

# Андройд – Лекц 13

Ж.Золжаргал  
([j.zoljargal@csms.edu.mn](mailto:j.zoljargal@csms.edu.mn))



# Using the Media APIs

- Андройд нь дараах форматуудыг дэмждэг
  - JPEG
  - PNG
  - OGG
  - Mpeg 4
  - 3GPP
  - MP3
  - Bitmap



# Playing Media Resources

- Андройд-д Multimedia-г MediaPlayer классын тусламжтай тоглуулна
- Апп-ийн нөөцөд, локал файлууд, эсвэл сүлжээнд URI-аар өгөгдсөн медиа-г тоглуулна
- Тухайн медиаг start функцээр эхлүүлэхээс өмнө prepare буюу урьдчилан бэлдэх функц дуудах шаардлагатай



# Playing Media Resources

```
String MEDIA_FILE_PATH = Settings.System.DEFAULT_RINGTONE_URI.toString();
MediaPlayer mpFile = new MediaPlayer();

try {
    mpFile.setDataSource(MEDIA_FILE_PATH);
    mpFile.prepare();
    mpFile.start();
}
catch (IllegalArgumentException e) {}
catch (IllegalStateException e) {}
catch (IOException e) {}
```

# Playing Media Resources

- `setLooping` – дуусаад давтаж тоглуулах
- `getFrame` – Одоогын тоглож байгаа бичлэгийн frame-ийг bitmap-аар авна
- `Release` – Одоогын тоглуулж буй медиаг зогсоож, нөөцийг суллана
- Андроид нь Media Player объектыг ашиглах тоо хязгаартай тул түүнийг суллахгүй бол Runtime exception шиддэг



# Recording Multimedia

- Медиа бичихэд MediaRecorder класс ашиглагддаг
- Аудио болон видео бичихийн тулд дээрх классын объектыг ашигладаг

```
MediaRecorder mediaRecorder = new MediaRecorder();
```

- Андройдод ямар нэгэн медиа бичихийн тулд харгалзах зөвшөөрлийг авна

```
<uses-permission android:name="android.permission.RECORD_AUDIO"/>  
<uses-permission android:name="android.permission.RECORD_VIDEO"/>
```

# Recording Multimedia

- Медиа бичлэг хийхэд видео эсвэл аудио, output format, video size, frame rate, encoder зэргийг тохируулж өгнө

```
// Set the audio source.  
mediaRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);  
// Set the output format.  
mediaRecorder.setOutputFormat(MediaRecorder.OutputFormat.DEFAULT);  
// Set the audio encoders to use.  
mediaRecorder.setAudioEncoder(MediaRecorder.AudioEncoder.DEFAULT);  
...
```

# Recording Multimedia

```
mediaRecorder.setOutputFile("myoutputfile.mp4");
```

- setOutputFile функц нь prepare функцээс өмнө setOutputFormat функцээс дараа дуудагдах ёстой.

- Prepare функцын хойноос Start функцыг дуудсанаар бичиж эхлэнэ.

```
mediaRecorder.prepare();  
mediaRecorder.start();
```





# Recording Multimedia

- Дуусгахдаа, Stop функцээр бичлэгийг зогсоож Release функцээр Media Recorder-ийг чөлөөлнө

```
mediaRecorder.stop();  
mediaRecorder.release();
```

- Бичлэг хийх явцад тухайн бичлэгээ дэлгэцэн харах шаардлагатай үед setPreviewDisplay функцыг ашиглана

# Recording Multimedia

- Тухайн апп-аар үүссэн медиа файлд өөр бусад апп-аас хандах боломжгүй үед Media Store Content Provider-д metadata-г бүртгүүлсэнээр хандах боломжтой болно
- Ингэхдээ Content Values классын объектыг үүсгэж бүртгүүлнэ.

```
ContentValues content = new ContentValues(3);  
content.put(Audio.AudioColumns.TITLE, "TheSoundandtheFury");  
content.put(Audio.AudioColumns.DATE_ADDED,  
            System.currentTimeMillis() / 1000);  
content.put(Audio.Media.MIME_TYPE, "audio/amr");
```

# Recording Multimedia

- Үүссэн медиа файлын зам

```
content.put(MediaStore.Audio.Media.DATA,  
            "myoutputfile.mp4");
```

- Програмын ContentResolver-д хандах бөгөөд түүнийг Media Store-д шинэ бичлэг нэмэхдээ ашиглана.

```
ContentResolver resolver = getContentResolver();  
Uri uri = resolver.insert(Audio.Media.EXTERNAL_CONTENT_URI, content);
```

- Media Store амжилттай нэмсэн бол Broadcast Intent-ээр түүнийг зарлана.

```
sendBroadcast(new Intent(Intent.ACTION_MEDIA_SCANNER_SCAN_FILE, uri));
```

# Using the Camera

- Camera- руу хандахад зөвшөөрлийг дараах байдлаар авна.

```
<uses-permission android:name="android.permission.CAMERA"/>
```

- Энэхүү зөвшөөрөл нь камерийн тохиргоо хийх, зураг авах, камерын урсгал харуулах
- Camera классын статик open функцээр камерийг эхлүүлэх бөгөөд release функцээр дуусгана.

```
Camera camera = Camera.open();  
[ ... Do things with the camera ... ]  
camera.release();
```

# Controlling Camera Settings

- Камерийн одоогын тохиоргоо `Camera.Parameters` об'єкетд хадгалагддаг бөгөөд `getParameters` функцээр одоогын параметр-г авна.
- Ирсэн параметр-н `set*` функцээр тохиргоог шинэчилдэг
- Тохиргоог өөрчилсөний дараагаар `setParameters` функцээр өөрчлөлтийг идэвхжүүлнэ.



# Controlling Camera Settings

```
Camera.Parameters parameters = camera.getParameters();  
parameters.setPictureFormat(PixelFormat.JPEG);  
camera.setParameters(parameters);
```

- Камерийн параметр нь image, preview size, image format, preview frame rate зэргийг тохируулахад хэрэглэгддэг.

# Using the Camera Preview

- Камераар preview хийж харах

```
camera.setPreviewDisplay(mySurface);  
camera.startPreview();  
[ ... ]  
camera.stopPreview();
```

- Camera объектын takePicture функцээр зураг авна

# Introducing the Sensor Manager

- Sensor Manager нь андройд төхөөрөмж дэх мэдрэгчүүдийг ашиглахад хэрэглэгддэг
- getSystemService функцээр sensor service-г барьж авна

```
String service_name = Context.SENSOR_SERVICE;  
SensorManager sensorManager = (SensorManager)getSystemService(service_name);
```

- Доорх байдлаар чагнах Event-үүдээ бүртгүүлнэ





# Introducing the Sensor Manager

- `onSensorChanged` функцыг даран тодорхойлох бөгөөд `sensor` –ын утгууд өөрчлөгдөхөд энэхүү функц нь ажиллах болно
- `onAccuracyChanged` функцыг даран тодорхойлох бөгөөд нарийн тогтсон утгууд авахад энэхүү функц нь ажиллана



# Introducing the Sensor Manager

`ISensorManager.SENSOR_STATUS_ACCURACY_HIGH` –

Боломжит хамгийн их утга

`ISensorManager.SENSOR_STATUS_ACCURACY_LOW` –

Боломжит хамгийн бага утга

`ISensorManager.SENSOR_STATUS_ACCURACY_MEDIUM` –

Дундаж утга

`ISensorManager.SENSOR_STATUS_UNRELIABLE` – тухайн утга  
нь мэдэгдэхгүй тохиолдолд



# Introducing the Sensor Manager

- `SensorManager.SENSOR_PRESSURE` – Агаарын даралт хэмжинэ
- `SensorManager.SENSOR_ACCELEROMETER` – Хурдатгал хэмжинэ
- `SensorManager.SENSOR_ORIENTATION` – Төхөөрөмжний хазайлт хэмжинэ
- `SensorManager.SENSOR_SOUND` – Орчины чимээг хэмжинэ



# Introducing the Sensor Manager

- `SensorManager.SENSOR_LIGHT` – гэрлийг мэдрэнэ
- `SensorManager.SENSOR_MAGNETIC_FIELD` – соронзон ороныг мэдрэнэ
- `SensorManager.SENSOR_PROXIMITY` – объектоос хол эсэхийг мэдрэнэ
- `SensorManager.SENSOR_TEMPERATURE` – дулааныг хэмжинэ



# Introducing the Sensor Manager

- Sensor Manager-г тухайн SensorListener-тэй дараах байдлаар холбоно

```
sensorManager.registerListener(mySensorListener,  
                                SensorManager.SENSOR_TRICORDER,  
                                SensorManager.SENSOR_DELAY_FASTEST);
```

шинэчлэх давтамжыг илэрхийлэх дараах  
ТОГТМОЛ утгуудтай

–SensorManager.SENSOR\_DELAY\_FASTEST – байж  
болох хамгийн хурдан нь

# Introducing the Sensor Manager

- `SensorManager.SENSOR_DELAY_GAME` – тоглоом удирдахад тохиромжтой хурд
- `SensorManager.SENSOR_DELAY_NORMAL` – энгийн хурдтай, анхдагч тохиргоо
- `SensorManager.SENSOR_DELAY_UI` – UI шинэчилэхэд тохиромжтой



# Using the Accelerometer and Compass

- Сүүлийн үеийн төхөөрөмжүүд хурдатгал болон луужинтай болсон
  - Төхөөрөмжийн одоогын чиглэлийг заах
  - Чиглэлийн өөрчлөлтийг удирдах
  - Хурдатгалыг удирдах, vertically, laterally, longitudinally – шилжилтийн хурд
- Accelerometer нь өгөгдсөн чиглэлт хэр хурдтай хөдөлсөнийг хэмжинэ





# Using the Accelerometer and Compass

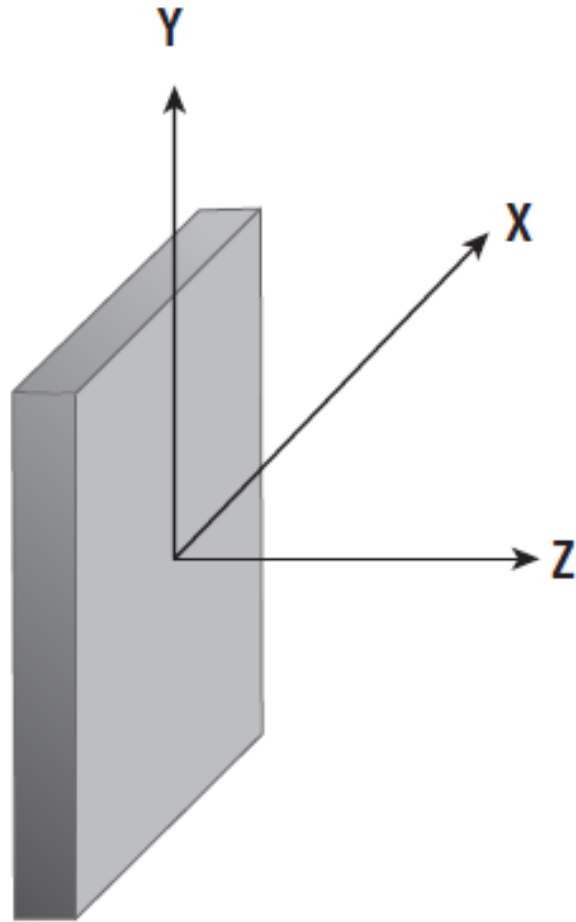
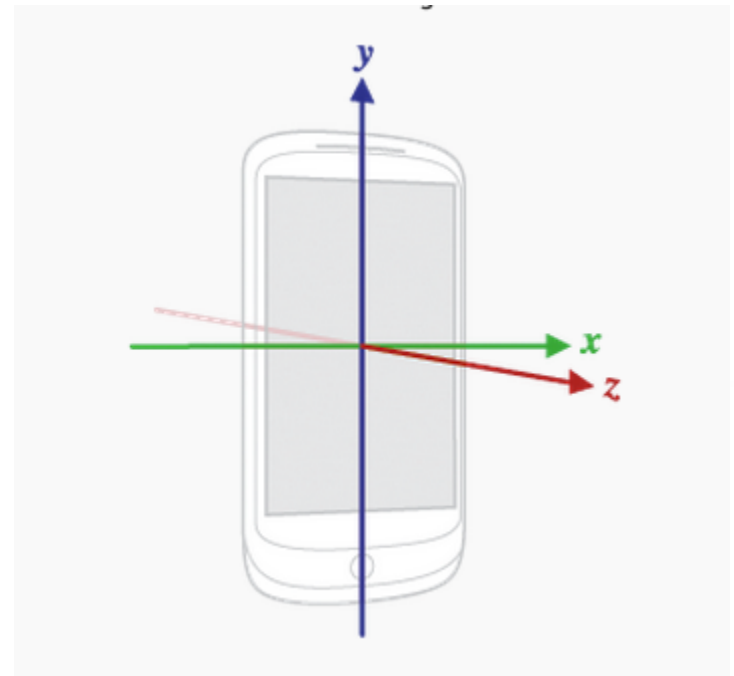


Figure 10-1



# Using the Accelerometer and Compass

```
SensorManager sm = (SensorManager) getSystemService(Context.SENSOR_SERVICE);  
sm.registerListener(mySensorListener,  
    SensorManager.SENSOR_ACCELEROMETER,  
    SensorManager.SENSOR_DELAY_UI);
```

```
SensorListener mySensorListener = new SensorListener() {  
    public void onSensorChanged(int sensor, float[] values) {  
        if (sensor == SensorManager.SENSOR_ACCELEROMETER) {  
            float xAxis = values[SensorManager.DATA_X];  
            float yAxis = values[SensorManager.DATA_Y];  
            float zAxis = values[SensorManager.DATA_Z];
```

```
            float raw_xAxis = values[SensorManager.RAW_DATA_X];  
            float raw_yAxis = values[SensorManager.RAW_DATA_Y];  
            float raw_zAxis = values[SensorManager.RAW_DATA_Z];
```

```
            // TODO apply the acceleration changes to your application.  
        }  
    }  
  
    public void onAccuracyChanged(int sensor, int accuracy) { }  
};
```

# Using the Accelerometer and Compass

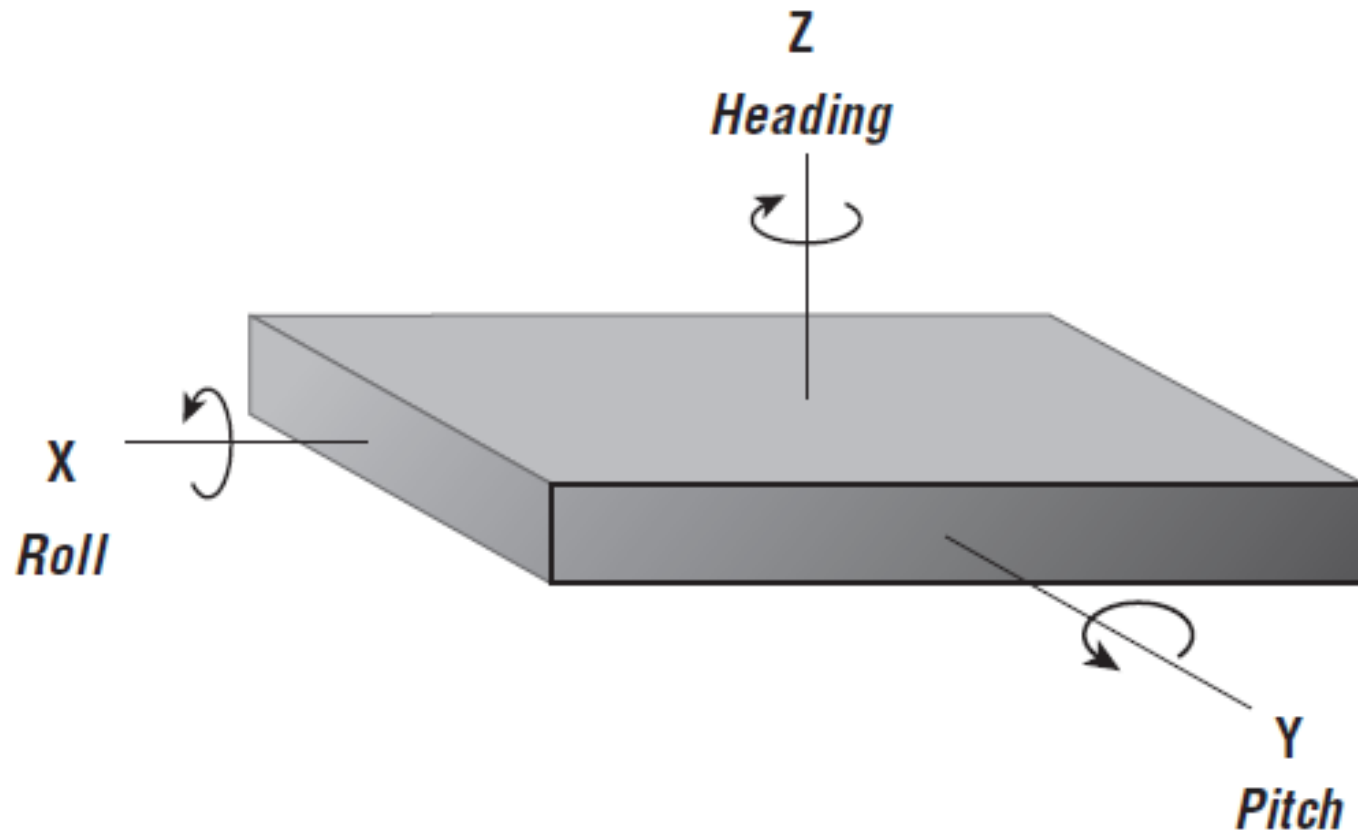


Figure 10-2

# Using the Accelerometer and Compass

- Чиглэл (Heading) – Z тэнхлэгийн дагуух эргэлт
  - 0/360 градус хойд зүг
  - 90 градус зүүн зүг
  - 180 градус урд зүг
  - 270 градус баруун зүг



# Using the Accelerometer and Compass

- Далайц (Pitch) – Х тэнхлэгийн дагуух эргэлт
  - 0 градус ар талаараа хавтгай хэвтсэн
  - -90 градус босоо (утасны дээд тал таазруу чиглэсэн)
  - -90 градус утас доошоо чиглэсэн
  - -180, -180 градус доошоо харсан



# Using the Accelerometer and Compass

- Эргэлт (Roll) – Ү тэнхлэгийн дагуух эргэлт
  - 90-ээс 90 градусын хооронд эргэлтийг дүрслэнэ
  - 0 градус нь хавтгай дээр ар талаараа байрласан
  - 90 градус утасны дэлгэц зүүн тийш харсан
  - 90 градус утасны дэлгэц баруу тийш харсан

