

# **Mobile Computing**

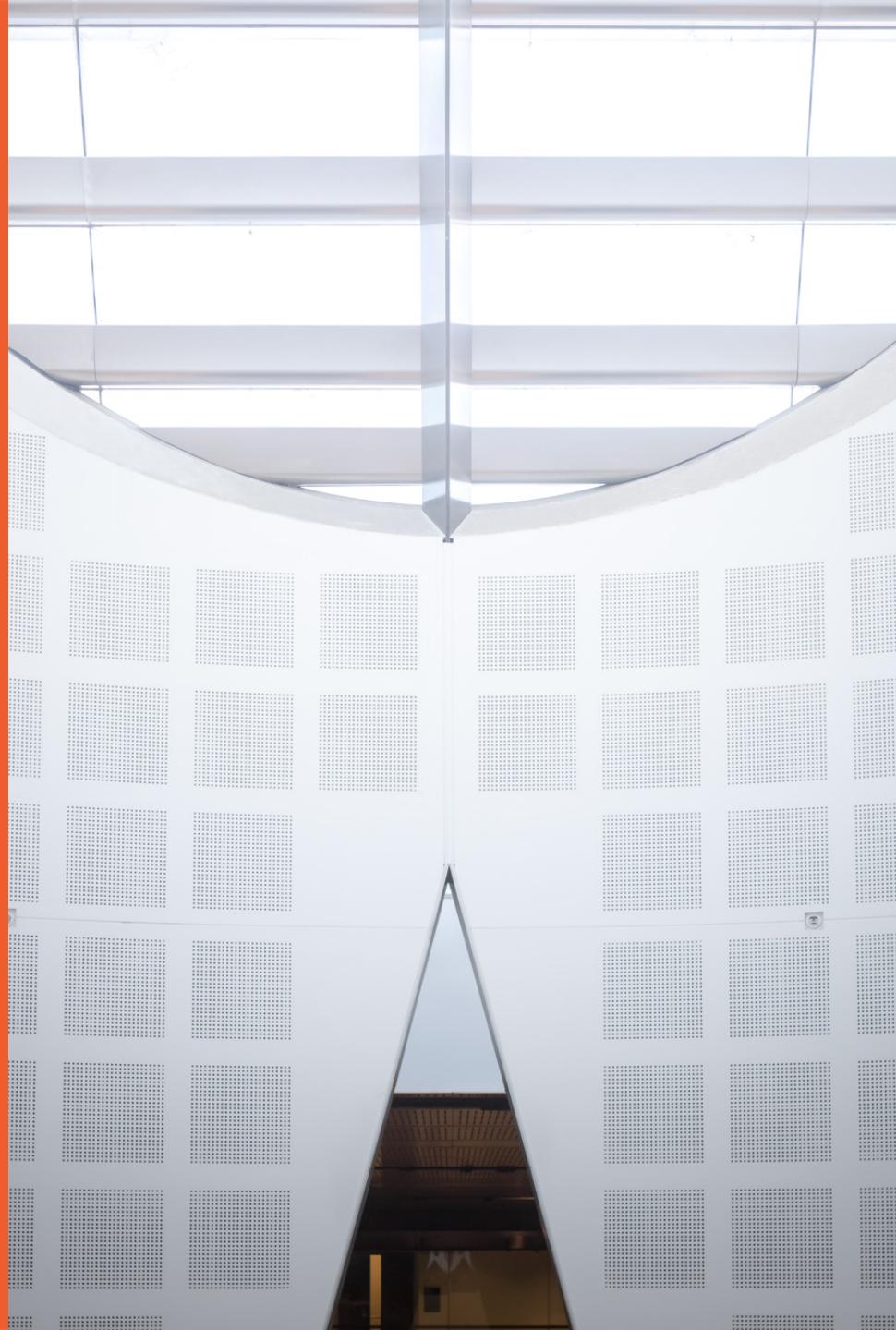
## **COMP5216**

**Week 03**  
**Semester 2, 2020**

**Dr Kanchana Thilakarathna**  
**School of Computer Science**



THE UNIVERSITY OF  
**SYDNEY**



# Announcements

- Change in Assignment 1 submission.
- 



COMP5216 Mobile Computing

2020S2

## Assignment 1 – Extending ToDoList app

**Total: 5 marks**

**Due date: 7pm 14<sup>th</sup> September 2020 (Week 04)**

**Submission:**

1. Submit all project files as one zipped file.
2. **Video of the screen capture demonstrating the actions/features that are required to be developed.**

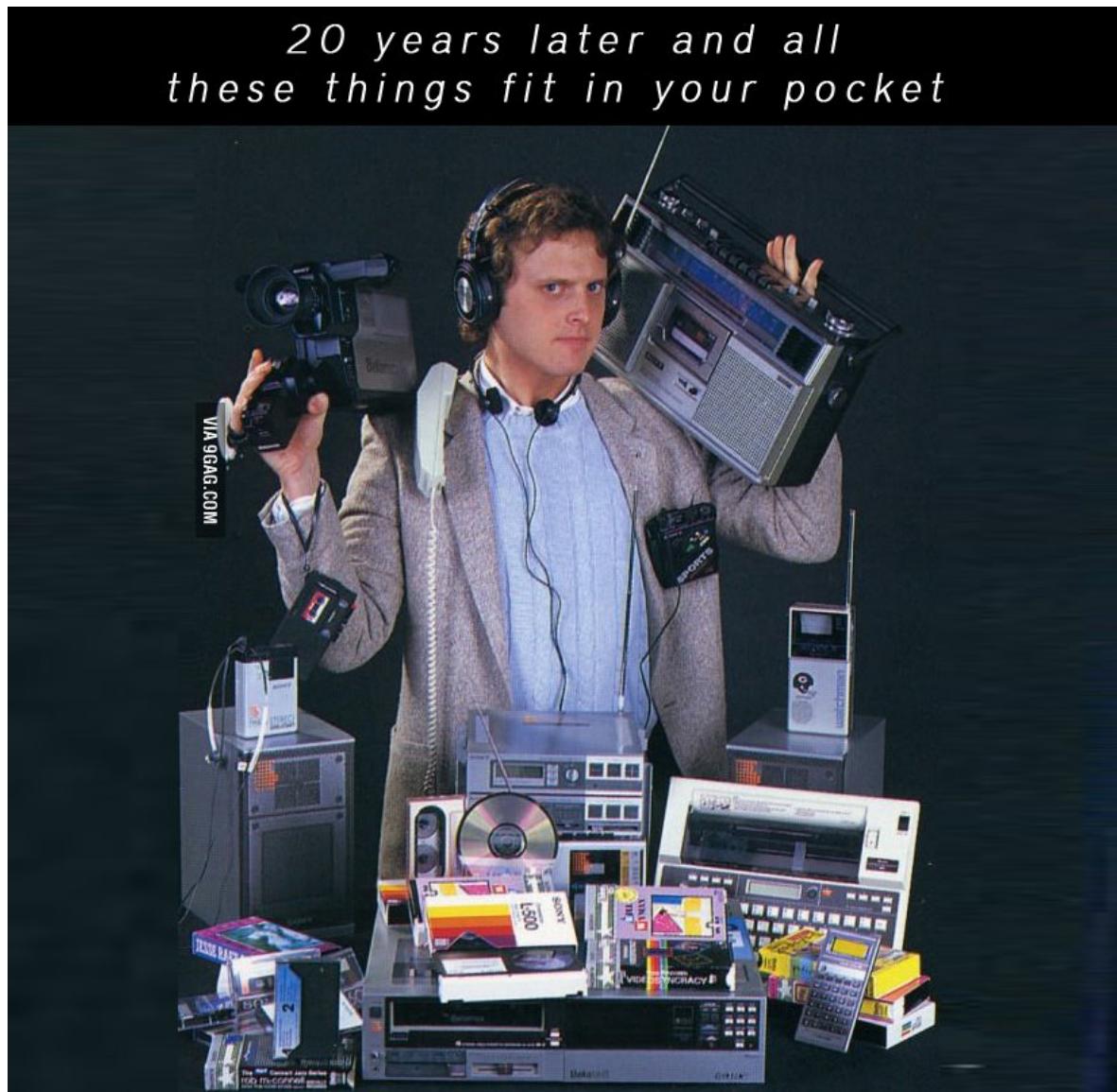
# Announcements

- 15 students still have not registered for a group.
  - Please talk to your tutor to help you find a group.
- There are 5 groups with less than 5 members.
  - Please talk to me !
- Recall the proposal:
  - What is the problem that your app will solve?
  - Why does the problem matter (e.g., motivation and significance)?
  - What is the app solution to the problem?
  - How will the solution be implemented (which should be clear for others to implement)?
- Discuss your idea with me.
- Sometimes I ask questions, argue, ...
  - Don't agree with me always, come with evidence !
- How do I look at your idea...
  - As a teacher, As an Engineer/Developer, As an Investor

# Outline

- Capabilities of modern smartphones
  - Sensors
  - Audio
  - Connectivity
  - Camera
- Android Basics 2
  - Broadcast Receiver
  - Content Provider
  - Services

# Capabilities of smartphones



# Smartphone capabilities



Smartphone

- **Motion** – Accelerometer, Gyroscope
- **Vision** – Multiple Cameras
- **Connectivity** – LTE, WiFi, Bluetooth, NFC
- **Location** – GPS, Assisted GPS
- **Audio** – Speaker, Microphone
- **Haptic** – Touch-screen, buttons
- **Biometric** – HR monitor, Iris scan, Fingerprint scan
- **Environment** – Magnetometer, Barometer, Proximity, Ambient Temperature, Ambient Light, Humidity, Ultraviolet (UV), Moisture, Pressure

# Today's Takeaway

- Smartphones are powerful.
- Allows developers to come up with innovative applications.
- How can we take advantage of capabilities of smartphones ?
  - Can you solve the problem you found using these capabilities ?
  - Can you use these capabilities to improve a current solution ?
  - Can you exploit these capabilities for an innovative new app ?

# **Hardware vs Software**

- **Hardware capability/sensor**
  - Physical components built into a handset
  - E.g. Accelerometer, Gyroscope, Ambient light, Pressure
- **Software capability/sensor**
  - Derive their data from one or more of the hardware-based sensors
  - E.g. Step Counter, Orientation
- **For sensors;**
  - Each sensor is accessible through the Android Sensor Framework [1]
  - Sensor availability is based on the actual handset and Android version.

[1] [https://developer.android.com/guide/topics/sensors/sensors\\_overview](https://developer.android.com/guide/topics/sensors/sensors_overview)

# Listing sensors in a device

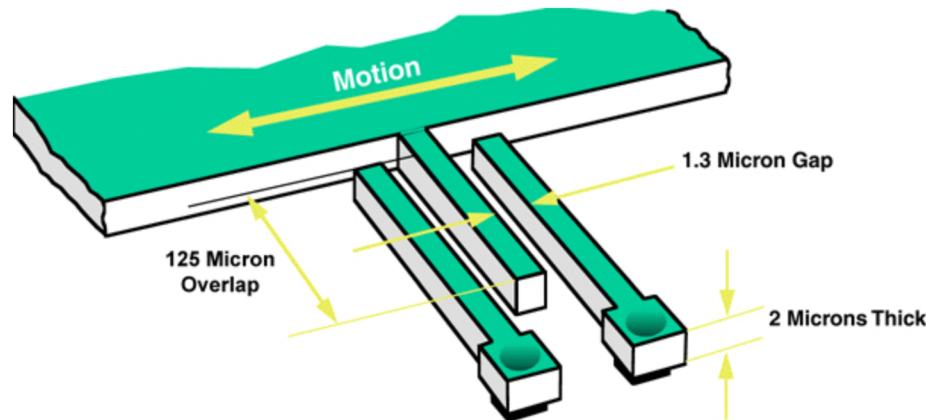
```
SensorManager mSensorManager =  
(SensorManager) getSystemService(Context.SENSOR_SERVICE);  
List<Sensor> deviceSensors = mSensorManager.getSensorList(Sensor.TYPE_ALL);  
  
for (Sensor temp : deviceSensors) {  
    Log.i(temp.getName(), " Sensors");  
}  
}
```

## E.g. List of sensors in Android

- Proximity
- Light
- Accelerometer
- Gyroscope
- Gyroscope (uncalibrated)
- Magnetometer
- Magnetometer (uncalibrated)
- Pressure
- Orientation
- Step detector
- Step counter
- Significant motion
- Gravity
- Linear Acceleration
- Rotation Vector
- Geomagnetic Rotation Vector
- Temperature
- Game Rotation Vector
- Tilt Detector
- Pickup Gesture
- Sensors Sync
- Double Twist
- Double Tap
- Window Orientation

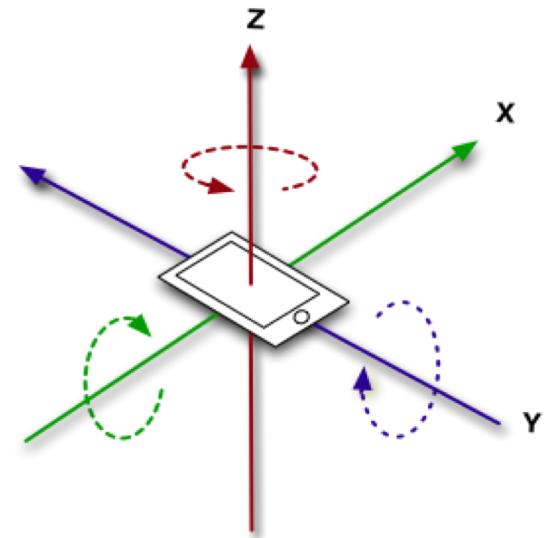
# Accelerometer

- Acceleration is the second derivative of displacement.
- One method is to get the displacement signal and calculate the second derivative.
- There are various types of accelerometers. Smartphones usually comes with Tri-axial MEMS (Micro-Electro-Mechanical Systems) accelerometers.



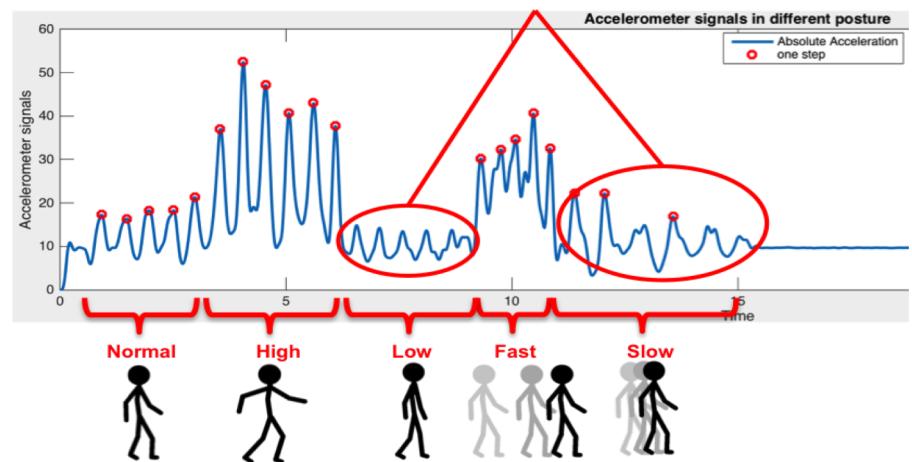
# Gyroscope

- Gyroscope measures the rate of rotation in rad/s around a device's x, y, and z axis
- Provides precise device orientation than accelerometer
- Often used in combination with accelerometer
- What can we do with Accelerometers and Gyroscopes ?
  - Camera/Photo apps



# Accelerometer + Gyroscope

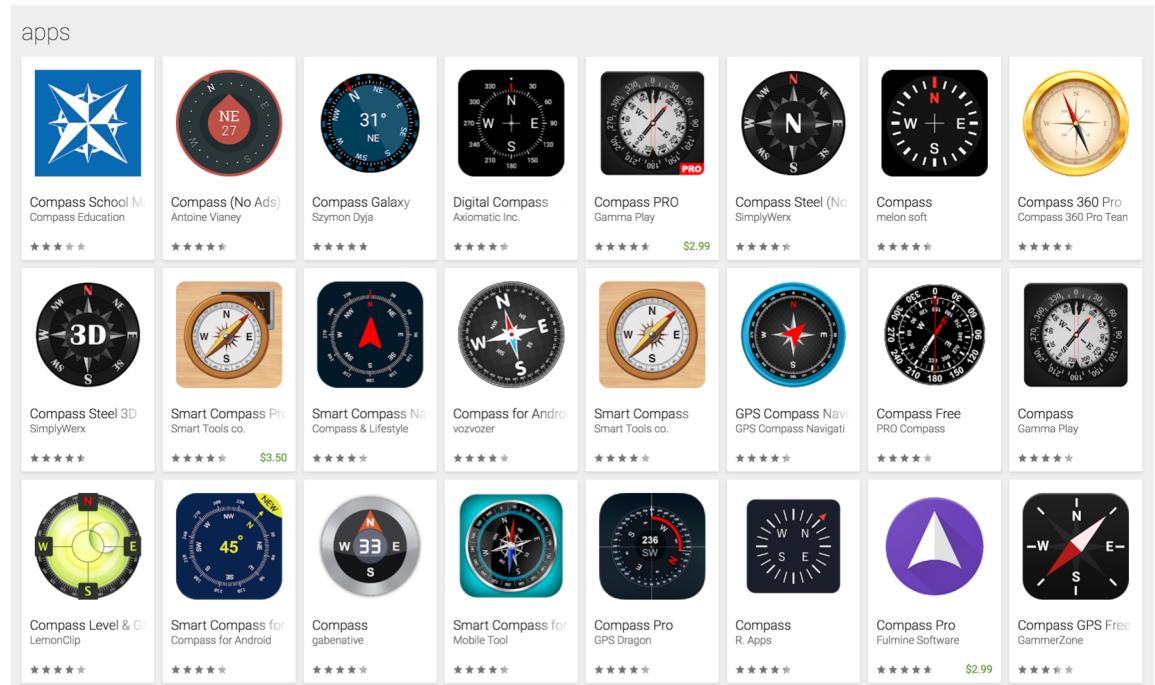
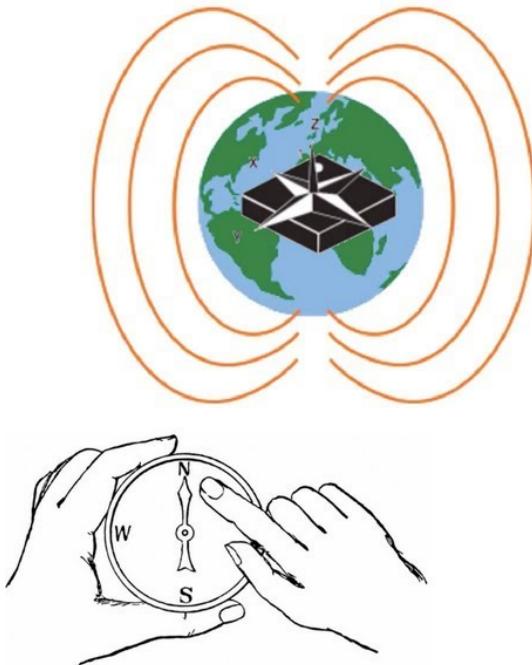
- One of the most used sensors
- Provides the information of device movement across all three axes.
- Determine device's orientation
  - Portrait or Landscape
  - Screen facing up or down
  - Games
- Activity monitoring
  - Step counting
  - Running or walking
  - Speed of running
  - Distance travelled (when GPS is not available) – How ?
- What else can we do?



# **Accelerometer + Gyroscope**

# Magnetometer

- Detect magnetic fields (e.g. earth magnetic field)

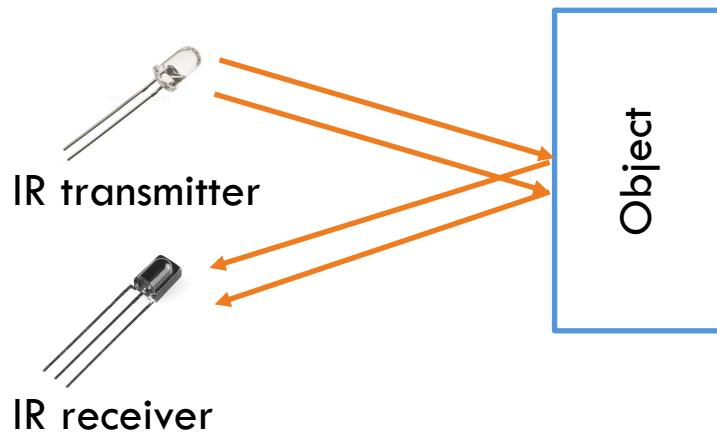


What else can we do ?

# Magnetometer

# Proximity sensor

- Infrared (IR) LED (Transmitter) and IR Receiver
- Not visible to human eye



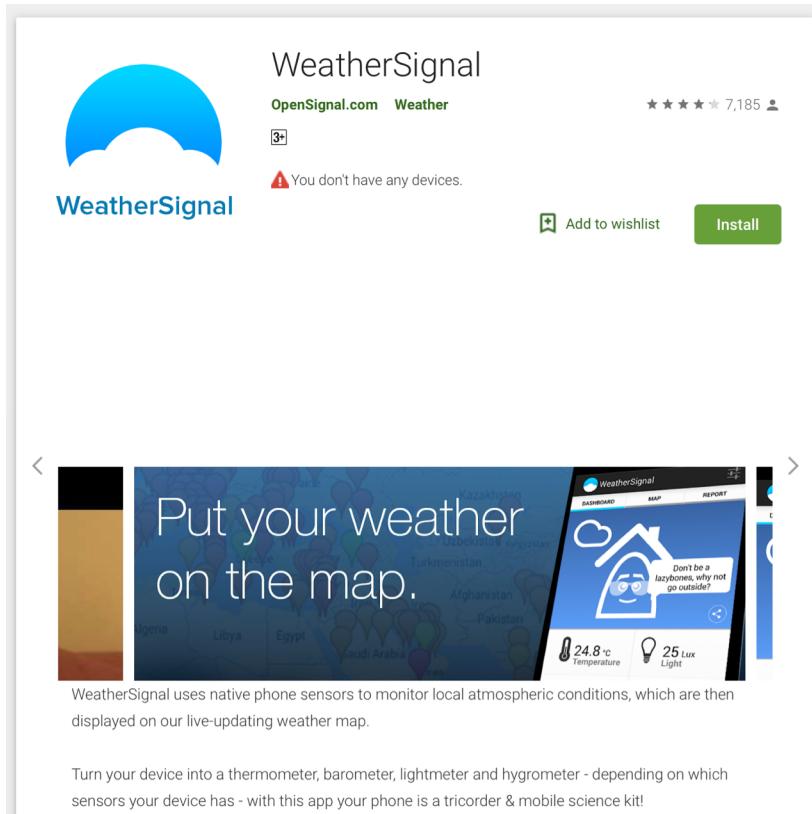
- **What is the most common use case of proximity sensor ?**
- **What else can we do ?**

# **Environment sensors**

- Ambient Light
  - Adjust screen brightness – Why ?
- Atmospheric pressure sensor – Barometer
  - Improve GPS accuracy – How ?
- Temperature sensor
  - Shuts device down if overheated
- Humidity sensor
  - Contributes to air quality measurements

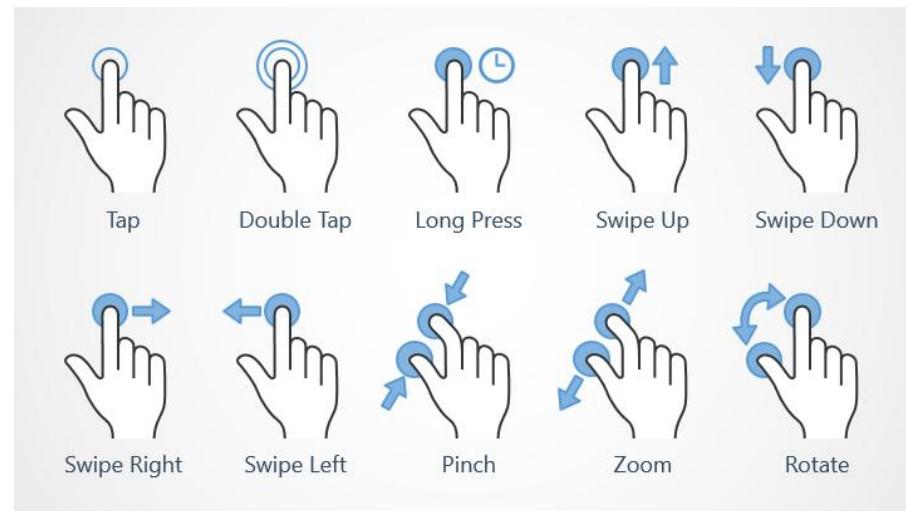
# Environment sensors

- Ambient Temperature, Ambient Light, Humidity, Ultraviolet (UV), Moisture, Pressure
- Crowdsourced weather apps



# Haptic

- **Touchscreen** - Advanced user interaction methods
  - Multiple modes of touch
  - Multiple modes of swipe



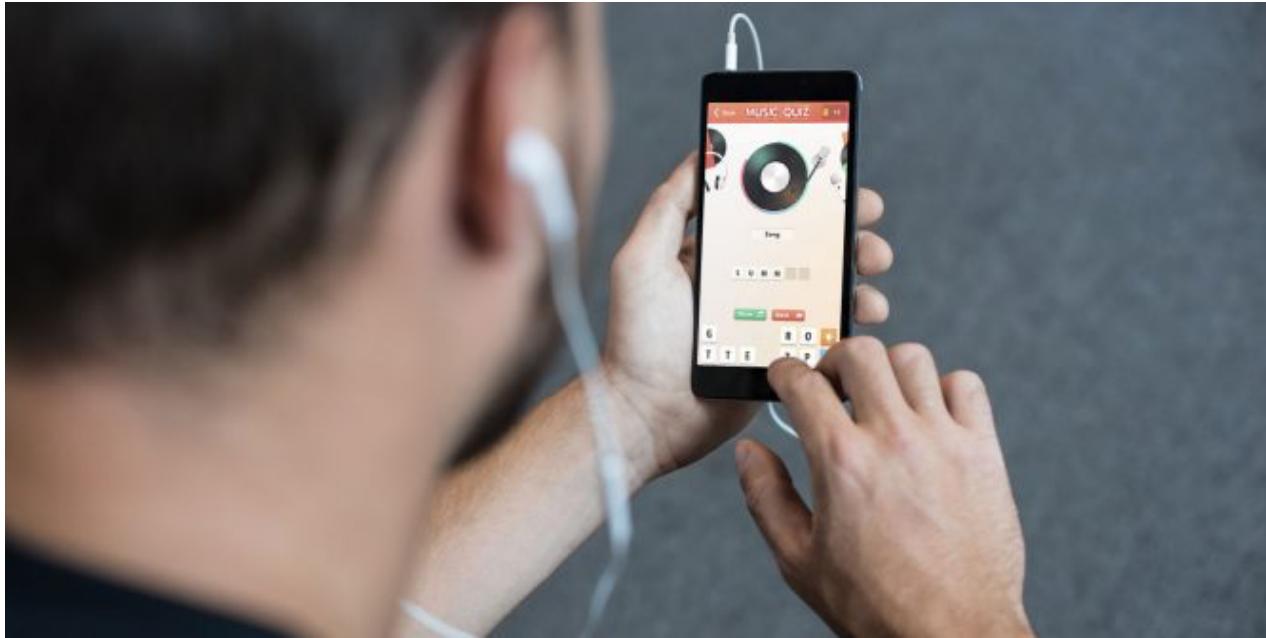
**What else can we do ?**

# Haptic

- Touch patterns for user authentication
  - Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. 2012. Touch me once and i know it's you! implicit authentication based on touch screen patterns. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12). Association for Computing Machinery, New York, NY, USA, 987–996.
- Touch for cross-device tracking of usage
  - Masood, R., Zhao, B. Z. H., Asghar, H. J.,&Kaafar, M. A. (2018). Touch and You're Trapp (ck) ed: Quantifying the Uniqueness of Touch Gestures for Tracking. *Proceedings on Privacy Enhancing Technologies*, 2018(2), 122-142.
- Soft biometric prediction using touch characteristics
  - Gender, age
  - Emotion
  - Miguel-Hurtado, O., Stevenage, S. V., Bevan, C.,&Guest, R. (2016). Predicting sex as a soft-biometrics from device interaction swipe gestures. *Pattern Recognition Letters*, 79, 44-51.
- Touch to enable parental controlling on the phone
  - X. Li, S. Malebary, X. Qu, X. Ji, Y. Cheng, and W. Xu, “icare: Automatic and user-friendly child identification on smartphones,” in Proceedings of the 19th International Workshop on Mobile Computing Systems; Applications, ser. HotMobile '18. New York, NY, USA: ACM, 2018, pp. 43–48.

# Audio

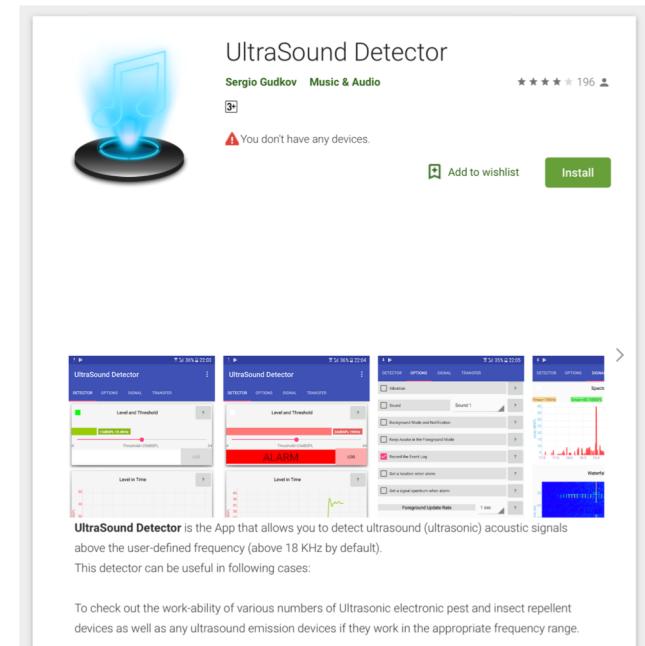
- Speakers
- Microphones



**What else can we do ?**

# Audio

- **Voice Recognition**
  - Personal assistants – “Hey Siri” and “OK Google”
  - Driven by the recent advances of deep machine learning
- **Ultra-sound (beyond 18kHz) based solutions**
- **How can we user Audio as a sensor for Advertising ?**

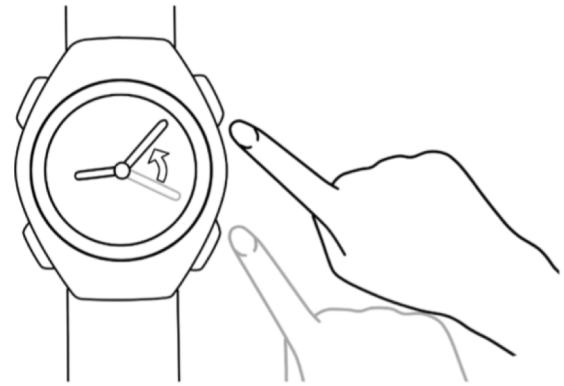


The screenshot shows the Google Play Store page for the "UltraSound Detector" app. The app is developed by Sergio Gudkov and falls under the category of Music & Audio. It has a rating of 4 stars from 196 reviews. The main image on the page shows a blue glowing sound wave icon. Below the image, there's a message: "You don't have any devices." There are "Add to wishlist" and "Install" buttons. The page also displays several screenshots of the app's interface, which includes various settings like "Level and Threshold", "Vibration", and "Sound 1". A detailed description at the bottom states: "UltraSound Detector is the App that allows you to detect ultrasound (ultrasonic) acoustic signals above the user-defined frequency (above 18 KHz by default). This detector can be useful in following cases: To check out the work-ability of various numbers of Ultrasonic electronic pest and insect repellent devices as well as any ultrasound emission devices if they work in the appropriate frequency range."

# Audio

## – Acoustic motion tracking

- Wenguang Mao, Jian He, and Lili Qiu. 2016. CAT: high-precision acoustic motion tracking. In Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom '16). ACM, New York, NY, USA, 69-81.
- Wei Wang, Alex X. Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom '16). ACM, New York, NY, USA, 82-94.



## – Encounter profiling

- Huanle Zhang, Wan Du, Pengfei Zhou, Mo Li, and Prasant Mohapatra. 2016. DopEnc: acoustic-based encounter profiling using smartphones. In Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom '16). ACM, New York, NY, USA, 294-307.

## – BreathPrint: breathing acoustic for user authentication

- Chauhan, J., Hu, Y., Seneviratne, S., Misra, A., Seneviratne, A., & Lee, Y. (2017, June). BreathPrint: Breathing acoustics-based user authentication. In Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services (pp. 278-291). ACM.

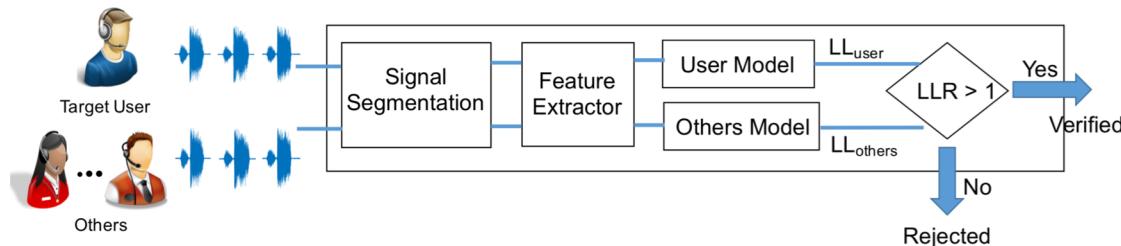
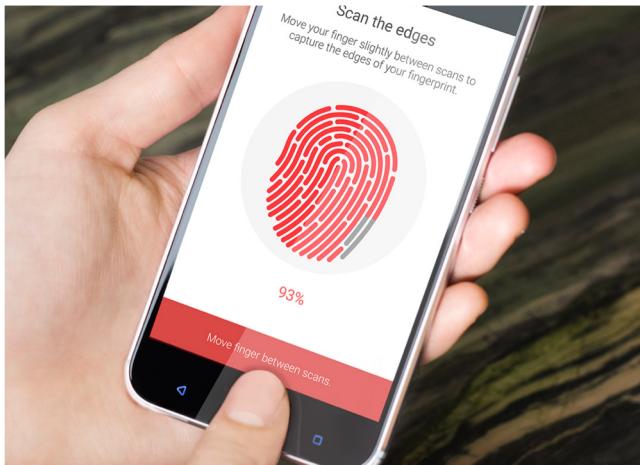


Figure 5: *BreathPrint*: System Architecture

# Biometric sensors

- Fingerprint and Iris scanning for authentication



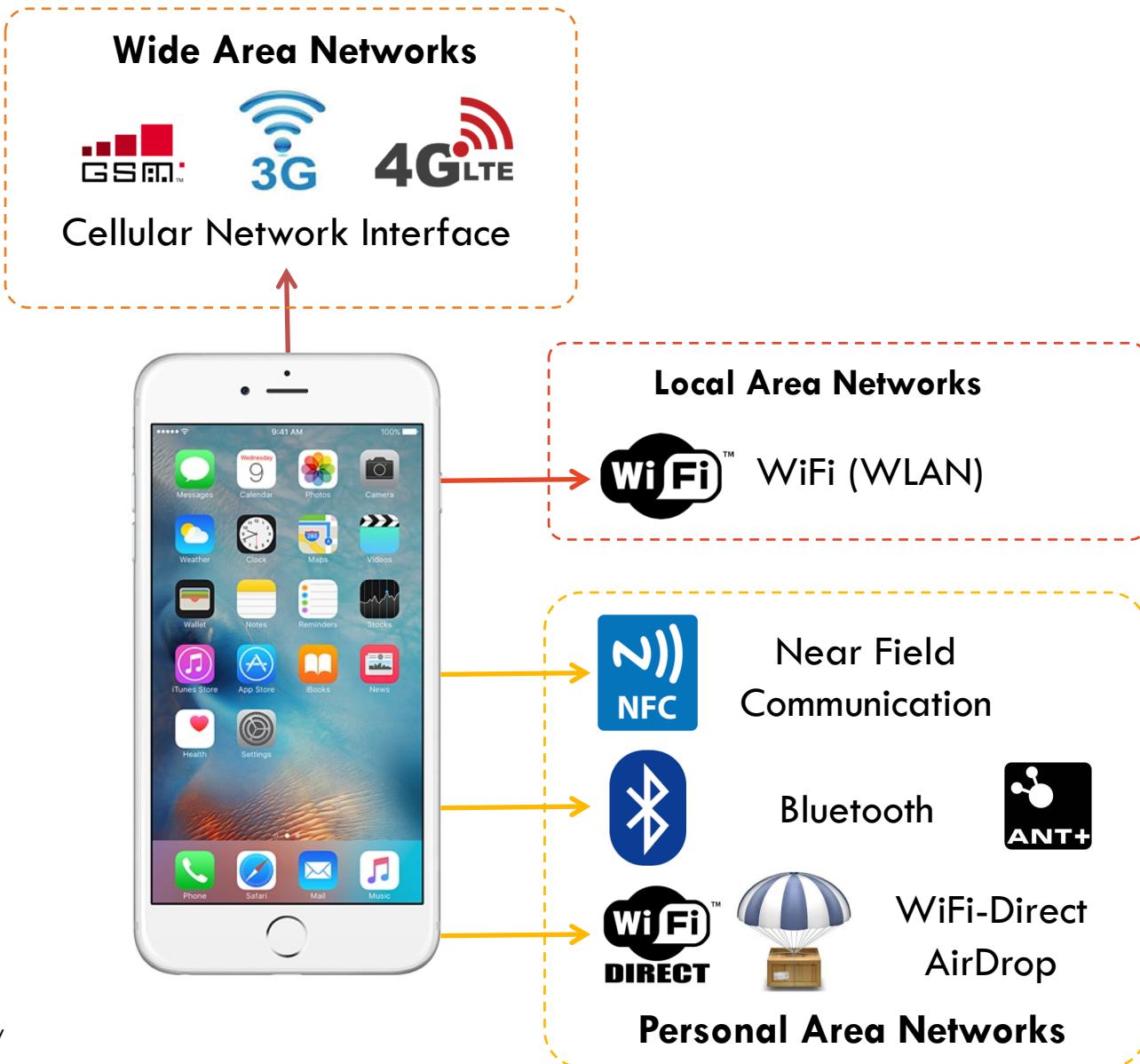
- Samsung Galaxy S9 comes with a heart rate, blood oxygen level sensor
  - Another IR base sensor
  - Measure the characteristics of the received signal

# External Sensors



Seneviratne, S., Hu, Y., Nguyen, T., Lan, G., Khalifa, S., Thilakarathna, K., ...&Seneviratne, A. (2017). A survey of wearable devices and challenges. *IEEE Communications Surveys & Tutorials*, 19(4), 2573-2620.

# Connectivity



# Connectivity

- Connectivity empowers smartphones to control variety of smart devices



- New services through short-range connectivity



Card-less Payment



Location based services through iBeacons

- Advertising, Indoor navigation, ...

# Connectivity

- **Secure key-generation with wireless signal characteristics**
  - Zhang, J., Duong, T. Q., Marshall, A., & Woods, R. (2016). Key generation from wireless channels: A review. *IEEE Access*, 4, 614-626.
- **Gesture recognition with wireless signals**
  - <http://wittrack.csail.mit.edu>
- **Non-invasive Breathing disorder detection with smartphones**
  - Nandakumar, R., Gollakota, S., & Watson, N. (2015, May). Contactless sleep apnea detection on smartphones. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (pp. 45-57). ACM.

# Today's Takeaway

- Smartphones are powerful.
- Allows developers to come up with innovative applications.
- How can we take advantage of capabilities of smartphones ?
  - Can you solve the problem you found using these capabilities ?
  - Can you use these capabilities to improve a current solution ?
  - Can you exploit these capabilities for an innovative new app ?

## Today's Takeaway

- **How can we take advantage of capabilities of smartphones ?**
- Example Question 1: (solving an specific problem)
- You have one smart electric light and home WiFi. You want to switch-on your light when you move closer to the light (or a specific location). However, you do not have motion detection sensors with you. The developer of the smart electric light provides SDK to develop third party apps. How do you design a mobile app for this specific purpose ?

# Camera

- 4K videos with 30fps
- Multiple cameras on one device



**What else can we do ?**

# Beyond photos and videos

ARuler - AR Ruler app, Measure tools



Grymala Tools

★★★★★ 1,669

3+

Offers in-app purchases

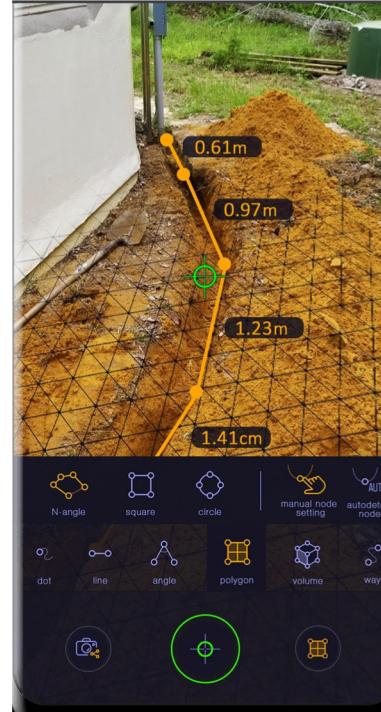
⚠ You don't have any devices.

Add to wishlist

Install



ARuler - the first and the best free Augmented Reality measurement app for Android!



# Beyond photos and videos

- Document scanning apps



## Scanner App: Scan PDF Document 4+

iScanner for iPhone & iPad

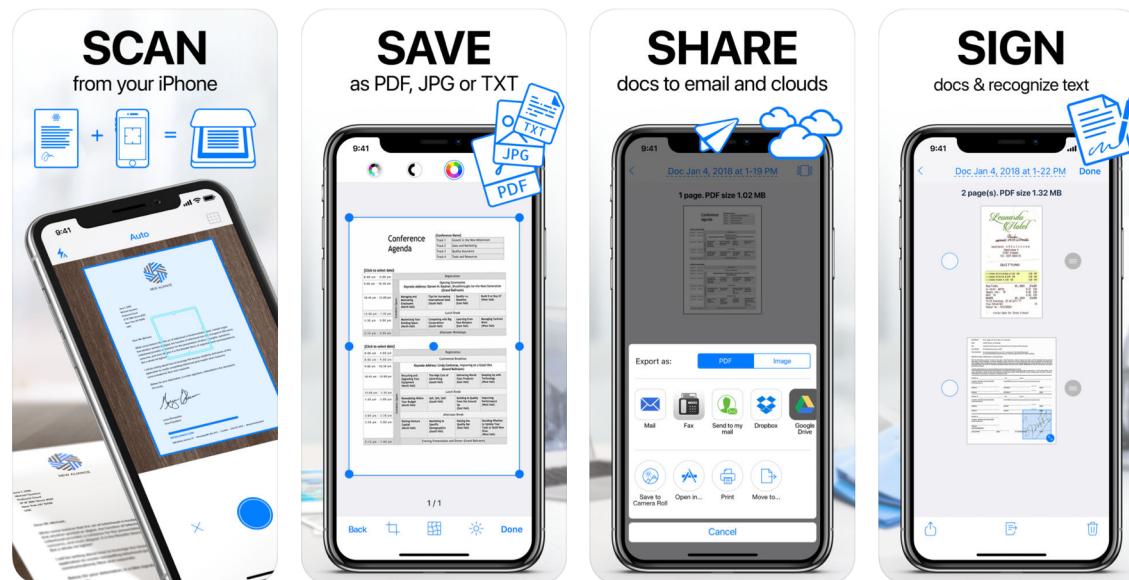
BPMobile

#2 in Business

★★★★★ 4.6, 29.6K Ratings

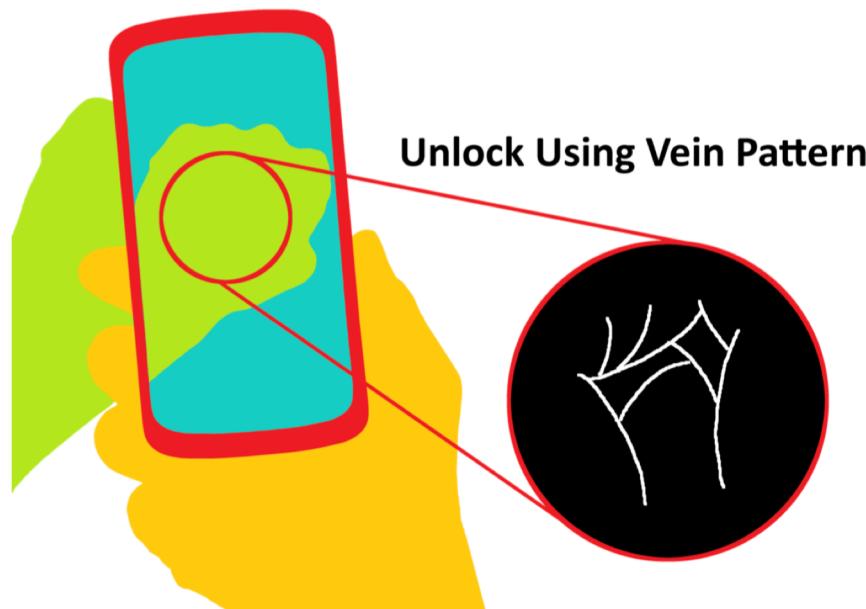
Free · Offers In-App Purchases

### Screenshots iPhone iPad



# Beyond photos and videos

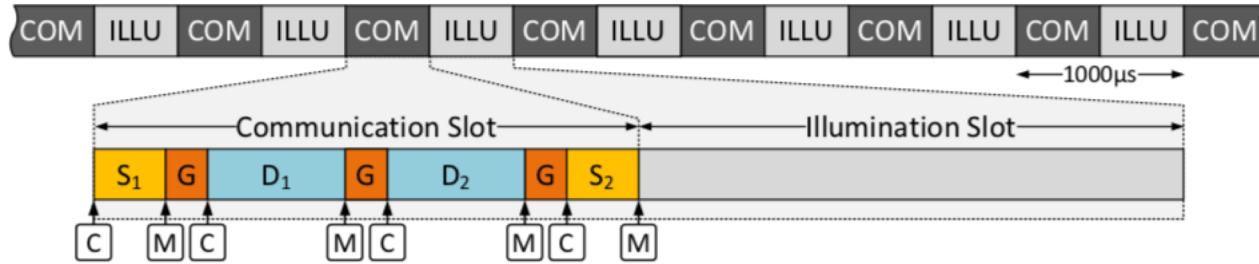
- VeinDeep – use vein patterns of your hand for authentication
- Zhong, Henry, Salil S. Kanhere, and Chun Tung Chou. "VeinDeep: Smartphone unlock using vein patterns." *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*. IEEE, 2017.



# Beyond photos and videos

## – Visible light communication

- Stefan Schmid, Linard Arquint, and Thomas R. Gross. 2016. Using smartphones as continuous receivers in a visible light communication system. In Proceedings of the 3rd Workshop on Visible Light Communication Systems (VLCS '16). ACM, New York, NY, USA, 61-66.



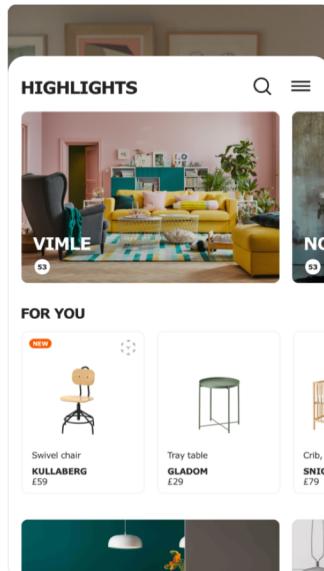
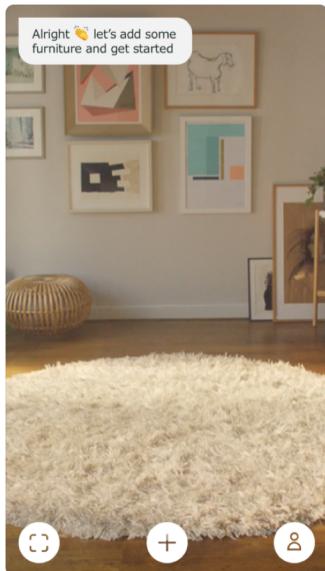
# Beyond photos and videos



**IKEA Place** 4+  
Augmented Reality Furnishing  
Inter IKEA Systems B.V.

★★★★★ 4.7, 1.6K Ratings  
Free

## Screenshots [iPhone](#) [iPad](#)

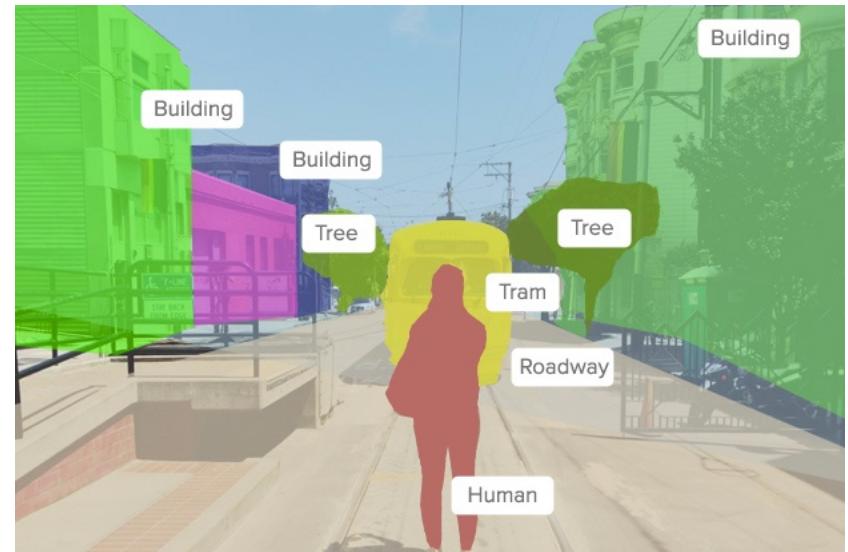


# Mobile Augmented Reality

- Many AR SDKs
  - Apple's ARKit - <https://developer.apple.com/arkit/>
  - Google's ARCore - <https://developers.google.com/ar/discover/>
  - Wikitude - <https://www.wikitude.com/>
- Cross-platform development – Unity
  - <https://unity.com/solutions/mobile-ar>

# Mobile AR

- Common features
  - Scene recognition
  - Object recognition
  - Tracking of objects
  - Motion tracking



# Mobile Mixed Reality

- Zapbox - <http://www.zappar.com/zapbox/>
  - Mixed Reality Video - <https://youtu.be/RnnZ3YQD4ig>
- Google Cardboard - <https://vr.google.com/cardboard/>
  - Google AR & VR - <https://arvr.google.com>
- Structure sensor for iPhone – <https://structure.io/structure-sensor>



## Today's Takeaway

- Smartphones are powerful.
- Allows developers to come up with innovative applications.
- How can we take advantage of capabilities of smartphones ?
  - Can you solve the problem you found using these capabilities ?
  - Can you use these capabilities to improve a current solution ?
  - Can you exploit these capabilities for an innovative new app ?

## Today's Takeaway

- How can we take advantage of capabilities of smartphones ?
- Example Question 2: (open-ended)
- You started a new company to sell small electronic items online. Smartphones can be used in numerous ways to enhance the efficiency of every business. How to do design an innovative mobile app to improve your productivity of your new business?

# **Building blocks of Android - 2**

# Building blocks of Android

## App components

- Activities
- Broadcast Receivers
- Content Providers
- Services
- Activating components – Intent
- **Android Developer Page**
  - <https://developer.android.com>
- Many books in the library
  - Android App Development by Franceschi  
<https://www.safaribooksonline.com/library/view/android-app-development/9781284092134/?ar>

# Intents

- Intent is a messaging object to request an action from another app component.
- Primary use-cases:
  - **To start an activity**
  - **To start a service**
  - **To deliver a broadcast**
- Intent types:
  - **Explicit Intents:** Communicate within the same application. Need to specify the exact name of the component , e.g. class name.
  - **Implicit Intents:** Communicate between applications. Requested by declaring the general action to perform, e.g. location.
- <https://developer.android.com/guide/components/intents-filters>

# Building blocks of an Intent

## 1. Component name

- Name of the component to start
  - Must specify the name for *Explicit Intent*, e.g. class name of the new Activity.
  - Empty for *Implicit Intent*

## 2. Action

- String that specifies the desired operation, e.g. view or pick
  - `ACTION_DIAL` - Dial a number
  - `ACTION_EDIT` - Display data to edit
  - `ACTION_SYNC` - Synchronise device data with a server
  - `ACTION_MAIN` - Start as initial activity of the app.

# Building blocks of an Intent

## 3. Data

- Data and type of data (MIME type) associated with the Intent
- Type of data should be related to the action
  - E.g. If the action is ACTION\_DIAL, data should be the phone number.
- Formatted as URI object (Uniform Resource Identifier)
  - `Uri.parse("http://www.google.com")`
- Note: To set both URI and MIME type, use `setDataAndType()`

# Building blocks of an Intent

## 4. Category

- String containing additional information about the component
  - `CATEGORY_BROWSABLE` – To start a web browser to display data
- Specify the category with `addCategory()`

## 5. Extras

- Key-value pairs that carry additional information to complete the action
- Add extra info with `putExtra()`

## 6. Flags

- Metadata for the intent
  - E.g. How to launch the activity, how to treat it after launching, etc.
- Can set flags using `setFlags()`

# Example

- Start another activity using an Intent
- Example: Tutorial 2
  - What type of an Intent is used ?

```
@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
    String updateItem = (String) itemsAdapter.getItem(position);
    Log.i("MainActivity", "Clicked item " + position + ": " + updateItem);

    Intent intent = new Intent(MainActivity.this, EditToDoItemActivity.class);
    if (intent != null) {
        // put "extras" into the bundle for access in the edit activity
        intent.putExtra("item", updateItem);
        intent.putExtra("position", position);
        // brings up the second activity
        startActivityForResult(intent, EDIT_ITEM_REQUEST_CODE);
        itemsAdapter.notifyDataSetChanged();
    }
}
```

## Example 2

- Communicate between apps.
- By declaring the general action to perform. In this case,
  - Action: **ACTION\_VIEW**
  - Data: Formatted as Uniform Resource Identifier (URI object) to send Intent Data

```
Intent browserIntent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.google.com"));
startActivity(browserIntent);
Intent chooser = Intent.createChooser(browserIntent, "Load http://www.google.com");
});
```

- What type of intent is this?
- What can go wrong with the code above code block?

# Intent Filters

- Declare which Intents that your app can receive with **intent-filter** element in your **AndroidManifest.xml**
- This is how Android pass Implicit Intents to relevant apps
- Define **<action/>**, **<data/>** and **<category/>**
- E.g. Declaration to receive **ACTION\_SEND** intent with **text** data

```
<activity android:name="ShareActivity">
    <intent-filter>
        <action android:name="android.intent.action.SEND" />
        <category android:name="android.intent.category.DEFAULT" />
        <data android:mimeType="text/plain" />
    </intent-filter>
</activity>
```

## Intent Filters

- Who had a look at the AndroidManifest.xml files of Tutorial 1?
- Were there any Intent filter?

# Intent Filters

- **ACTION\_MAIN** indicates this activity is the main entry point when the user launch the app and does not expect any intent data.
- **CATEGORY\_LAUNCHER** indicates that activity's icon should be placed in the system's app launcher.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="comp5216.sydney.edu.au.todolist">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".EditToDoItemActivity"
            android:label="@string/app_name" >
        </activity>
    </application>

</manifest>
```

# Broadcast Receiver

- Things are happening all the time on an Android device.
- Many components need to know that some events have occurred.
  - New package installed.
  - Phone call received.
  - WiFi is connected.
  - Device is rebooted.
- Android uses a **Broadcast Intent** to tell everyone about it.
- All intents can be found at `BROADCAST_ACTIONS.TXT` file in the relevant SDK
- <https://developer.android.com/guide/components/broadcasts>

# Broadcast Receiver

- Broadcast Receivers register to receive events.
- Android routes the Broadcast Intents to Broadcast Receivers that have registered to receive them.
- Register for Broadcast Intents
  - **Statically:** at `AndroidManifest.xml`
  - Using `<receiver>` and `<intent-filter>`

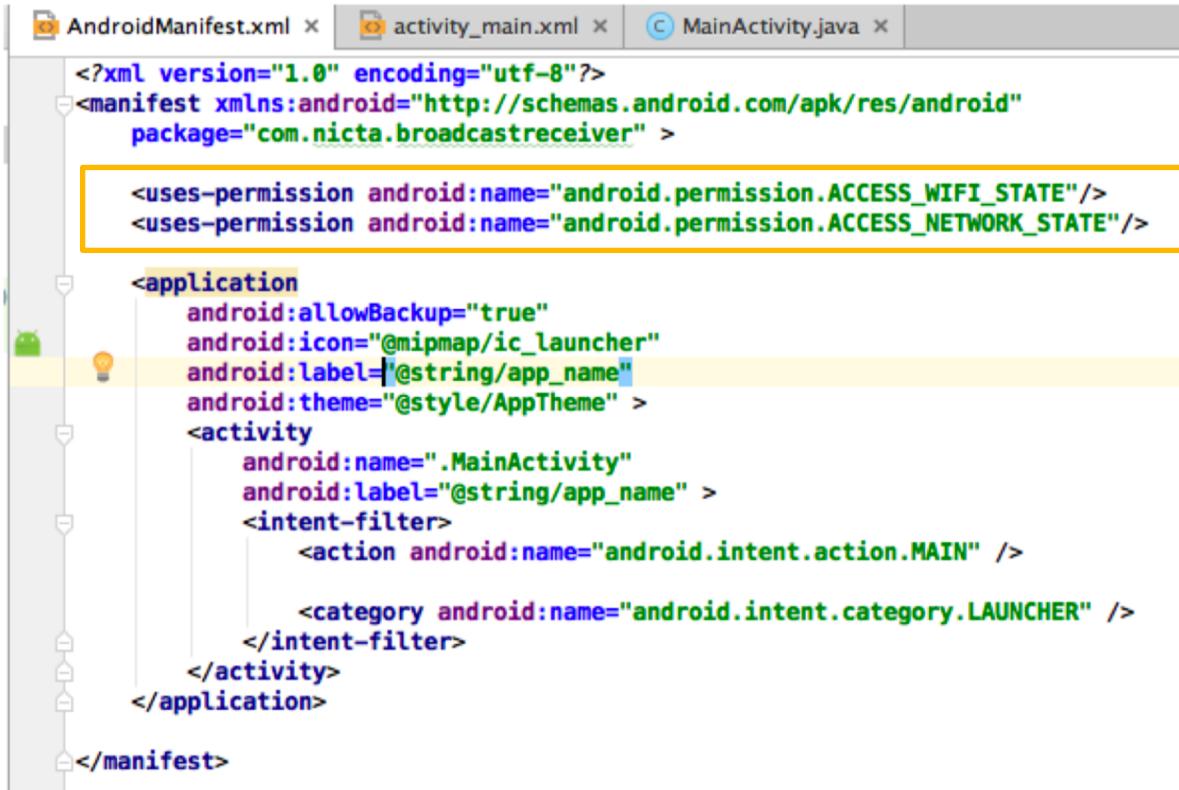
```
<receiver>
  <intent-filter>
    <action android:name="android.intent.action.BOOT_COMPLETED" />
  </intent-filter>
</receiver>
```

# Dynamic Registration

- At `MainActivity.java`
- Steps:
  - Create an Intent Filter
  - Create a `BroadcastReceiver`
  - Register `BroadcastReceiver` (`registerReceiver()`)
  - Unregister `BroadcastReceiver` (`unRegisterReceiver()`).
- To receive many broadcasts → ?

# User permission

- To protect the privacy of the user
- Permissions are categorized as *normal* and *dangerous*
- Declare permission at **AndroidManifest.xml**



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.nicta.broadcastreceiver" >

    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

# Content Provider

- Provides access to a central repository of structured data.
- Use to **securely** exchange data between applications.
- Access Content Providers through a **CursorLoader** and **ContentResolver**
  - Database style interface.
- Android content providers
  - **Contacts, Audio, Video, Images, Calendar, User Dictionary**
- For more info: <http://developer.android.com/guide/topics/providers/content-providers.html>

# Content Provider

- Accessing content provider – API “CURD”
  - Create (insert)
  - Retrieve (query)
  - Update
  - Delete
- Example: Accessing User Dictionary. Stores non-standard words that user wants to keep

word	app id	frequency	locale	_ID
mapreduce	user1	100	en_US	1
precompiler	user14	200	fr_FR	2
applet	user2	225	fr_CA	3
const	user1	255	pt_BR	4
int	user5	100	en_UK	5

# Content Provider

- Querying from another app.
- `mCursor = getContentResolver().query(`
- `Uri,`  → The content URI of the words table, FROM table-name
- `projection,`  → The columns to return for each row
- `selection,`  → Selection criteria, WHERE col = value
- `selectionArgs,`  → Selection criteria
- `sortOrder)`  → The sort order for the returned rows
- Content URI have the syntax:
  - **content://authority/path/id**
  - E.g. //user\_dictionary/words/5
- Remember to declare permission

# Services

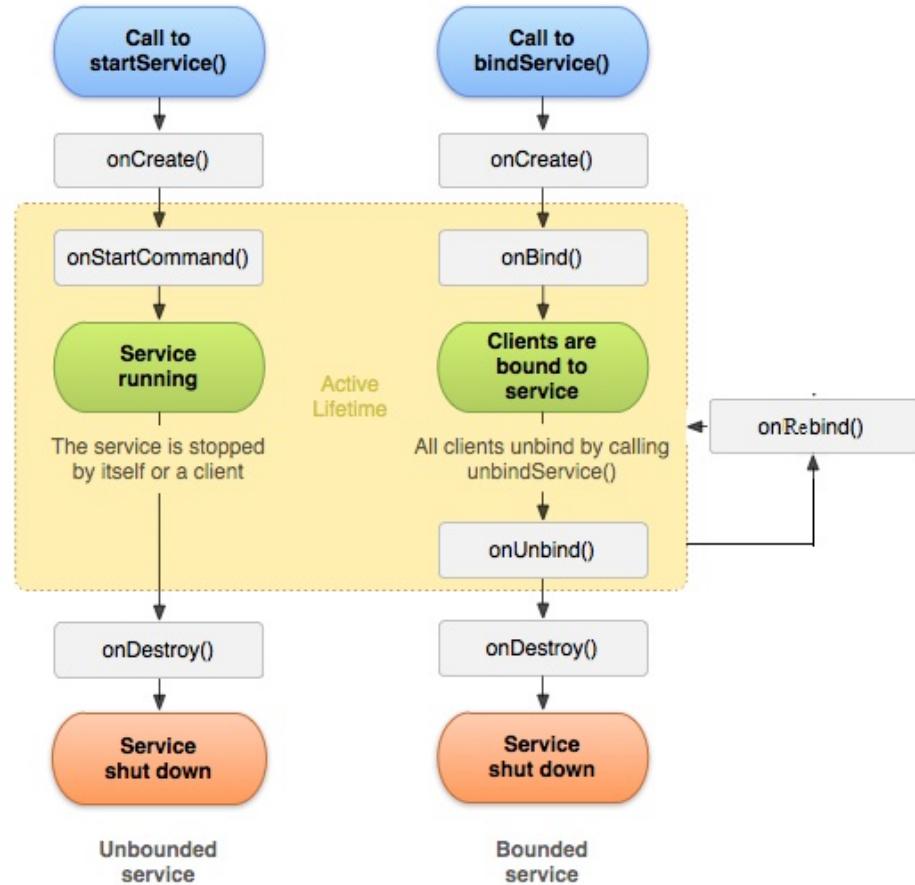
- Does not involve a GUI component. Runs in the **background** and suitable for long running processes.
- Example functionalities achieved through services are network communications, play music, and software updates.
- Three types of services:
  - **Foreground**
  - **Background**
  - **Bound**
- For more info:  
<https://developer.android.com/guide/components/services>

# Services

- **Foreground**
  - Operation is noticeable to the user and must display a Notification
  - Does not require user interaction
  - E.g. Audio playback
- **Background**
  - Runs in the background
  - E.g. downloading a movie
- **Bound**
  - Offers client-server interface
  - Allows other app components to interact with the service
  - E.g. for long running services and multiple operations
  - <https://developer.android.com/guide/components/bound-services>

# Service Lifecycle

- Use explicit intent when starting a Service – Why ?
- Example:
  - Design and app to increase the screen brightness to maximum when phone is charging.



# Declaring components – AndroidManifest.xml

- Describes essential information about your app to Android OS
- Examples:
  - App package name
  - Minimum API level required by the app
  - User permissions
  - Declare third party API libraries
  - Declare app's components, e.g. Activity
  - Declare component capabilities through Intents and Intent filters

```
<?xml version="1.0" encoding="utf-8"?>
<manifest ... >
    <application android:icon="@drawable/app_icon.png" ... >
        <activity android:name="com.example.project.ExampleActivity"
                  android:label="@string/example_label" ... >
            </activity>
            ...
        </application>
    </manifest>
```

<https://developer.android.com/guide/topics/manifest/manifest-intro>

# What's Next ?

## Tutorial 3 – Data Storage and Management

- Each app is having its own storage where it can write – **Internal Storage**
- If it wants to write to the **external storage** it needs request for the permissions `android.permission.WRITE_EXTERNAL_STORAGE`.
- More details can be found at
- <http://developer.android.com/training/basics/data-storage/files.html>
- Week 4: Challenges associated with mobile computing