European Commission
TEMPUS

escuela superior ingeniería informática

Programming Mobile Applications with Android

22-26 September, Albacete, Spain

Jesus Martínez-Gómez

25 años
UCLM
UNIVERSIDAD DE CASTILLA-LA MANCHA

Programming Mobile Applications with Android

- Lesson 4.- Multimedia Elements
  - Multimedia Elements.- Introduction
  - Images.- Type of images and how to used them
  - Videos.- Different ways to load videos in our applications
  - Sounds.- Developing of a simple jukebox
  - Android Lab IV.- Create, compile and execute a multimedia application including images, sounds and videos

Programming Mobile Applications with Android

- Lesson 4.- Multimedia Elements
  - In this lesson, we will learn:
    - What are multimedia elements and their use in android applications
    - Some basics about images and how to include them
    - How to include videos
    - How to develop a simple jukebox from zeros
    - How to include the use of the camera as another tool within our applications

Programming Mobile Applications with Android

- Multimedia Elements.- Introduction
  - Multimedia definition

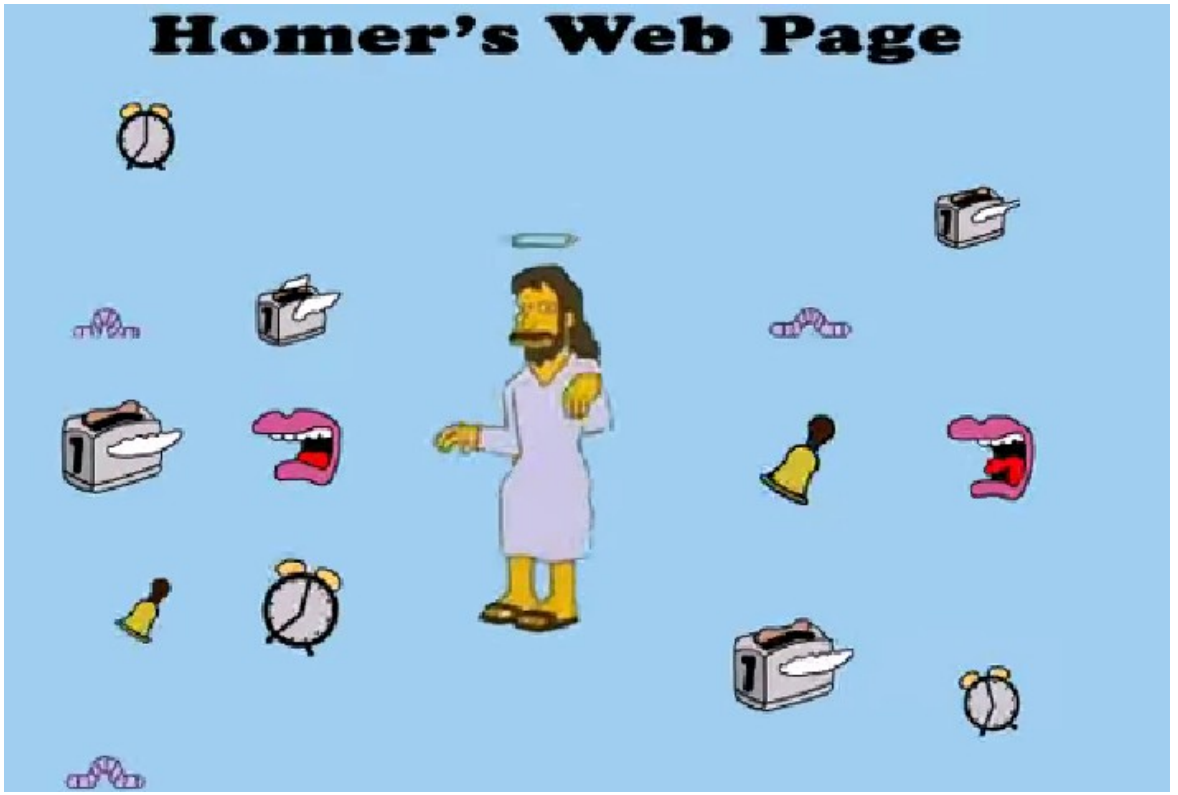
    “Relative to the use of several media simultaneously”

  - Multimedia applications can be more attractive from the user point of view but …

    https://www.youtube.com/watch?v=N7otDLzQCug

Programming Mobile Applications with Android

- Multimedia Elements.- Introduction



  - Too many multimedia elements can be negative

Programming Mobile Applications with Android

- Multimedia Elements.- Introduction
  - Android Applications → networks → payment
    - We should include videos, high quality images or sounds only when necessary
  - We have to main options
    - Use most the multimedia elements as android resources
      - The size of the application will be large but its use would not involve additional network connections
    - Download the required elements when needed
      - Small-sized application (optimal for download) but the use of the network can make users stop using it

Programming Mobile Applications with Android

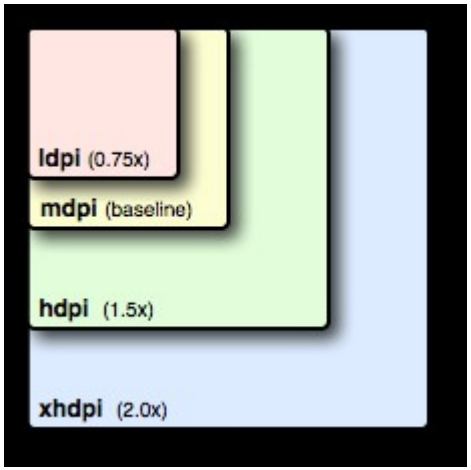
- Images.- Different type of images and how to use them.
  - Images consist of some basic parameters
    - Format (compression yes/no)
    - Density
    - Width
    - Height
    - Color Space
      - RGB, YUV, HSV, etc

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Some of the main formats are
    - BMP.- No compression
    - PNG.- Compression without loss
      - Recommended for icons
    - GIF.- Compression without loss but low resolution
    - JPEG.- Compression with loss

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Android creates different folders to store resource images in different densities
    - Ldpi.- Low density ~120 dpi
    - Mdpi.- Medium density ~160 dpi
    - Hdpi.- Hight density ~ 240 dpi
    - Xhdpi.- Extra hight density ~320 dpi



Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Layouts can also be designed for different size of screens
    - Small.- 240x320 ldpi, 320x480 mdpi, 480x800 hdpi
    - Normal.- 480x800 mdpi
    - Large.- 600x1024 mdpi
    - Extra Large.- 720x1280 mdpi, 800x1280 mdpi

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - There are several open source icon/images repositories
    - In these repositories, you can find the same image at different resolutions → Each image to the correct folder

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - There are several View elements that are visualized as images
    - ImageView
    - ImageButton
    - The background of any layout
  - Images as resources are accessed as
    - *@drawable/resName* in .xml files
    - *getResources().getDrawable(R.drawable.idRes)*

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Let's assume that we have an image called zombie.png in res/drawable folders

```
<ImageView
    android:id="@+id/imageView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/zombie" />

final ImageView imView = (ImageView)findViewById(R.id.imageViewChange);
```

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Images can also be established from two different sources:
    - Images downloaded from the Internet
    - Images acquired with the camera
    - Images loaded from the gallery

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from Internet
    - Previous.- Set Internet permissions in the manifest file

      <uses-permission android:name="android.permission.INTERNET" />
    - Steps
      - Create a URL connection with URL of the image
      - Encode the obtained image using BitMapFactory and establish it to a View Element
    - This cannot be done in the main Thread → use Asynchronous tasks

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from Internet
    - Create a new class that extends from AsyncTask<String,Void,Bitmap>
      - Two methods implemented:
        - Bitmap doInBackGround(String … params)

```
InputStream in = new java.net.URL(params[0]).openStream();
Bitmap myBitmap = BitmapFactory.decodeStream(in);
```

        - void onPostExecute(Bitmap result)

```
myImageView.setImageBitmap(result);
```

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from Internet
    - Create a new class for download
      - Constructor()
        - It should include a reference to the ImageView object
      - Bitmap doInBackGround(String … params)
        - This method is firstly executed in a separate thread and generates as result a Bitmap object
      - void onPostExecute(Bitmap result)
        - As soon as the doInBackGroud method ends, this method is invoked with the Bitmap generated as result

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from Internet
    - Create a new class for download
    - In our code, we need to create a new object of the new class and then call the execute method

new Download((ImageView) findViewById(R.id.imageView1)).execute("url.jpg");

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from camera
    - Previous.- Set permissions in the manifest file

```
<uses-permission android:name="android.permission.CAMERA" />
<uses-feature android:name="android.hardware.camera" />
<uses-feature android:name="android.hardware.camera.autofocus" />
```

    - Steps
      - Launch a new activity in charge of opening the camera
      - Receive the result and encode it as image

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from camera

```
private static final int TAKE_PHOTO = 1;
Intent cameraIntent = new Intent(
android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
startActivityForResult(camaraIntent, TAKE_PHOTO);
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
if (requestCode == TAKE_PHOTO) {
Bitmap image1 = (Bitmap) data.getExtras().get("data");
ImageView iv_foto = (ImageView) findViewById(R.id.imgViewId);
iv_foto.setImageBitmap(image1);
```

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them.
  - Load images from gallery
    - Previous.- Set permissions in the manifest file

      <uses-permission

      android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    - Steps
      - Launch a new activity in charge of opening the gallery
      - Receive the result and encode it as image

Programming Mobile Applications with Android

- Images.- Different type of images and how to use them. Load images from gallery

```
private int SELECT_IMAGE = 237487;
Intent intent = new Intent(Intent.ACTION_PICK,
android.provider.MediaStore.Images.Media.INTERNAL_CONTENT_URI);
intent.setType("image/*");
startActivityForResult(intent, SELECT_IMAGE);
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
if (resultCode == Activity.RESULT_OK){
Uri selectedImage = data.getData();
ImageView iv_foto = (ImageView) findViewById(R.id.imageView1);
iv_foto.setImageURI(selectedImage); } }
```

Programming Mobile Applications with Android

- Videos.- Loading videos in our applications.
  - The use of videos in our applications can improve the final result, but be careful with the size
  - There are several ways to include them
    - Video Resource → VideoView
    - Load Video from local storage
    - Load Videos from YouTube

Programming Mobile Applications with Android

- Videos.- Loading videos in our applications.
  - Videos as Resources
    - Copy the video to the /res/raw folder
    - Include a VideoView in the layout of our applications
    - Now in the .java activity files
      - Get the reference of the VideoView object
      - Get the reference to the video resource
      - Associate both references
      - Start playing the video

Programming Mobile Applications with Android

- Videos.- Loading videos in our applications.
  - Videos as Resources

```
VideoView videoToPlay;
videoToPlay = (VideoView)findViewById(R.id.videoViewRes);
Uri path = Uri.parse("android.resource://"+getPackageName()
+"/"+R.raw.sample_video);
videoToPlay.setVideoURI(path);
videoToPlay.start();
```

Programming Mobile Applications with Android

- Videos.- Loading videos in our applications.
  - Videos from local storage
    - Permissions

      <uses-permission android:name="android.permission.STORAGE">
    - Now in the .java activity files
      - Get the reference of the VideoView object
      - Associate the video path

      videoToPlay.setVideoPath("/sdcard/videoName.mp4");
    - The VideoView elements could be improved by adding control buttons (play, stop, pause, etc)

Programming Mobile Applications with Android

- Videos.- Loading videos in our applications.
  - Videos from Youtube
    - Youtube is the highest collection of videos
      - Music, Cinema, etc
    - The easiest way to load videos from youtube is to start a new activity in charge of visualizing the video

      startActivity(new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.youtube.com/watch?v=h_L4Rixya64")));

    - We then can open the video with
      - Any browser ready for that
      - Youtube application

Programming Mobile Applications with Android

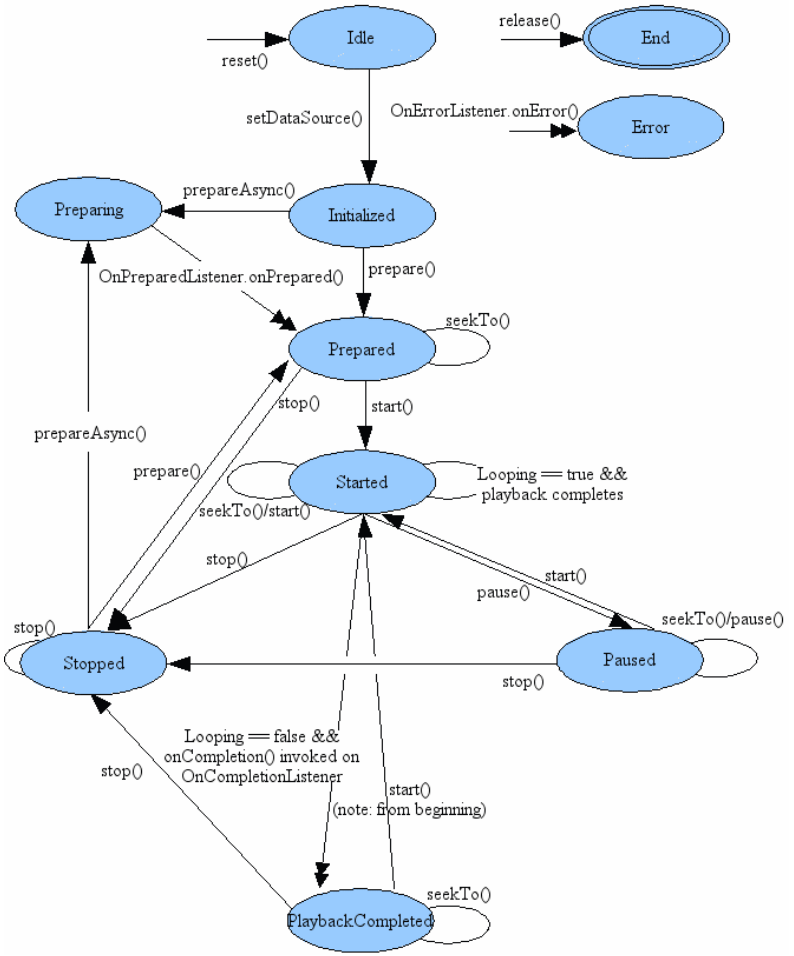
- Sounds.- Development of a simple jukebox
  - The use of sounds can also improve the acceptance of our application when used properly
    - The should be played only if the user explicitly wants
  - Sounds are characterized by
    - Quality
    - Size
    - ...

Programming Mobile Applications with Android

- Sounds.- Development of a simple jukebox
  - The easiest way to include sounds is the use of the MediaPlayer class
    - Main methods
      - Play()
      - Stop()
      - Pause()

Programming Mobile Applications with Android

- Sounds.- Development of a simple jukebox



Programming Mobile Applications with Android

- Sounds.- Development of a simple jukebox
  - Use of the MediaPlayer class, steps:
    - Move all the sounds to the /res/raw folder
    - Create a MediaPlayer object from the desired sound resource file
    - Play and Stop the sound depending on the user actions

```
MediaPlayer mediaPlayer = MediaPlayer.create(this, R.raw.sample_song);
mediaPlayer.start();
mediaPlayer.stop();
mediaPlayer.start();
```

Programming Mobile Applications with Android

- Lesson 4.- Visual Interfaces
  - Android Lab IV.- Create, compile and execute a multimedia application including images, sound and videos
    - Follow the instructions to create a multimedia application with controls to include new images and play videos and sounds

European Commission
TEMPUS

European Commission
TEMPUS

escuela superior ingeniería informática

Programming Mobile Applications with Android

22-26 September, Albacete, Spain

Jesus Martínez-Gómez

25 años
UCLM
UNIVERSIDAD DE CASTILLA-LA MANCHA