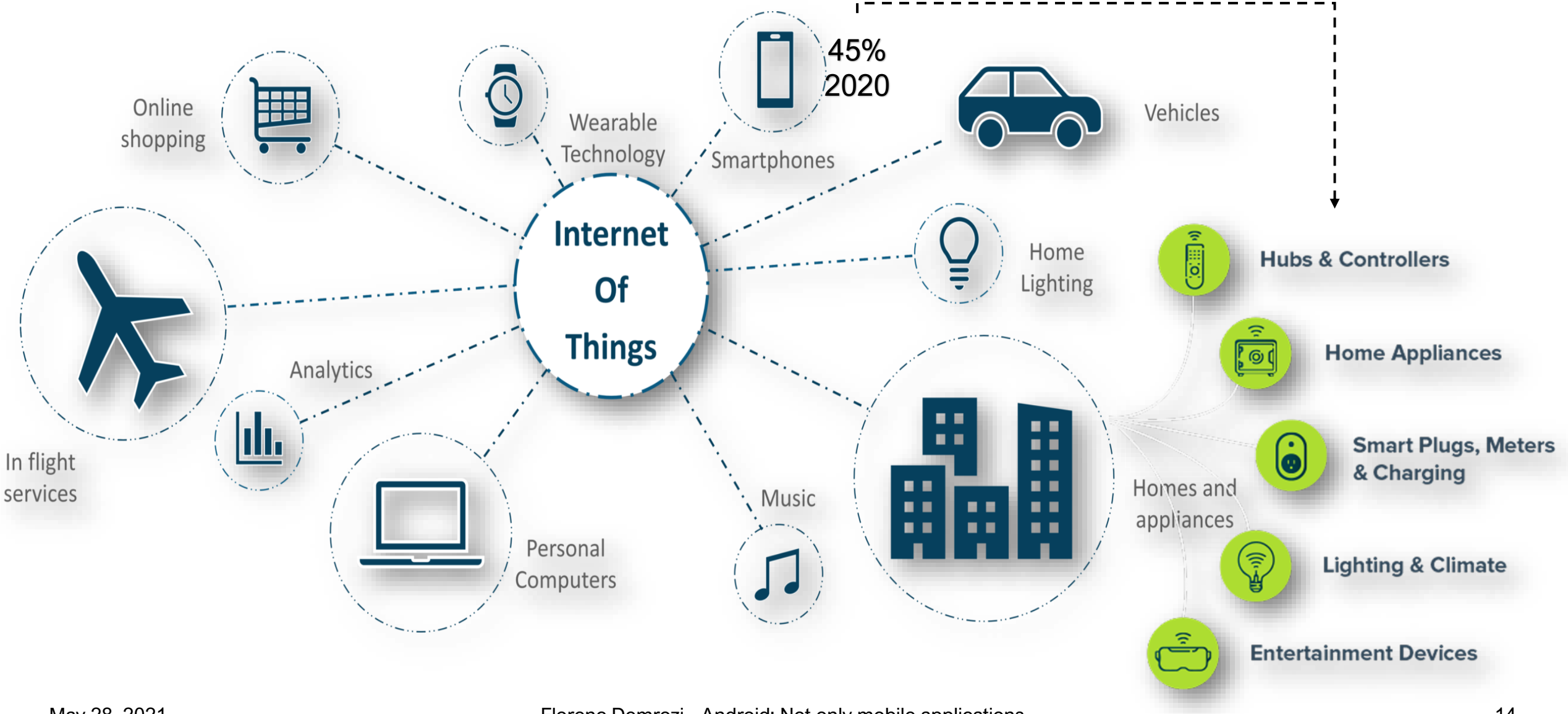


Internet of Things (IoT) connected devices[8]



[8] statista.com

IoT Devices

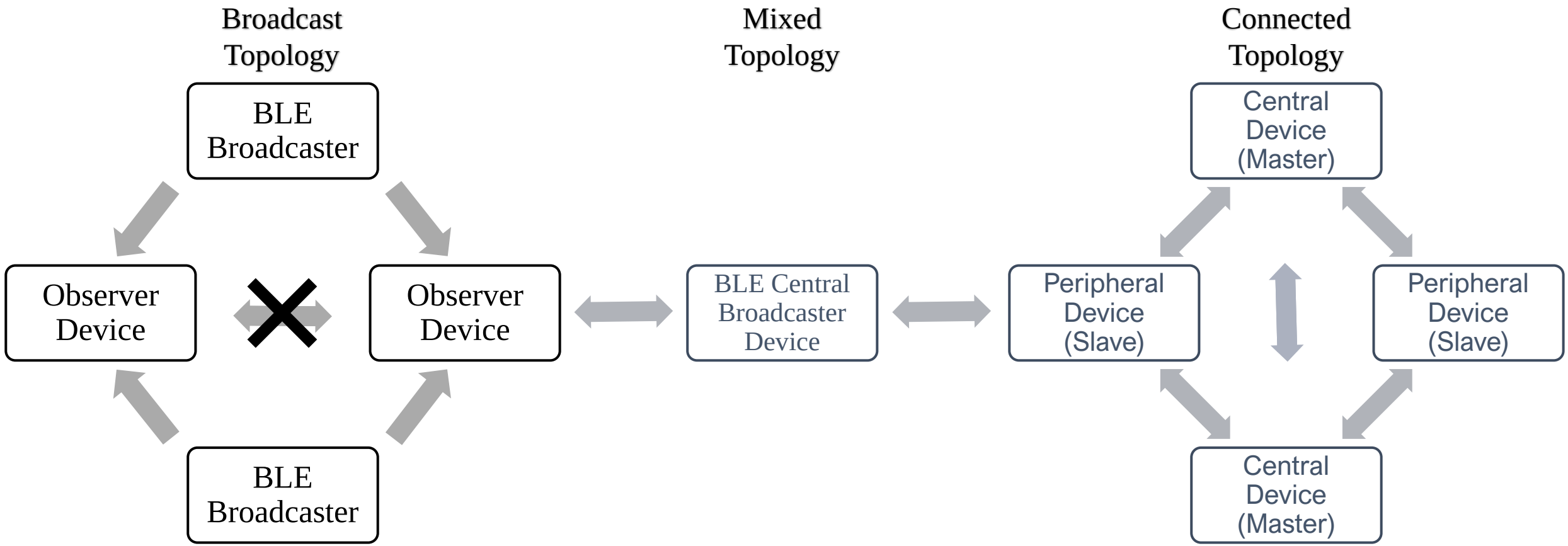

Internet Of Things
Online shopping
Wearable Technology
Smartphones
45% 2020
Vehicles
Home Lighting
Hubs & Controllers
Home Appliances
Smart Plugs, Meters & Charging
Lighting & Climate
Entertainment Devices
Homes and appliances
In flight services
Analytics
Personal Computers
Music

IoT communication protocols[9]

IoT Networking Technologies	Bluetooth	WiFi	ZigBee	MQTT IoT	IoT Data Protocols
	CoAP	DDS	NFC	GSM/3G/4G	
	AMQT	LoRaWAN	RFID	Z-Wave	
	Sigfox	Thread	EnOcean		

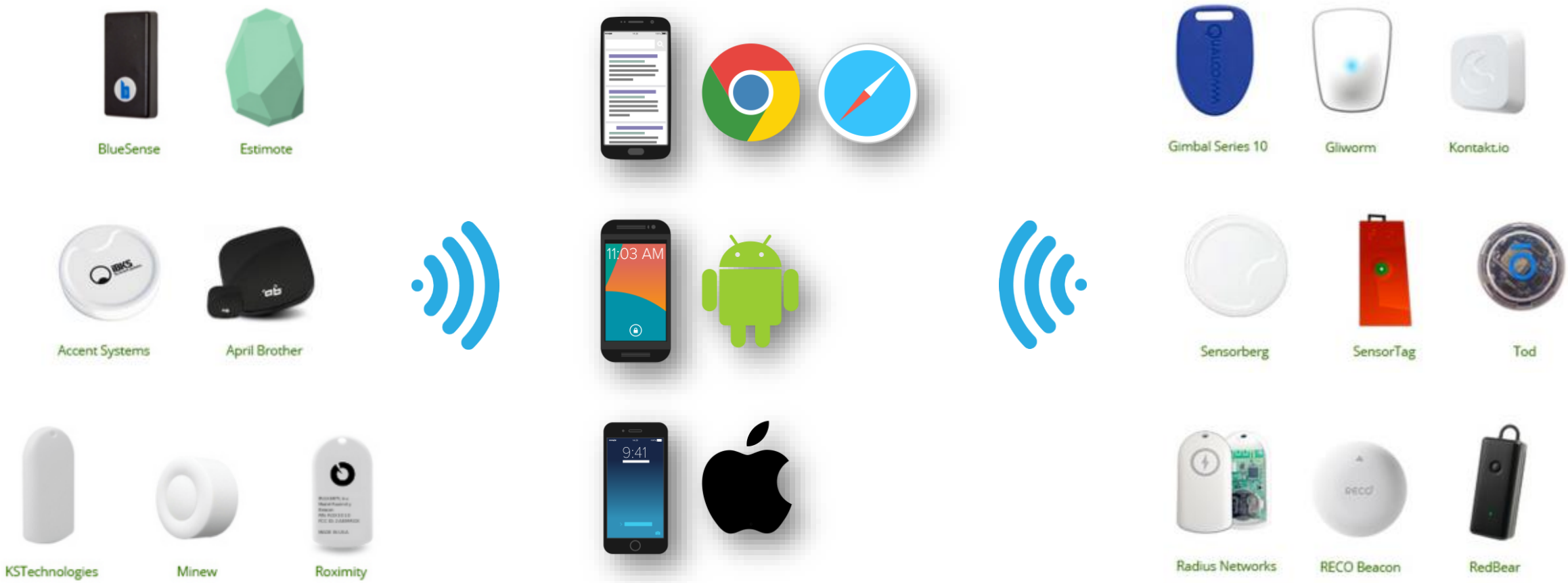
[9] ubuntupit.it

Bluetooth Low Energy (BLE)[10]



[10] oreilly.com

BLE Beacons


BlueSense
Estimote
Gimbal Series 10
Gliworm
Kontakt.io
Accent Systems
April Brother
11:03 AM
Sensorberg
SensorTag
Tod
KSTechnologies
Minew
Roximity
9:41
Radius Networks
RECO Beacon
RedBear

Titan Beacons example

Secret ad beacon network uncovered, shut down in New York City

The discovery of secret beacons installed in New York City pay phone booths opens up some questions.

For almost a year, a company called Titan has operated a network of advertising beacons, devices that are capable of identifying nearby smartphones and which

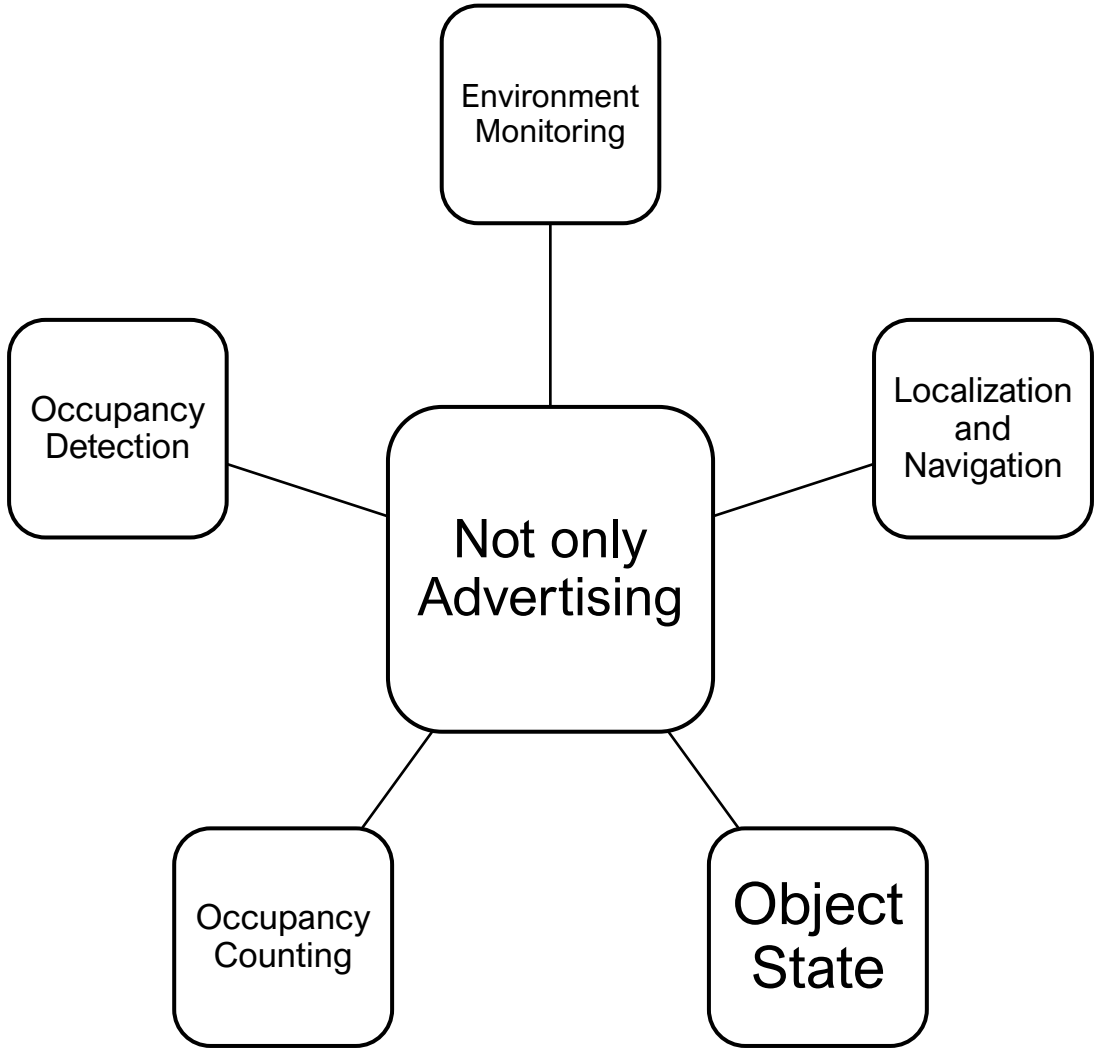
The iBeacon Mile Amsterdam

A route of up to 2km with beacons lining it.
This will represent a testing area for those who want to test applications such as:

- Public way finding
- Tourist routes
- iBeacon signing
- Hyper local points of interest
- Augmenting existing apps with additional proximity data

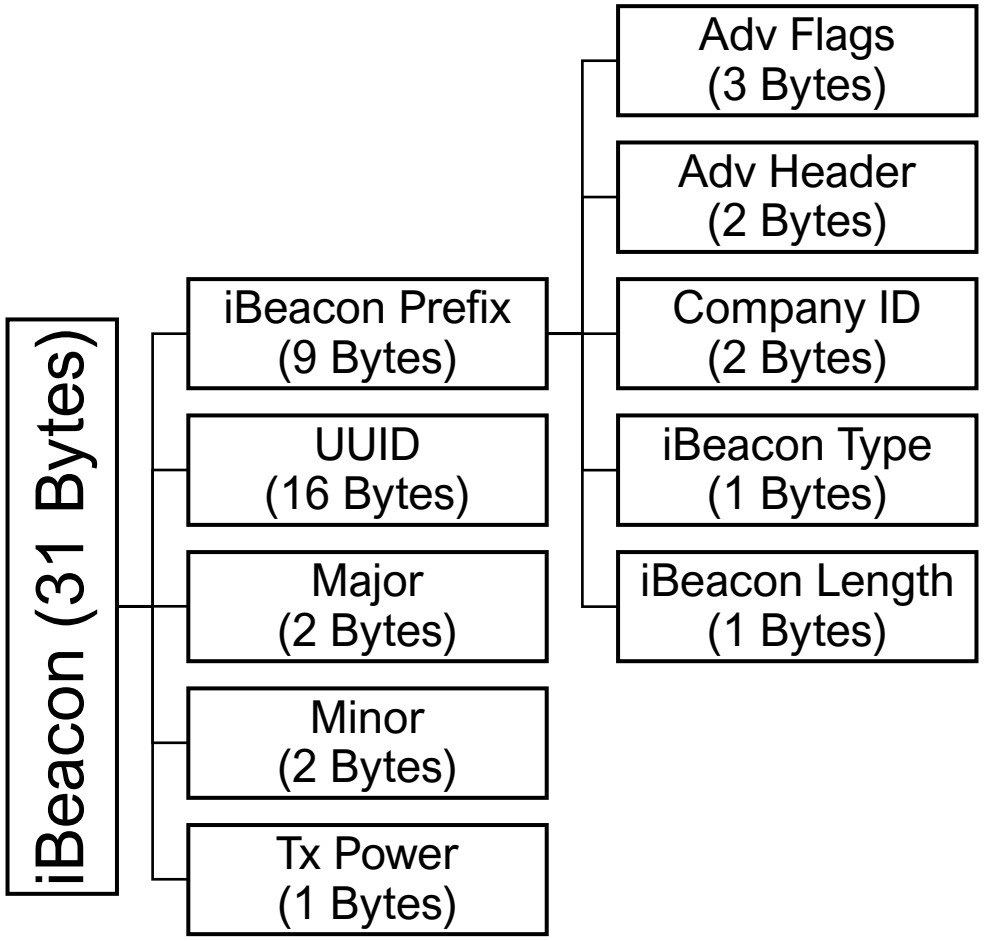
Beacons Applications

Applications	
	Activate Specific Mobile Applications
	Localization (Beacon Mile, Amsterdam)
	Navigation (Beacon Mile, Amsterdam)
	Proximity Marketing
	Automatic Check In
	Contactless Payment
	Social Distancing (COVID +19)



iBeacon

iBeacon
- Apple
- 2013

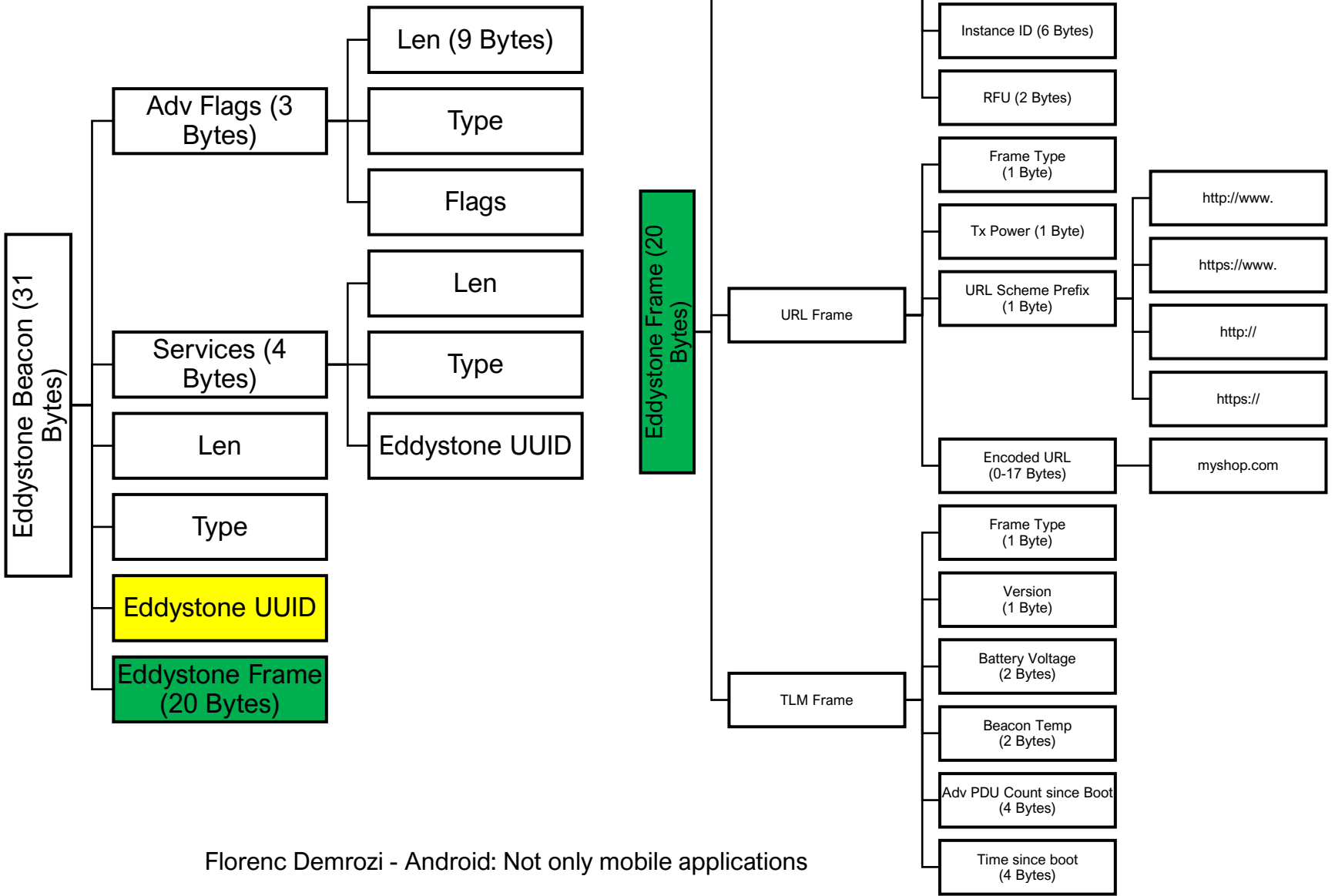


Eddystone

Eddystone

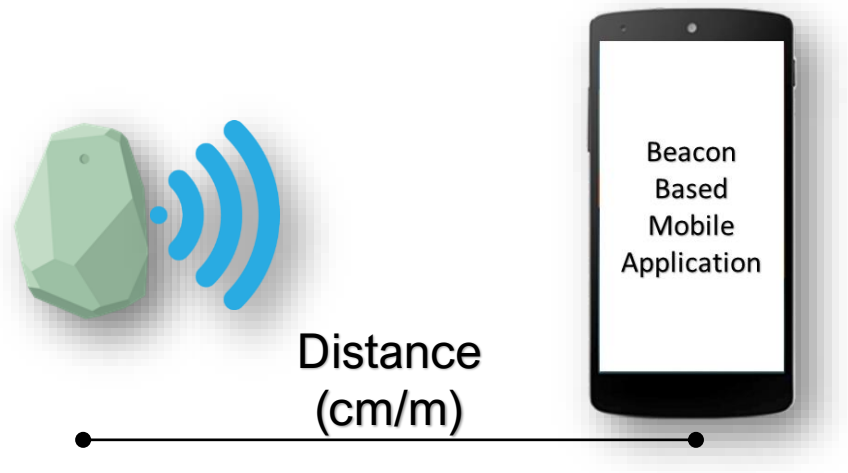
Google

2015



BLE but still Radio Waves

Radio waves are a type of electromagnetic radiation with wavelengths in the electromagnetic spectrum longer than infrared light. Radio waves have frequencies as high as 300 gigahertz (GHz) to as low as 30 hertz (Hz). At 300 GHz, the corresponding wavelength is 1 mm, and at 30 Hz is 10,000 km. Like all other electromagnetic waves, radio waves travel at the speed of light in vacuum.



Received Signal Strength Indicator (RSSI)

- is a measurement of the **power** present in a *received radio signal after antenna and cable loss*

Distance function

- d = 10 ^ (($T_x Power - T_{RSSI}$) / 10N)

RSSI threshold Value

- $T_{RSSI} = -(10*N) * log(d) + T_x Power$

Hight Frequency

- Higher Accuracy

Low Frequency

- Lower Accuracy

Our Works

Costa K, Demrozi F, Tramarin F, and Pravadelli G. A graph-based approach for mobile localization exploiting real and virtual landmarks. In 2018 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC) 2018 Oct 8 (pp. 249-254). IEEE.

Bragoi V, Demrozi F, Tramarin F, and Pravadelli G. An indoor localization system to detect areas causing the freezing of gait in Parkinsonians. In 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE) 2019 Mar 25 (pp. 952-955). IEEE.

Serlonghi N, Demrozi F, **Turetta C**, Pravadelli C, and Pravadelli G. Exploiting BLE smart tags for virtual coaching. In 2021 IEEE World Forum on Internet of Things, New Orleans, USA.

Jereghi M, Demrozi F, and Pravadelli G. Towards the Automatization of the Data Annotation Phase in Human Activity Recognition Based on Wearable Devices and BLE Beacons. In 2021 IEEE INERTIAL, Hawaii, USA

Chiarani F, Demrozi F, and Pravadelli G, A low-cost BLE-based distance estimation, occupancy detection and counting system. In 2021 ACM/IEEE Design, Automation & Test in Europe Conference & Exhibition (DATE). Grenoble, France

Chiarani F, **Turetta C**, Demrozi F, Kindt H. P, and Pravadelli G, Estimating indoor occupancy through low-cost BLE-based devices, IEEE Sensors Journal (2021)

Android Operating System

What is an Operating System?

An operating system, or "OS," is software that communicates with the hardware and allows our programs to run

Common desktop operating systems include Windows, OS X, and Linux

Common mobile OS include Android, iOS, and Windows Phone

What is Android?

Android is Linux based operating system designed primarily for mobile devices such as smartphones and tablets

Android was first developed as a ADVANCE OPERATING SYSTEM for digital cameras

There are more than 3.000.000 apps in android play store

Source code is called Android Open-Source Project (AOSP)

Launched in 2007 for the first time (Android 1.0)

Programming languages: Java, Kotlin, XML,C++

Current Android version is Android 12

Development tool: Android Studio

Android OS and its origin

Monolithic kernel (based on linux kernel)

Shell (written in JAVA)

Android was founded in Palo Alto, California in October 2003 by **Andy Rubin**, **Rich Miner**, **Nick Sears** and **Chris White** who work at "Google" to develop.

Android was purchased by the in AUGUST, 2005 for 50 million $

Now, android covers 74% of the mobile OS market.

HTC Dream was the first android device launched in September 2008

Creator's of android extracted the kernel from Linux OS 2.6 and rewrite the shell part using java

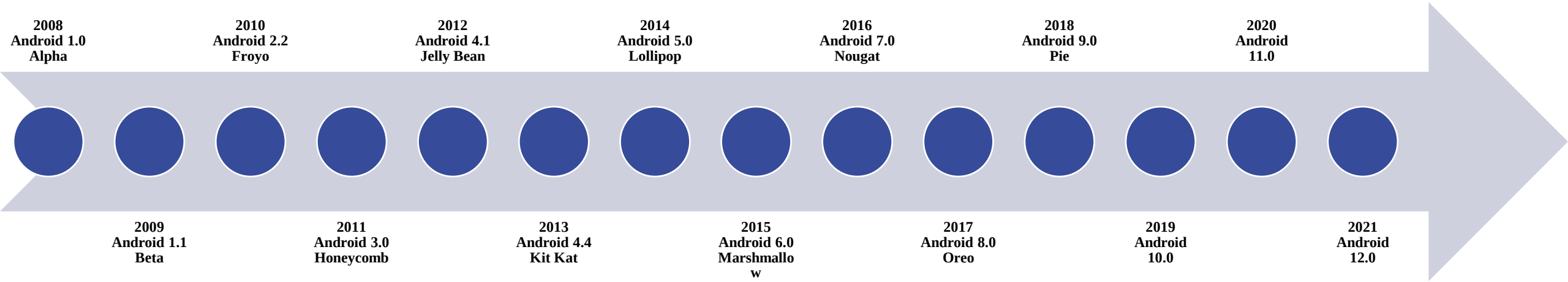
Open Handset Alliance (OHA)

OHA is a consortium of several companies who aimed to develop open standard for mobile device

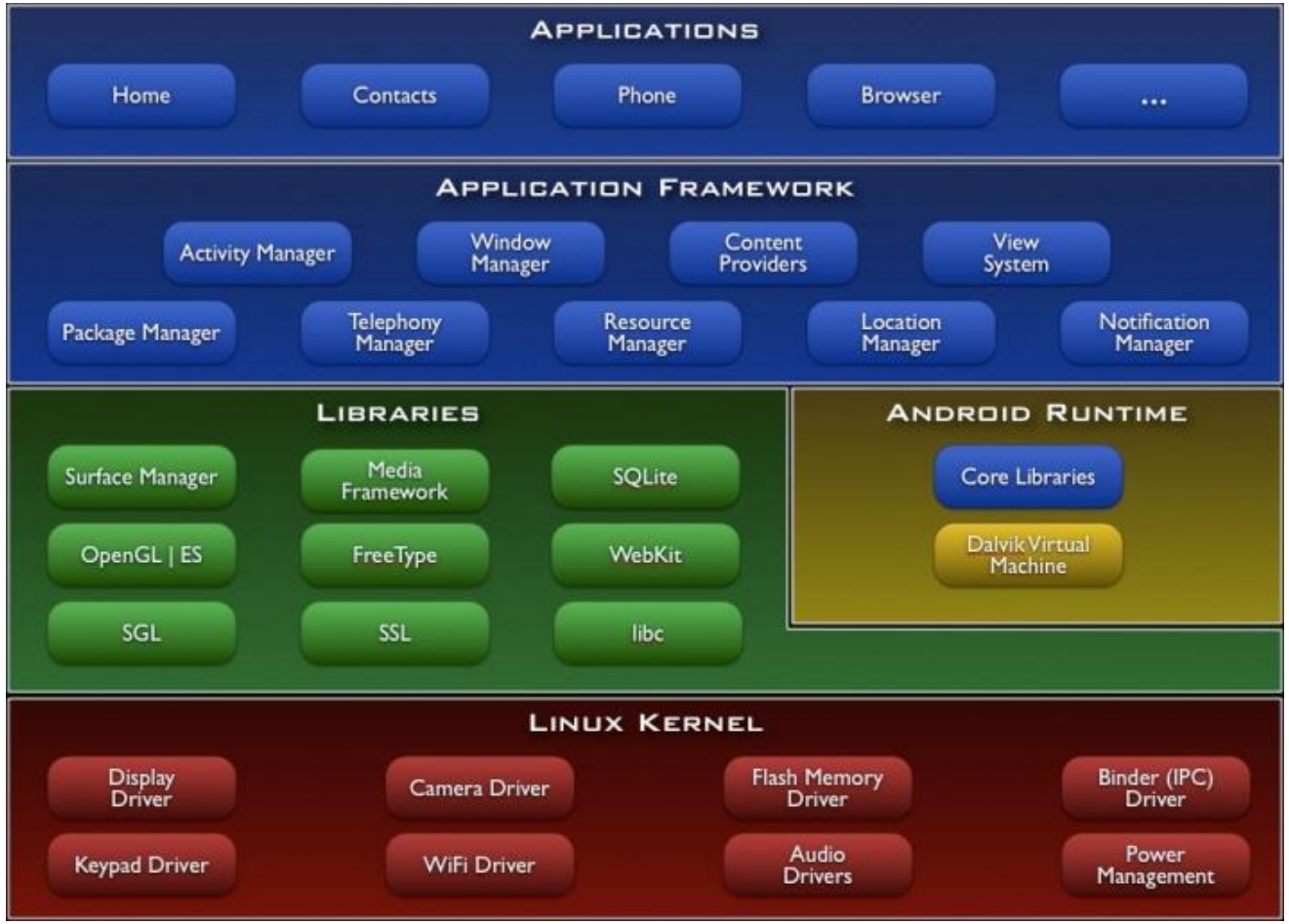
OHA includes 84 firms to develop open standard for mobile devices

HTC, Sony, Dell, Intel, Motorola, QUALCOMM, Google , Samsung Electronics, LG Electronics, T-Mobile, Nvidias

Android Timeline



Android Architecture



Each layer of the stack, and the corresponding elements within each layer, and they provide the optimal application development and execution environment for mobile devices.

Linux Kernel

The Linux Kernel provides a level of abstraction between the device hardware and the upper layers of the Android software stack

Based on Linux version 2.6, the kernel provides

- preemptive multitasking,
- low-level core system services such as: memory, process and power management

besides, provides a network stack and device drivers for hardware management of devices such display, Wi-Fi and audio

Android RunTime

Consists of Dalvik Virtual machine (now replaced by ANT) and Core Java Libraries

Android RunTime (ART) is an application runtime environment used by the Android operating system

ART performs the translation of the application's bytecode into native instructions that are later executed by the device's runtime environment

Libraries

All these libraries are written in java language (C++)

Libc- c standard library

SSL- Secure Socket Layer

Surface Manager- Responsible for composing different drawing surfaces onto the screen

Open GL and GSL are graphic libraries

Open GL/ES- 3D image engine

SGL- 2D image engine

Media Framework- core of the android multi media like MP3, MPEG4

Free Type- To render the fronts

Web kit- Open source browser engine that help to work well on small screen

QLite-Embedded Database

Application Framework

The Application Framework is a set of services that collectively form the environment in which Android applications run and are managed

This framework implements the concept that Android applications are constructed from reusable, interchangeable and replaceable components

Applications

Located at the top of the Android software stack are the applications.

These comprise both the native applications provided with the particular Android implementation (for example web browser and email applications)

and the third party applications installed by the user after purchasing the device.

Android Application Framework key entities

ACTIVITY

```
public class ClassName
extends
Activity {
}
```

SERVICE

```
public class ClassName
extends
Service {
}
```

CONTENT PROVIDER

```
public class ClassName
extends
BroadcastReceiver {
}
```

BROADCAST RECEIVER

```
public class ClassName
extends
ContentProvider {
}
```

Android Activity (1)

Activity Service

- Activity is nothing but a java class in Android which has some pre-defined functions which are triggered at different **app states**, which we can override to perform anything we want

Active State

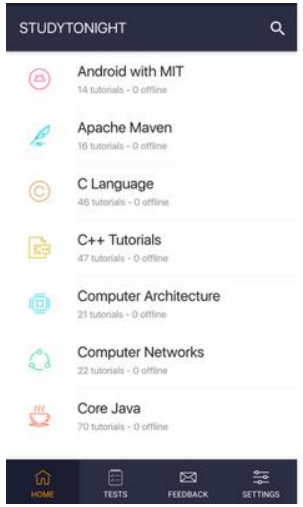
- it means it is active and running,
- it is visible and the user is able to interact with it,
- highest priority and never killed.

Paused State

- the user can still see the activity in the background,
- the user cannot interact with the activity,
- usually is not killed, may do so in an extreme case.

Stopped State

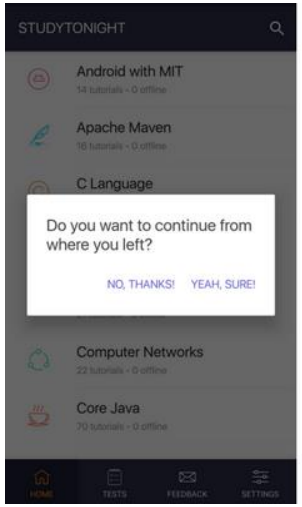
- new activity is started on top of the current one or when a user hits the Home key,
- is invisible, but it is not destroyed,
- may be killed.



Activity State:
Created, Started
or Resumed

Process state is Foreground.

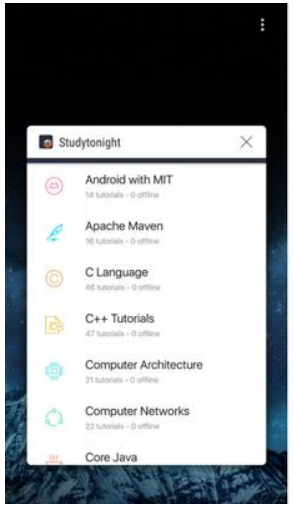
And the likelihood of killing the app is Least.



Activity State:
Paused

Process state is Background(lost focus).

And the likelihood of killing the app is More.

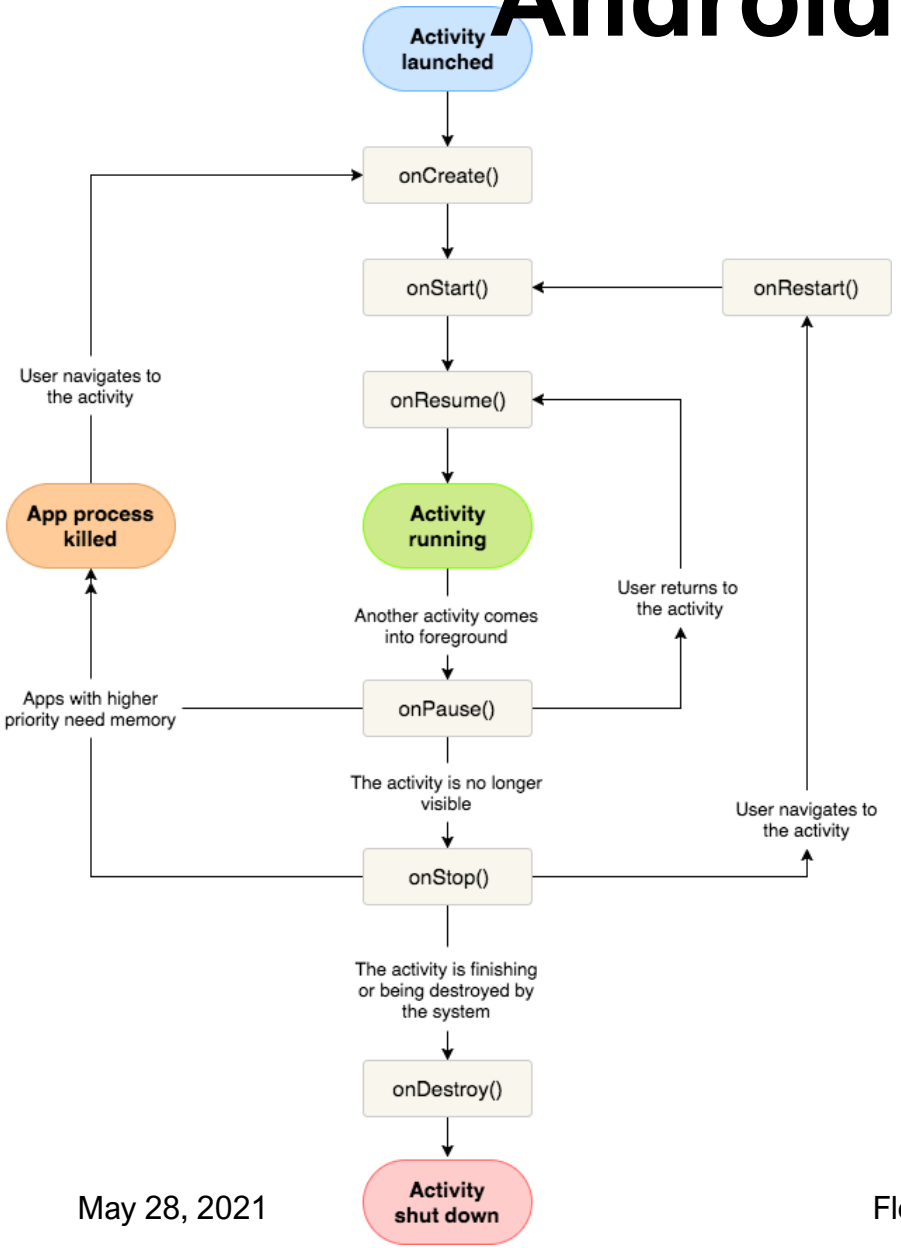


Activity State:
Stopped

Process state is Background(not visible).

And the likelihood of killing the app is Most.

Android Activity (2): Life Cycle



Method	What does it do?
onCreate()	Whenever an Activity starts running, the first method to get executed is onCreate(). This method is executed only once during the lifetime. If we have any instance variables in the Activity, the initialization of those variables can be done in this method. After onCreate() method, the onStart() method is executed.
onStart()	During the execution of onStart() method, the Activity is not yet rendered on screen but is about to become visible to the user. In this method, we can perform any operation related to UI components.
onResume()	When the Activity finally gets rendered on the screen, onResume() method is invoked. At this point, the Activity is in the active state and is interacting with the user.
onPause()	If the activity loses its focus and is only partially visible to the user, it enters the paused state. During this transition, the onPause() method is invoked. In the onPause() method, we may commit database transactions or perform light-weight processing before the Activity goes to the background.
onStop()	From the active state, if we hit the Home key, the Activity goes to the background and the Home Screen of the device is made visible. During this event, the Activity enters the stopped state. Both onPause() and onStop() methods are executed.
onDestroy()	When an activity is destroyed by a user or Android system, onDestroy() function is called.

Android Service (1)

Services

- Activity is nothing but a java class in Android which extends the **Service** class.
- Background process,
- No user interface.

Example

- an app in which an activity starts a service running on user interaction, with the service perhaps uploading data to a web resource. The user can continue to interact with the activity while the service runs because it executes in the background.

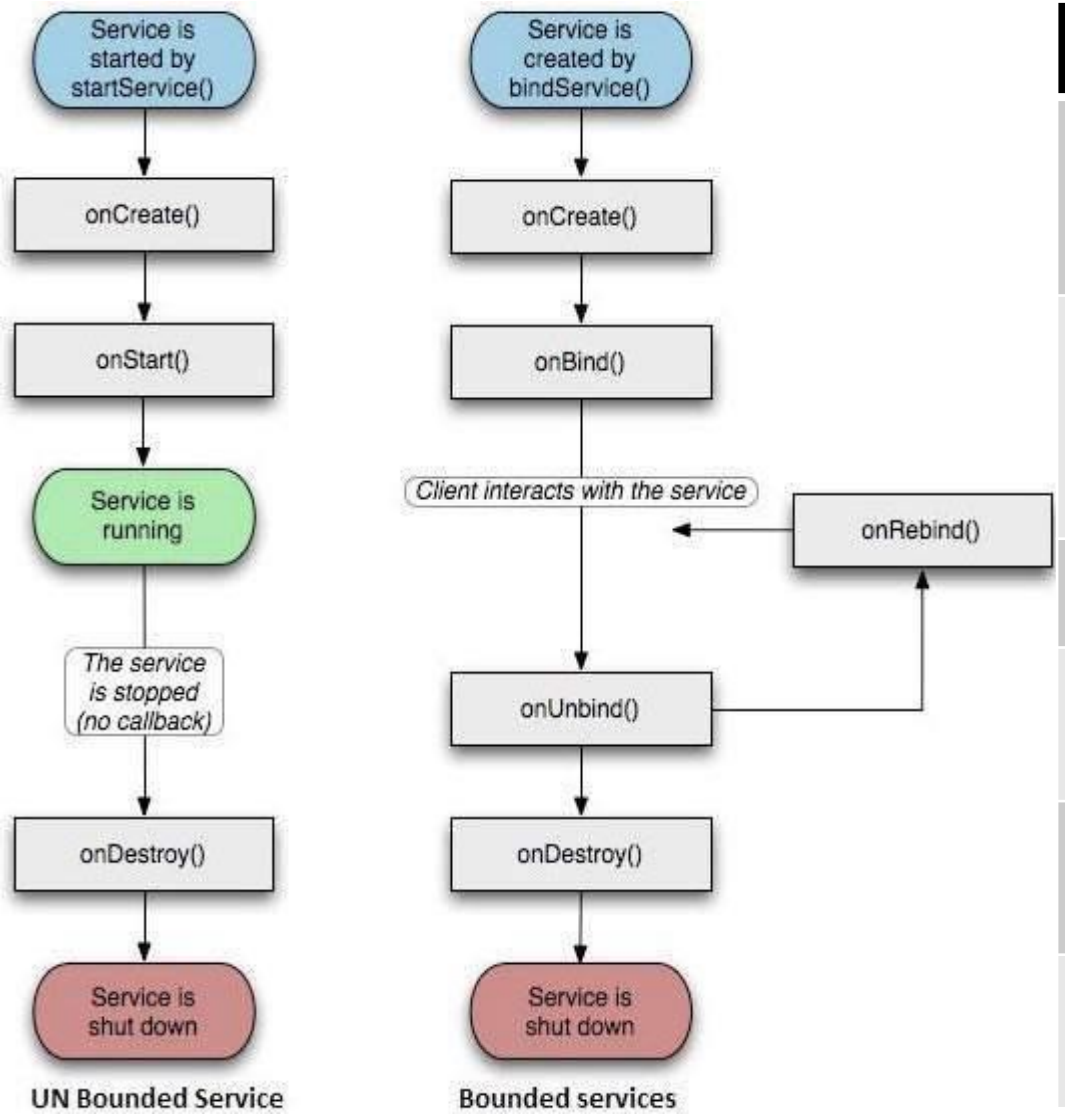
Started

- A service is started when an application component, such as an activity, starts it by calling startService().
- Once started, a service can run in the background indefinitely, even if the component that started it is destroyed.

Bound

- A service is bound when an application component binds to it by calling bindService().
- A bound service offers a client-server interface that allows components to interact with the service, send requests, get results, and even do so across processes with interprocess communication (IPC).

Android Service (2)



Method	What does it do?
startService()	The system calls this method when another component, such as an activity, requests that the service be started, by calling startService(). If you implement this method, it is your responsibility to stop the service when its work is done, by calling stopSelf() or stopService() methods.
onBind()	The system calls this method when another component wants to bind with the service by calling bindService(). If you implement this method, you must provide an interface that clients use to communicate with the service, by returning an IBinder object. You must always implement this method, but if you don't want to allow binding, then you should return null.
onUnbind()	The system calls this method when all clients have disconnected from a particular interface published by the service.
onRebind()	The system calls this method when new clients have connected to the service, after it had previously been notified that all had disconnected in its onUnbind(Intent).
onCreate()	The system calls this method when the service is first created using startService() or bindService(). This call is required to perform one-time set-up.
onDestroy()	The system calls this method when the service is no longer used and is being destroyed. Your service should implement this to clean up any resources such as threads, registered listeners, receivers, etc.

Android Intent Service

An Intent is

- a messaging object that you can use to request an action from an app component,
- is basically an intention to do an action,
- a way to communicate between Android components,
- request an action from a component, and by different components,
- a message that Android listens for and then react accordingly by identifying and invoking the app's appropriate component,
- within that same app or some other app as well,
- multiple apps can respond to the message then Android provides the user with a list of those apps from which a choice can be made.

Uses of Intent in Android

- start an Activity,
- start a Service,
- deliver a Broadcast.

In Android, there are two types of Intents:

1. Explicit Intents,
2. Implicit Intents.

Type of Intents

Explicit Intent

1. Create an Intent object

```
Intent i = new Intent(this, MyJavaFile.class);
```

2. Start Activity

```
startActivity(i);
```

3. Pass Arguments to the new activity

```
i.putExtra("key1", "I am value1");
i.putExtra("key2", "I am value2");
startActivity(i);
```

4. Receive data into the new Activity

```
String a = getIntent().getStringExtra("key1");
```

Implicit Intent

1. Create an Intent object

```
Intent i = new Intent(Intent.ACTION_VIEW);
```

2. Provide Data

```
i.setData(Uri.parse("http://www.google.co.in"))
```

3. Start Activity

```
startActivity(i);
```

Use of Intents

To start an Activity

- You can start a new instance of an Activity by passing an Intent to startActivity(),
- The Intent describes the activity to start and carries any necessary data along.

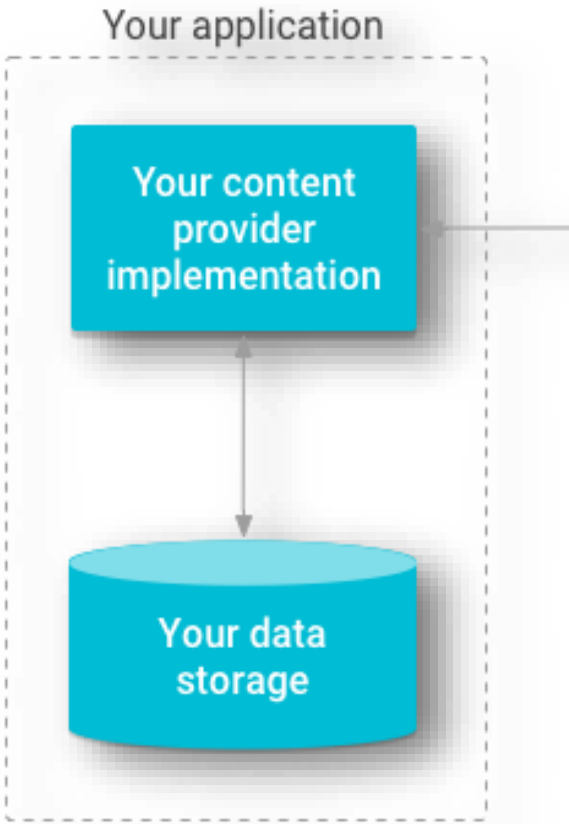
To start a Service

- A Service is a component that performs operations in the background and does not have a user interface,
- You can start a service to perform a one-time operation (such as downloading a file) by passing an Intent,
- The Intent describes which service to start and carries any necessary data.

To deliver a Broadcast

- A broadcast is a message that any app can receive,
- The system delivers various broadcasts for system events, such as when the system boots up or the device starts charging,
- You can deliver a broadcast to other apps by passing an Intent to sendBroadcast() or sendOrderedBroadcast(),

Android Content Provider



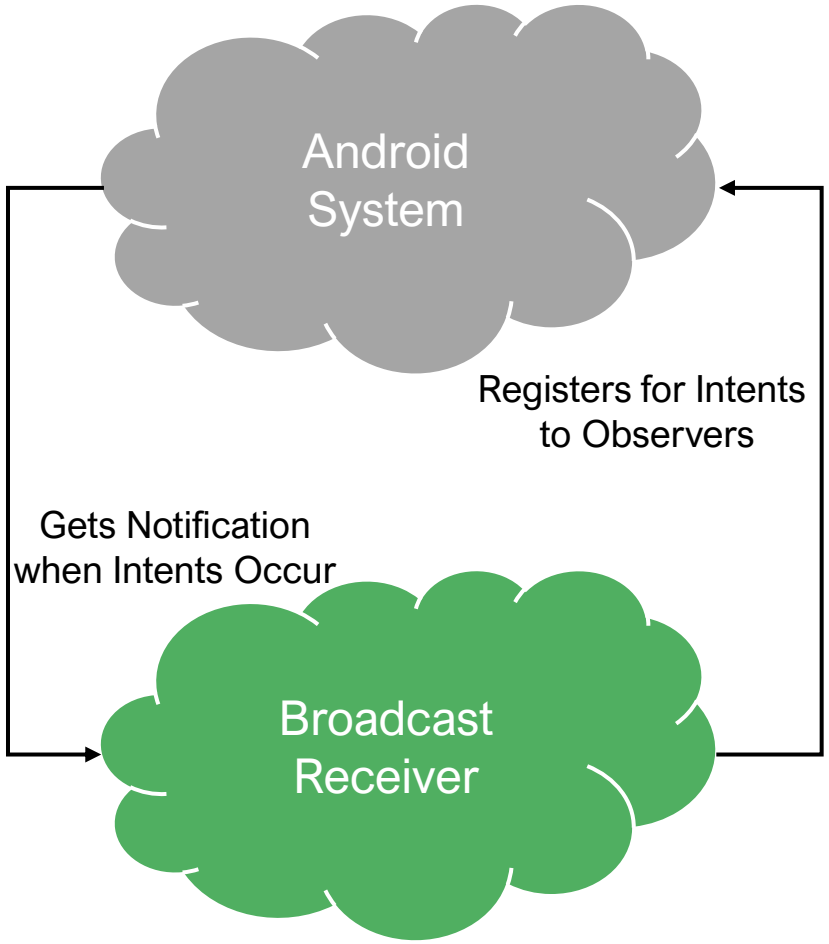
Content Provider

- is a component for managing a data set,
- can be a private or shared dataset,
- used by activity UI components to interact with data.

Tip

- the only occasions in which you need to create a content provider are:
  - when you want other apps to access your structured data,
  - when you want to copy structured data from your apps to other apps,
  - when you want to make use of the search framework.

Android Broadcast Receivers



Broadcast Receivers

- Android broadcasts system information as battery's status, screen brightness etc.
- Broadcast apps can be developed,
- Android broadcasts announcements as intents.

Example

- an app in which an activity starts a service running in background and based on the battery status the sampling frequency of BLE is changed,
- battery status, call button, bug reports etc.

Android Additional Components

S.No	Components & Description
1	**Fragments** Represents a portion of user interface in an Activity.
2	**Views** UI elements that are drawn on-screen including buttons, lists forms etc.
3	**Layouts** View hierarchies that control screen format and appearance of the views.
4	**Resources** External elements, such as strings, constants and drawable pictures.
5	**Manifest** Configuration file for the application.

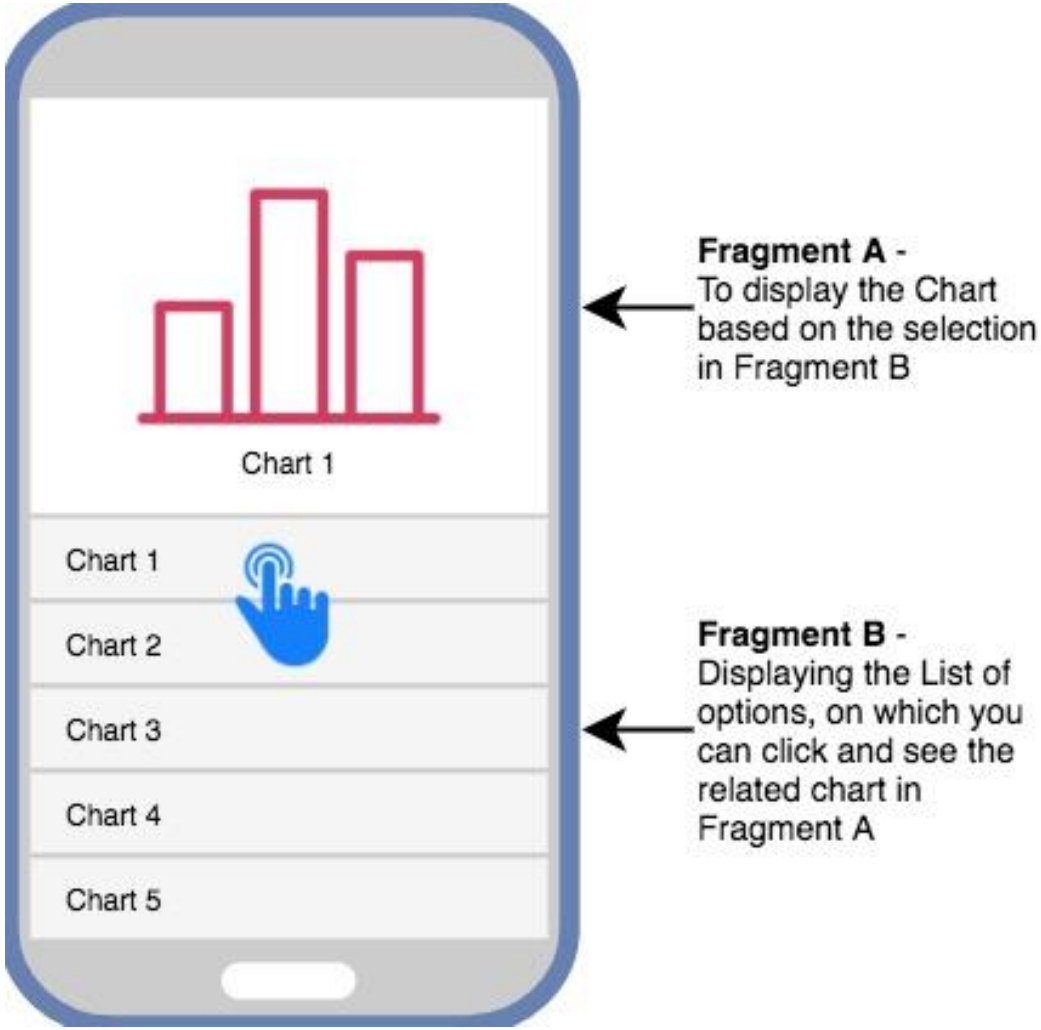
Android Fragments

Fragment characteristics

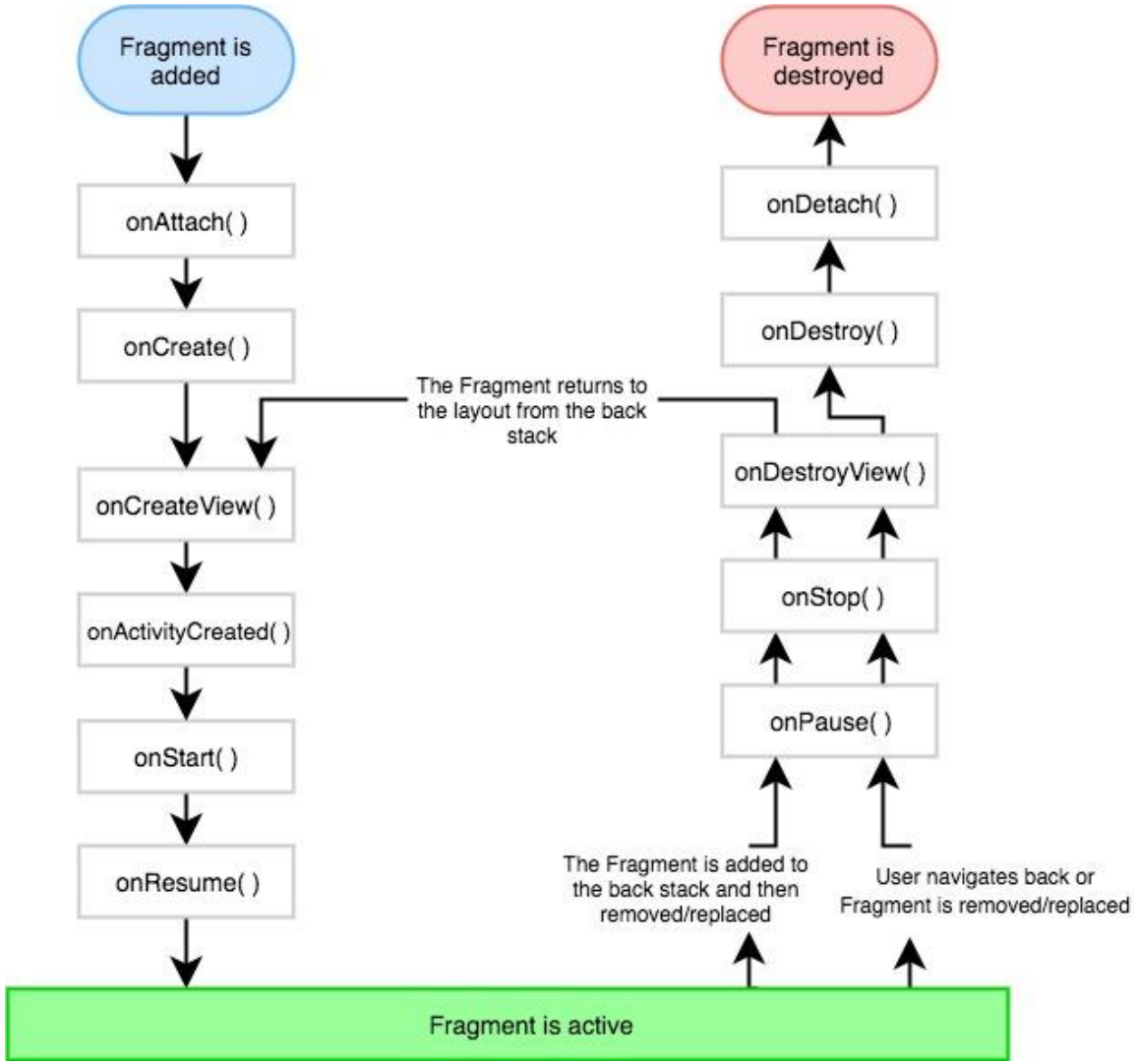
- a component to define an independent modular UI component,
- functions independently, but is linked to the Activity,
- when an activity is destroyed, the fragment also gets destroyed,
- own lifecycle events,
- an Activity can have many Fragments (not usual),
- re-usable, can be included in many Activities.

Uses

- Modularity,
- Reusability,
- Adoptability.



Android Fragments



Method	What does it do?
onAttach(Activity)	It is called once, when the fragment is attached to the activity.
onCreate(Bundle)	The system calls this method when a fragment is created.
onCreateView()	This method usually returns a View component but if the fragment doesn't have a UI, then you can return a null.
onActivityCreated()	This method is called when the host activity is created. By this time, we can even access the fragment's view using the findViewById() method.
onStart()	This method is called when the fragment becomes visible on the device's screen.
onResume()	----------
onPause()	This is called when a fragemnt is no longer interactive and the user is about to leave the fragment. At this point, it is suggested to save any data for the existing user session, if required.
onStop()	This method is called when the fragment is no longer visible.
onDestroyView()	This is called when the fragment is to be be destroyed. Here you can do the clean up the resources before the fragment is destroyed.
onDestroy()	This is called for the final clean up of fragment's state.
onDetach()	It is called just before the fragment is detached from the host activity.

More on Android

Alert Dialoges	Animations	Audio Capture	AudioManager	Auto Complete	Best Practices	Bluetooth	Camera
Clipboard	Custom Fonts	Data Backup	Developer Tools	Emulator	Facebook Integration	Gestures	Google Maps
Image Effects	ImageSwitcher	Internal Storage	JetPlayer	JSON Parser	Linkedin Integration	Loading Spinner	Localization
Login Screen	MediaPlayer	Multitouch	Navigation	Network Connection	NFC	PHP/MySQL	Progress Circle
ProgressBar	Push Notification	RenderScript	RSS Reader	Screen Cast	SDK Manager	Sensors	Session Management
Shared Preferences	SIP Protocol	Spelling Checker	SQLite Database	Support Library	Testing	Text to Speech	TextureView
Twitter Integration	UI Design	UI Patterns	UI Testing	WebView Layout	Wi-Fi	Widgets	XML Parsers

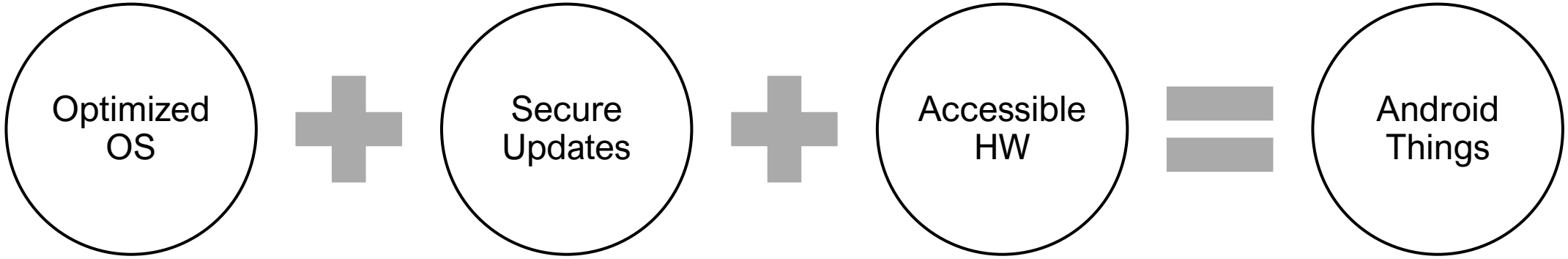
Android: Not only smartphone applications

Android Things

Connected Devices for a wide variety of costumer, retail, and industrial applications

Wear OS by Google

Android Things (1)



Android Things (2)

Android Things Operating System

- Customized version of Android OS,
- Google I/O 2015, codename “Brillo”,
- Based on Android 7.0+ (2015),
- BLE support,
- Launcher for IoT configuration,
- Single application mode,
- “light” version of Google Play Services,
- No app store- apps installed via ROM updates,
- Drivers for peripheral devices.

Android vs Android Things

- No Play Store,
- Users Cannot install on their own,
- App updates provided with OTA updates,
- No system application drawer,
- No notifications,
- UI is optional.

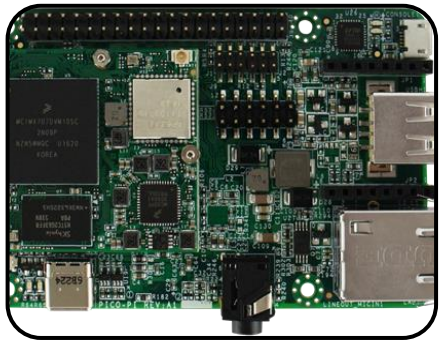
Android Things (3)

Raspberry Pi 3 Model B

- Quad-core ARMv8 Cpu at 1.2Ghz
- Wireless Lan 802.11n
- Bluetooth 4.0
- 8Gb SD card
- USB
- HDMI

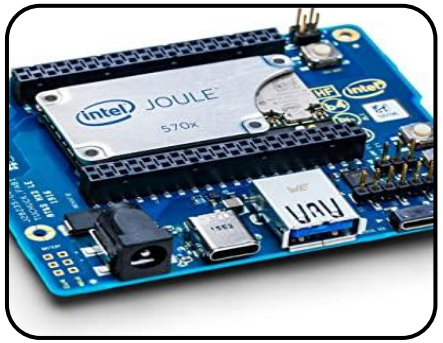
Intel Edison

- Intel Dual-core Atom at 500MHz
- 1 Gb DDR3 Ram and 4 Gb eMMC flash
- Compatible with Arduino (using an Arduino breakout Kit)
- Bluetooth and WiFi



NXP Pico i.MX6UL

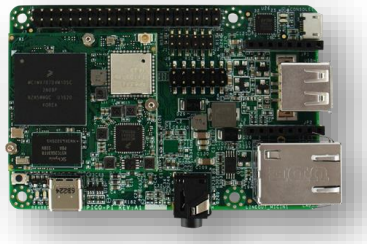
- i.MX 6UltraLite 696 MHz Arm® Cortex®-A7 core
- 4 Gb DDR3L SDRAM, 400 MHz
- 256 MB Quad SPI Flash
- MicroSD connector
- Footprint for eMMC
- Footprint for NAND Flash



Intel Joule 570x

- 64bit, 1.7GHz quad core Intel atom T5700, 2.4GHz
- 4GB LPDDR4 RAM and 16GB eMMC memory
- Intel HD graphics with 4K video capture and display
- 802.11ac Wi-Fi with MIMO and Bluetooth 4.1
- USB 3.0, MPI CSI and DSI

Android Things (4)



Android Things

- Building devices on a trusted platform, without previous knowledge of embedded system design,
- Develop using the Android SDK and Android Studio,
- Access hardware such as displays and cameras natively through the Android framework,
- Connect your apps with Google services,
- Integrate additional peripherals through the Peripheral I/O APIs (GPIO, I2C, SPI, UART, PWM),
- Use the Android Things Console to push OTA feature and security updates.

Hardware

- Android Things enables you to build apps on top of popular hardware platforms like the Raspberry Pi 3,
- Board Support Package (BSP) managed by Google, so no kernel or firmware development is required,
- Software images are built and delivered to devices through the Android Things Console.

SDK

- Android Things extends the core Android framework with additional APIs provided by the Things Support Library, which lets you integrate with new types of hardware not found on mobile devices,
- More flexible access to hardware peripherals and drivers than mobile devices,
- System apps are not present to optimize startup and storage requirements,
- Apps are launched automatically on startup to immerse your users in the app experience,
- Devices expose only one app to users, instead of multiple like with mobile devices.

Android Things Projects

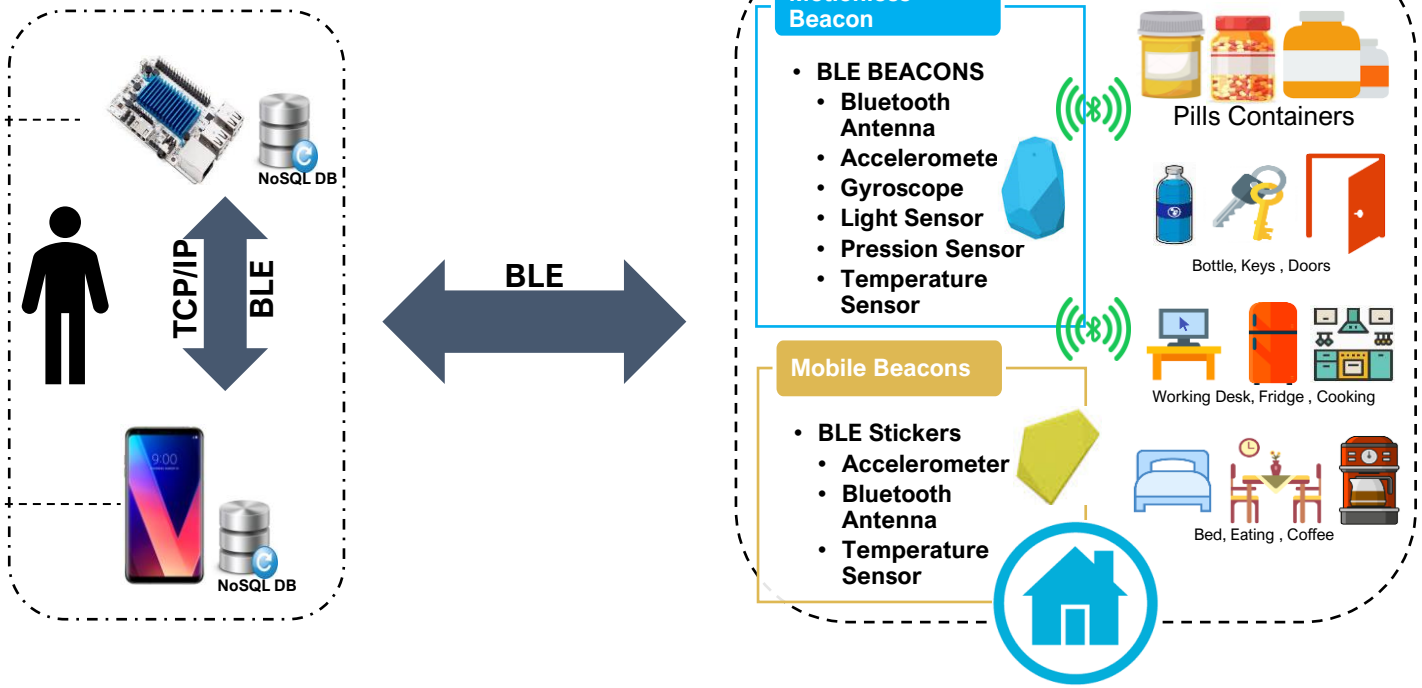
- subscription.packtpub.com

Android Things (5)

Still Missing

Android Things

Android



For Thursday

Homework

- Study the AltBeacon Library,
- Configure a second smartphone as a Beacon Simulator,
- Use the AltBeacon library to create a mobile application that reads the received Beacons emitted by the Beacon Simulator,
- Think About possible applications of beacons.

AltBeacon[11]

What Does This Library Do?

- It allows Android devices to use beacons much like iOS devices do. An app can request to get notifications when one or more beacons appear or disappear. An app can also request to get a ranging update from one or more beacons at a frequency of approximately 1Hz. It also allows Android devices to send beacon transmissions, even in the background.

What kinds of beacons does it detect?

- iBeacon,
- Eddystone,
- other beacon formats meeting the open AltBeacon standard.

What devices can detect beacons?

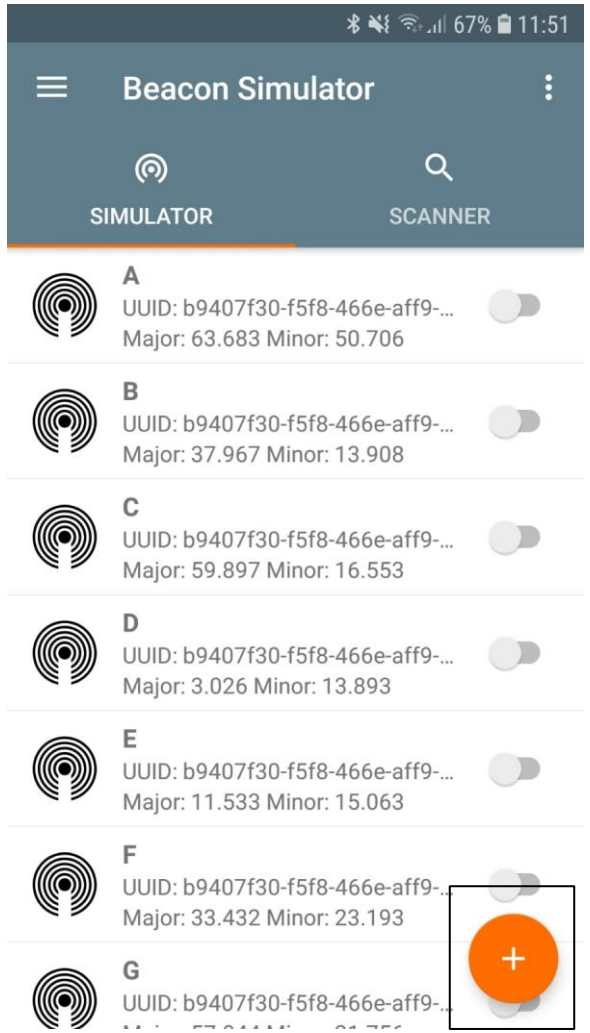
- Android 4.3+,
- Bluetooth Low Energy chipset.

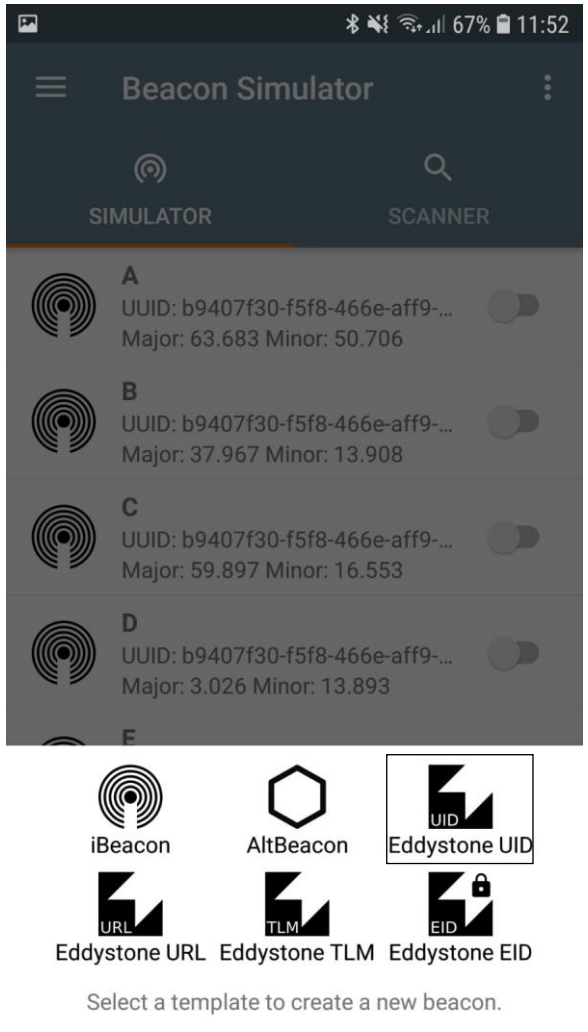
Covid Contact Tracing Beacon Support

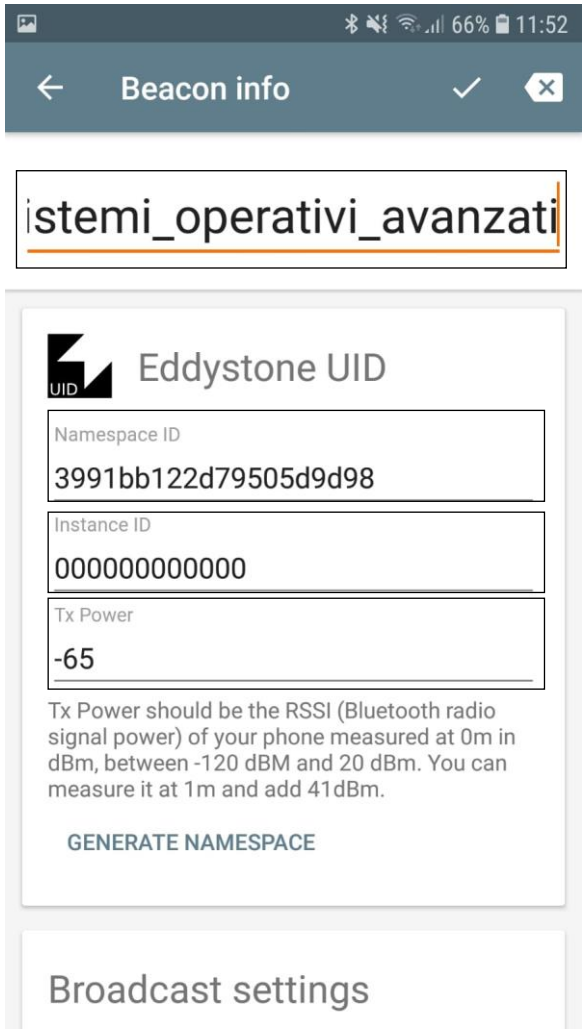
- new BeaconParser().setBeaconLayout("s:0-1=fd6f,p:0-0:-63,i:2-17,d:18-21")

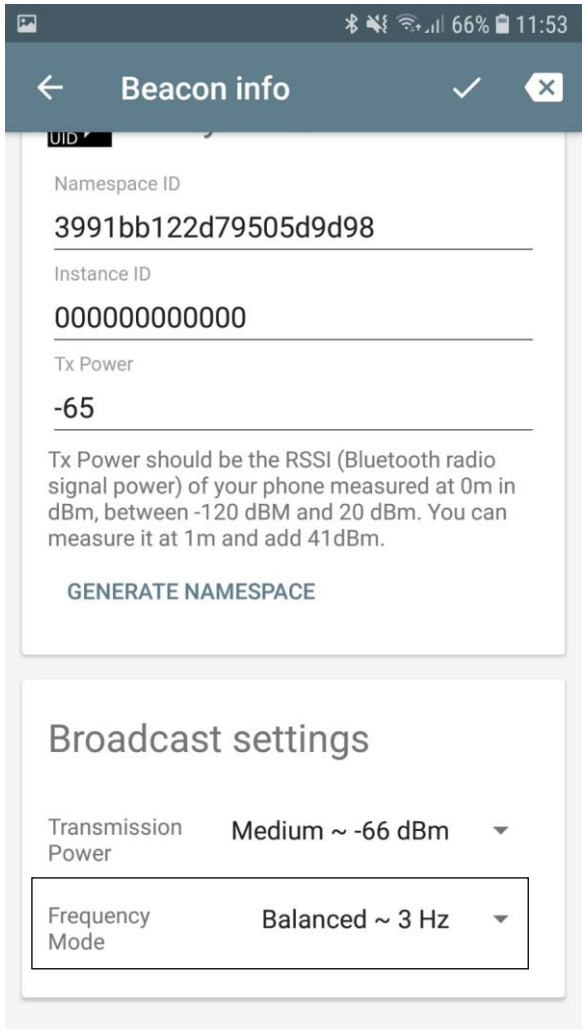
[11] AltBeacon.github.io

Beacon Simulator

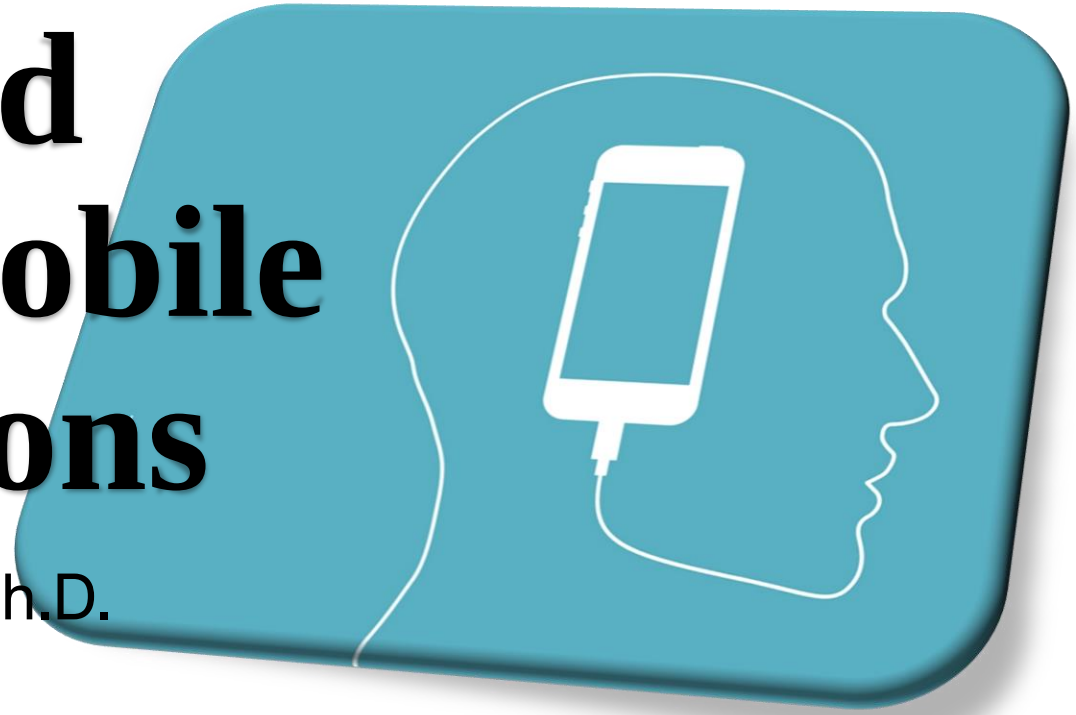








Android not only mobile applications



Florenc Demrozi Ph.D.

28/05/2021

Embedded Operating Systems

Thank you

IoT4Care

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA