

# Super-Charge Your Mobile Game with YouTube

Integrating YouTube API V3 Uploads and Video Playback with Unity 3D

Jarek Wilkiewicz, Developer Advocate, YouTube  
Corey Johnson, Field Engineer, Unity Technologies  
v01.00 (5/14/2013)



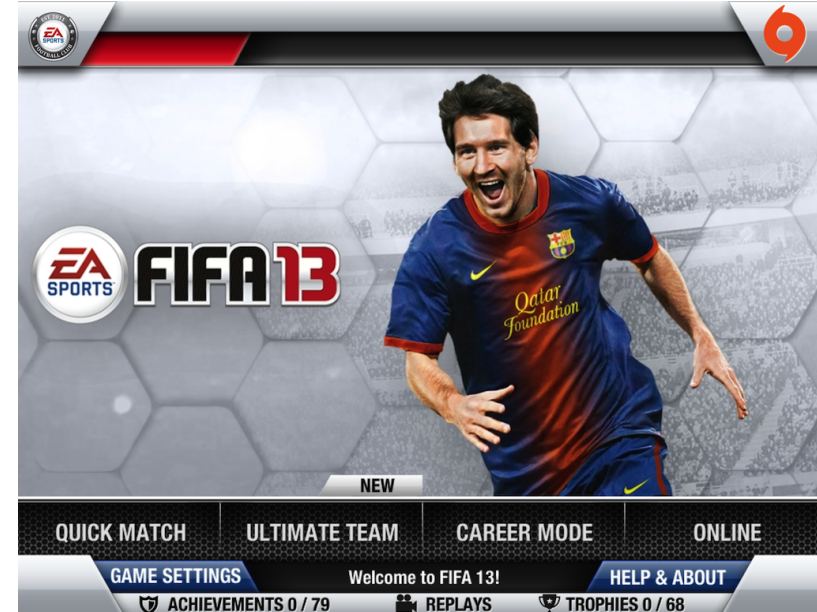


Credit: Dan Zen, <http://www.flickr.com/photos/danzen/4387877247/sizes/o/>



# More Users and Better Engagement

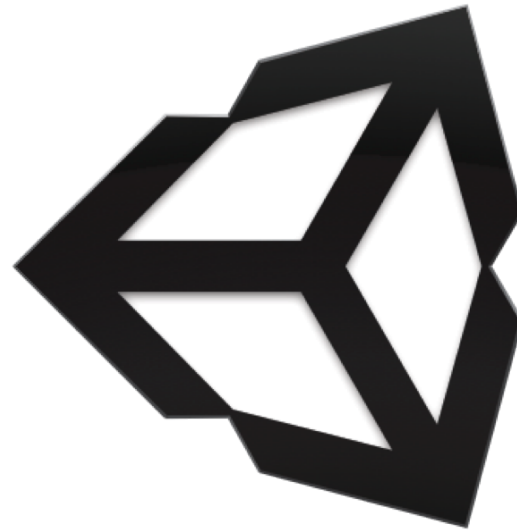
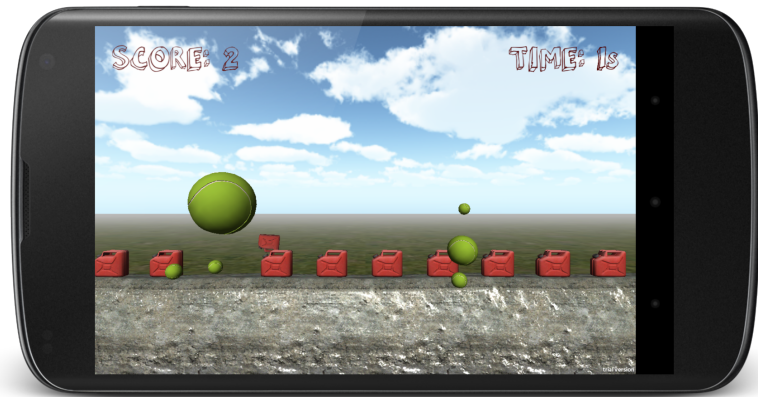
95%<sup>[1]</sup>



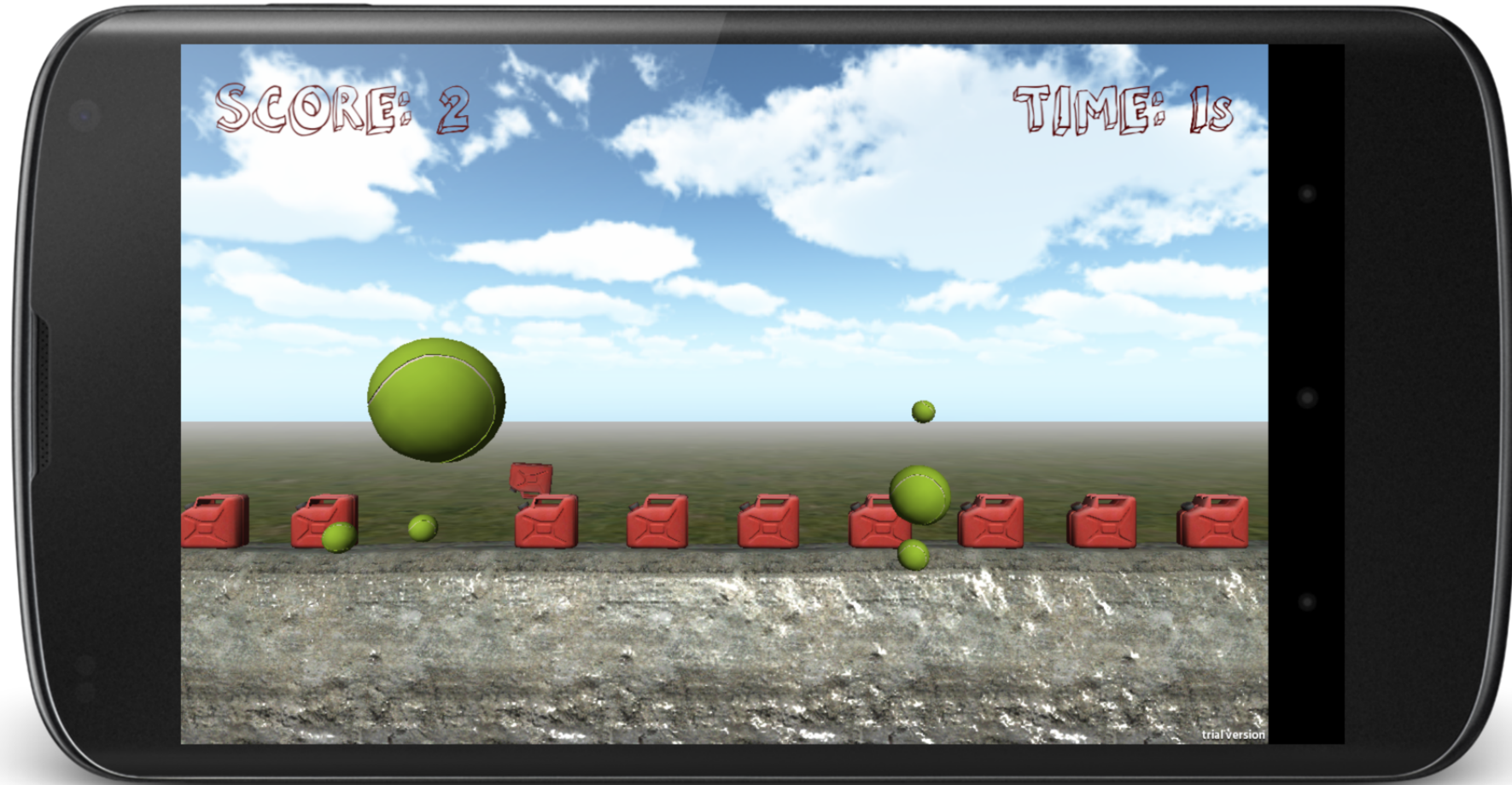
[1] Source: Blitz Pulling the Trigger to Purchase, 2011. Base: 1,000 respondents, age 13–35 who play video games 7+ hours per week.



# Agenda



# The Game





# Let's Build It!

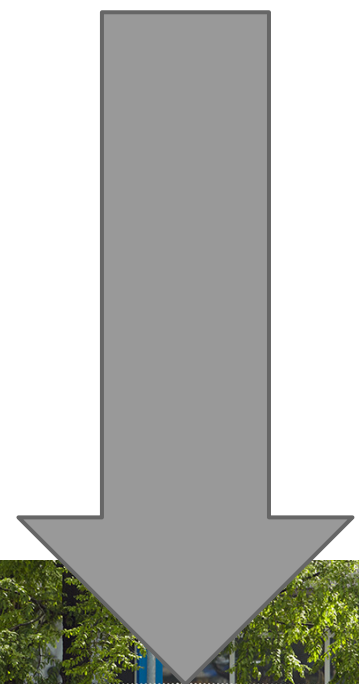
Mobile Game Development with Unity





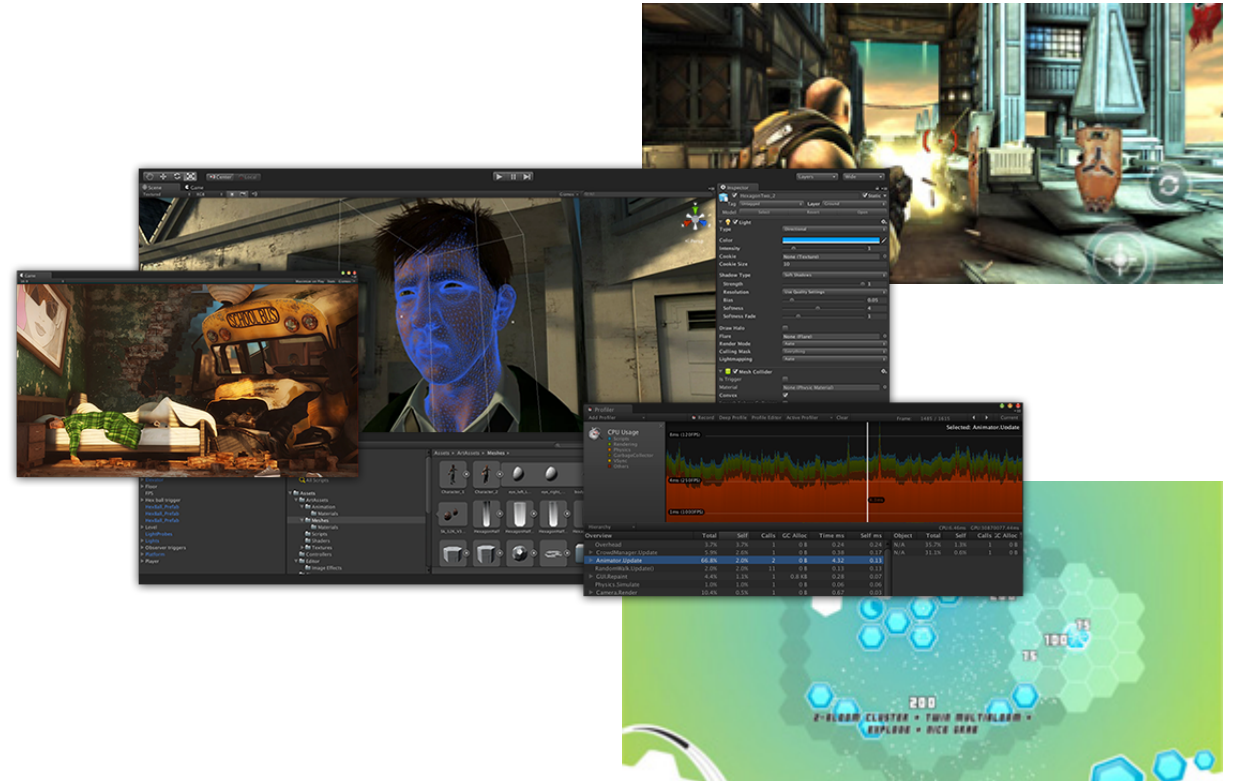
# Who?!

Corey Johnson  
Field Engineer  
Unity Technologies



# What is Unity?

- A tool to create 2D/3D content
  - Rapid Learning Curve & Iteration
- Build Once, Deploy Everywhere
  - Multi Platform Engine
- Mission: Democratize Game Development
  - Community



# Where is Everywhere?

- Desktop
  - Window, Mac, Linux, Web Player
- Mobile
  - Android, iOS, Windows Phone 8, Blackberry
- Console
  - Xbox 360, Playstation 4/Vita, Wii U
- Union
  - Distribution to new platforms





# Community

- 1,800,000 developers
- 400,000 monthly active
- 5,000,000 hours of creation monthly
- Asset Store
  - Developer Marketplace



# Plugins

## Native Code from Game Scripts\*

- Game scripts written in C#, Javascript, or Boo
  - Mono under the hood
  - Script *doesn't* mean interpreted
    - Unity scripts are compiled to native bytecode
- Native code can vary based on platform
  - Android: C/C++ or Java
  - iOS: Objective-C
- Focusing on Android for this talk

\*Unity Pro feature



# Plugins - Native Code Example

Build a shared object (.so) using the Android NDK

```
//C Plugin Code of a Minimal Plugin:  
extern "C" {  
    float FooPluginFunction () { return 5.0F; }  
}
```

C

Place in Unity project's `Assets/Plugins/Android` folder





# Plugins - Native Code Example

Call the Native Code

```
//C# Game Code
using UnityEngine;
using System.Runtime.InteropServices;

class SomeScript : MonoBehaviour {

    [DllImport ("PluginName")]
    private static extern float FooPluginFunction ();

    void Awake () {
        // Calls the FooPluginFunction inside the plugin and prints 5 to the console
        print (FooPluginFunction ());
    }
}
```

C#



# Plugins - Java Code

- Build a Java archive file (.jar) using Eclipse and the ADT
  - Place in Unity project's `Assets/Plugins/Android` folder
- Uses the Java Native Interface to interact with native code
- Unity provides helper classes
  - For raw JNI handling there's `AndroidJNI` and `AndroidJNIHelper`
  - For a higher level of automation and caching there's `AndroidJavaObject` and `AndroidJavaClass`



# Plugins - Java Code

C#

```
//The comments describe what you would need to do if you were using raw JNI
AndroidJavaObject jo = new AndroidJavaObject("java.lang.String", "some_string");
// jni.FindClass("java.lang.String");
// jni.GetMethodID(classID, "<init>", "(Ljava/lang/String;)V");
// jni.NewStringUTF("some_string");
// jni.NewObject(classID, methodID, javaString);
int hash = jo.Call<int>("hashCode");
// jni.GetMethodID(classID, "hashCode", "()I");
// jni.CallIntMethod(objectID, methodID);
```



# Plugins

## Pro Tips

- AndroidManifest.xml
  - Update with new activities and added permissions, etc.
  - Place in `Assets/Plugins/Android` folder
    - Unity will automatically use it instead of generating a default one
- Create a wrapper script
  - Native code needs annotations and Java needs additional code
  - Makes a nicer API for users
- Marshaling data
  - Expensive! Both time and memory
  - Weigh need vs cost





# Getting Data for Video

- Video - `OnRenderImage`
  - Called after all rendering is complete
  - Rendertarget dependant
- Audio - `OnAudioFilterRead`
  - Inserts custom DSP filter
  - Called every ~20ms depending on your samplerate
- `WaitForEndOfFrame`
  - Yields until all rendering is complete, just before final frame buffer rendered to screen



# Editor Walkthrough



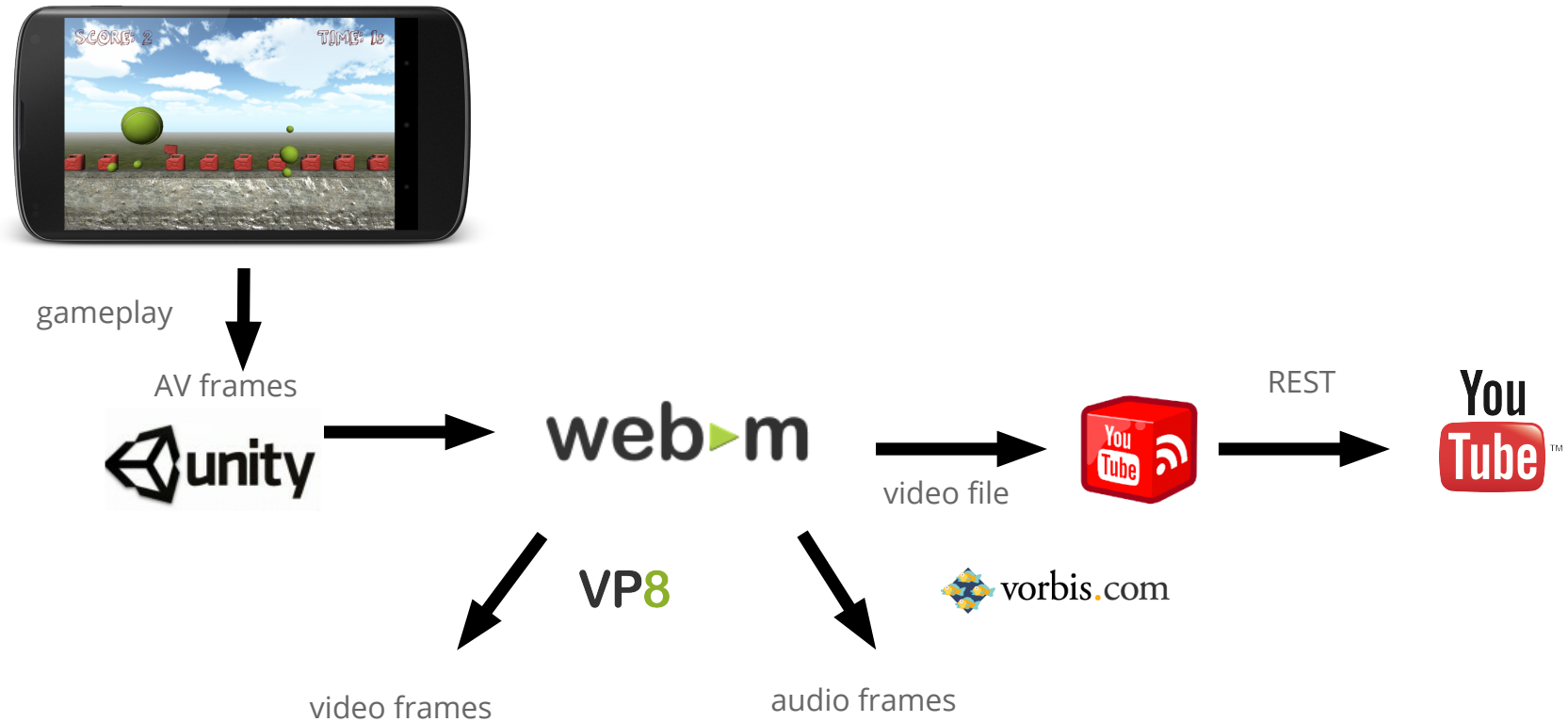


# Uploads

WebM + YouTube Data API

# End-to-End Upload Flow

## WebM/VP8+Vorbis Encoding







# Unity Plugin

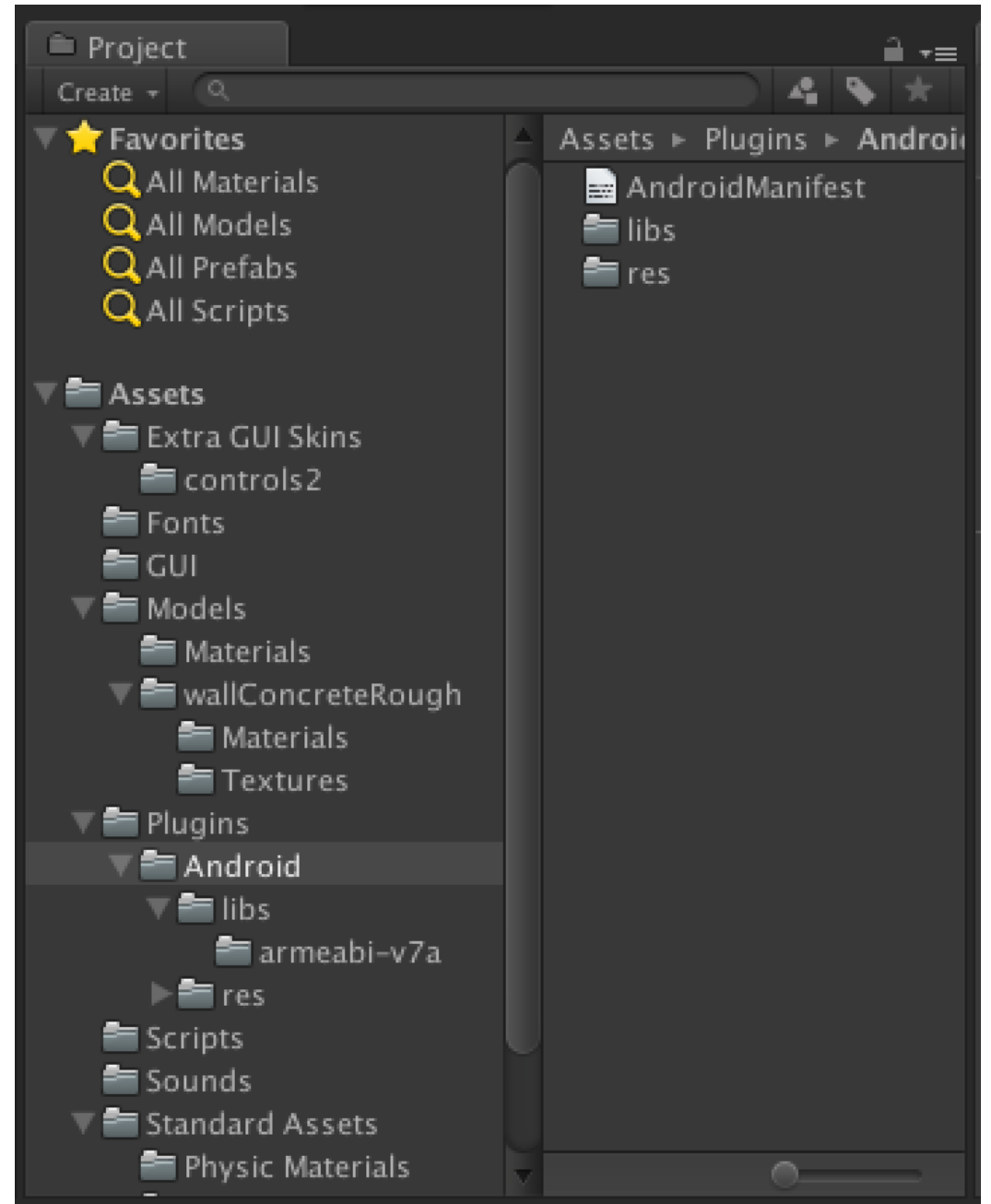
Example

# Start with the Unity Plugin

A few hints

- AndroidManifest.xml
  - find it in Temp/StagingArea first
- Resource merge (res/ directory)
  - Sorry, no R.id for you

```
int buttonId = getResources().  
getIdentifier("submitButton", "id",  
getPackageName());
```

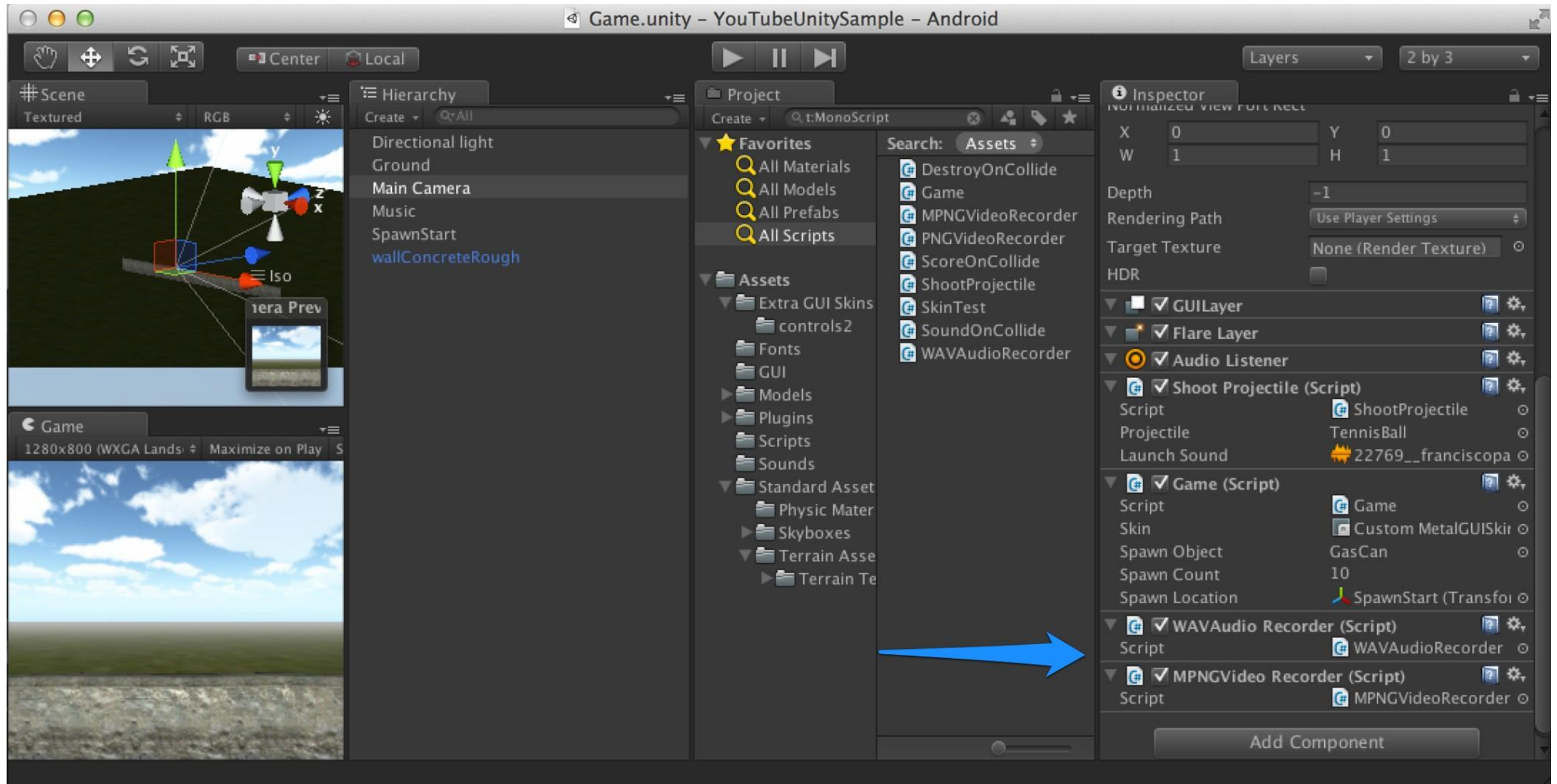




# Audio and Video Capture

`OnAudioFilterRead`, `WaitForEndOfFrame`, `glReadPixels`

# Audio and Video Capture





# Unity Configuration

## Audio Mixer Rate and Video Capture

- Audio (24 kHz for our demo)
  - `AudioSettings.outputSampleRate`
- Video (10 FPS for our demo)
  - `Time.captureFramerate`
    - Don't confuse with
      - `Application.targetFrameRate`



# Audio Recording

OnAudioFilterRead (PCM)

C#

```
void OnAudioFilterRead (float[] data, int channels) {  
    lock (this) {  
        if(Active) {  
            ConvertAndWrite(data);  
        }  
    }  
}
```



# Video Recording

## WaitForEndOfFrame

C#

```
StartCoroutine(CaptureVideo());

IEnumerator CaptureVideo() {
    Time.captureFramerate = captureFramerate;
    while (Active) {
        yield return new WaitForEndOfFrame();
        captureVideoFrame();
        [...]
    }
}
```



# Video Recording

## Texture2D.ReadPixels

C#

```
Texture2D tex= new Texture2D( captureWidth, captureHeight,  
                             TextureFormat.RGB24, false );  
  
// Read screen contents into the texture  
tex.ReadPixels(new Rect(0, 0, captureWidth, captureHeight), 0, 0);  
tex.Apply();  
// Encode texture into PNG  
bytes = tex.EncodeToPNG();
```





# Video Recording

## glReadPixels

Java

```
private byte[] glReadPixels() {  
    GL10 gl = getGL();  
    framebuffer.rewind();  
    gl.glReadPixels(0, 0, encoder.getWidth(), encoder.getHeight(),  
    GL10.GL_RGBA, // <-- RGB, we need YUV  
    GL10.GL_UNSIGNED_BYTE, framebuffer);  
    return framebuffer.array();  
}
```



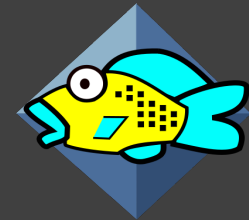


# Encoding

WebM/VP8/Ogg Vorbis

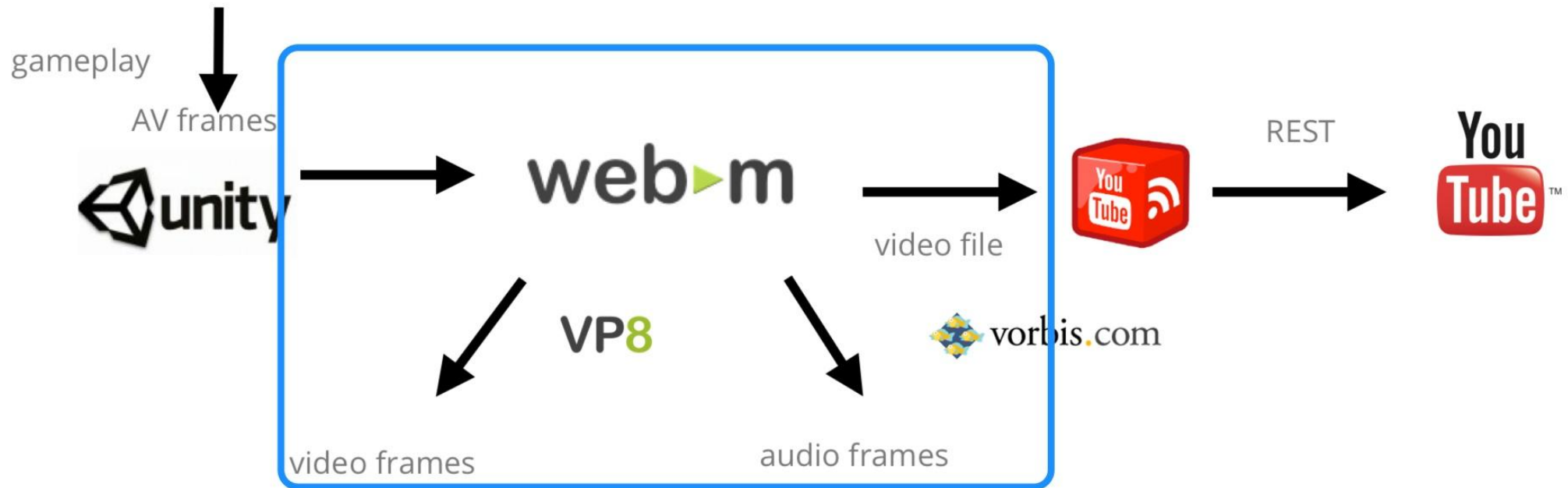
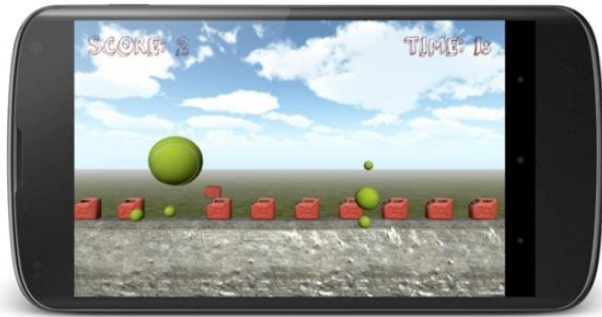
web▶m

VP8



# Encoding

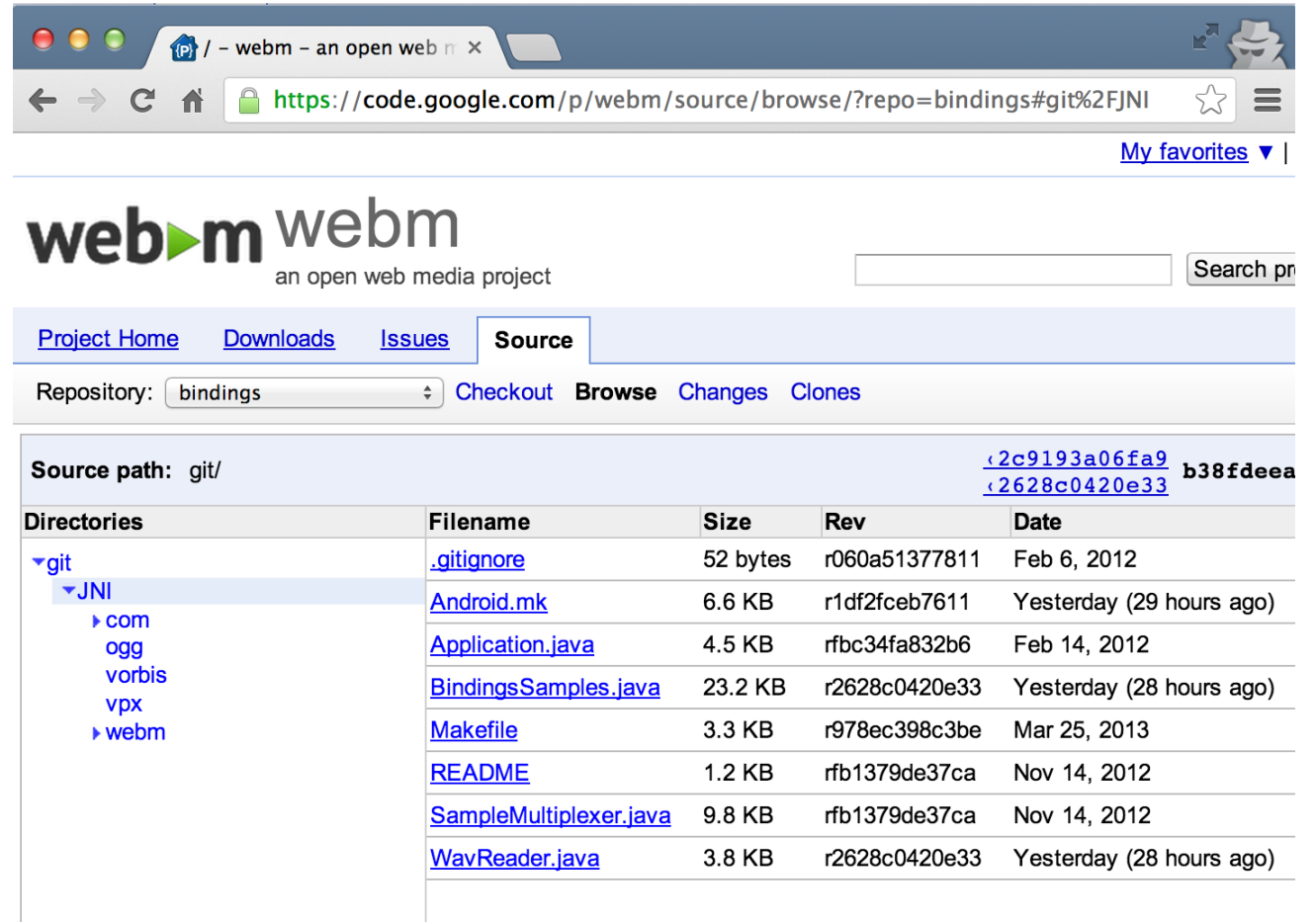
WebM/VP8/Vorbis



# WebM Bindings Project

Open Source JNI Bindings for WebM, VP8 and Vorbis

- Get it from [code.google.com/p/webm/source/browse/?repo=bindings](https://code.google.com/p/webm/source/browse/?repo=bindings)
- Depends on
  - Android NDK
  - libwebm
  - libvpx
  - libvorbis
  - libogg
  - libyuv (optional)



The screenshot shows a web browser window displaying the source code for the 'bindings' repository of the 'webm' project. The page title is 'webm webm an open web media project'. The navigation menu includes 'Project Home', 'Downloads', 'Issues', and 'Source'. The 'Source' tab is active, showing a repository dropdown set to 'bindings' and buttons for 'Checkout', 'Browse', 'Changes', and 'Clones'. The source path is 'git/'. A table lists the files and directories in the 'JNI' directory:

Directories	Filename	Size	Rev	Date
git	<a href="#">.gitignore</a>	52 bytes	r060a51377811	Feb 6, 2012
JNI	<a href="#">Android.mk</a>	6.6 KB	r1df2fceb7611	Yesterday (29 hours ago)
com	<a href="#">Application.java</a>	4.5 KB	rabc34fa832b6	Feb 14, 2012
ogg	<a href="#">BindingsSamples.java</a>	23.2 KB	r2628c0420e33	Yesterday (28 hours ago)
vorbis	<a href="#">Makefile</a>	3.3 KB	r978ec398c3be	Mar 25, 2013
vpx	<a href="#">README</a>	1.2 KB	rfb1379de37ca	Nov 14, 2012
webm	<a href="#">SampleMultiplexer.java</a>	9.8 KB	rfb1379de37ca	Nov 14, 2012
	<a href="#">WavReader.java</a>	3.8 KB	r2628c0420e33	Yesterday (28 hours ago)



# Encoding and Muxing

## Audio and Video Encoders

Java

```
VorbisEncoderConfig vorbisConfig;           // Audio encoder
VorbisEncoderWrapper vorbisEncoder;

LibVpxEncConfig vpxConfig;                   // Video encoder
LibVpxEnc vpxEncoder;

Segment muxerSegment;                        // Muxer
MkvWriter mkvWriter;
```



# Encoding and Muxing

## Inputs and Outputs

Java

```
byte[] pcmArray; // Audio (PCM)

Bitmap bitmap; // Video (RGB)
byte[] rawVideoArray; // Video (YUV)

AudioFrame audioFrame; // Encoded audio (Vorbis)
ArrayList<VpxCodecCxPkt> encPkt; // Encoded video (VP8)
```





# Encoding and Muxing

Encode, then Combine Audio and Video Streams

```
Bitmap bitmap = mpngReader.getNextRGBBitmap();
pcmArray = wavReader.readSamples(samplesToRead);
vorbisEncoder.encodeAudio(pcmArray);
audioFrame = vorbisEncoder.readCompressedAudio();
rawVideoArray = mpngReader.convertToYUV(bitmap);
encPkt = vpxEncoder.encodeFrame(rawVideoArray, frameStart,
                                nextFrameStart - frameStart);

muxerSegment.addFrame(pkt.buffer, newVideoTrackNumber,
                      pkt.pts, isKey);
muxerSegment.addFrame(audioFrame.frame.array(),
                      newAudioTrackNumber, audioFrame.timestamp,
                      true);
```

Java



# WebM Bindings Classes

Object-Oriented Video Encoding



```
com.google.libvorbis.VorbisEncoderConfig  
com.google.libvpx.LibVpxEncConfig
```

**<- Config**

```
com.google.libvorbis.VorbisEncoderWrapper  
com.google.libvpx.LibVpxEnc
```

**<- Encoders**

```
com.google.libwebm.mkvmuxer.AudioTrack  
com.google.libwebm.mkvmuxer.VideoTrack
```

**<- Tracks**

```
com.google.libwebm.mkvmuxer.MkvWriter  
com.google.libwebm.mkvmuxer.Segment
```

**<- Muxer**





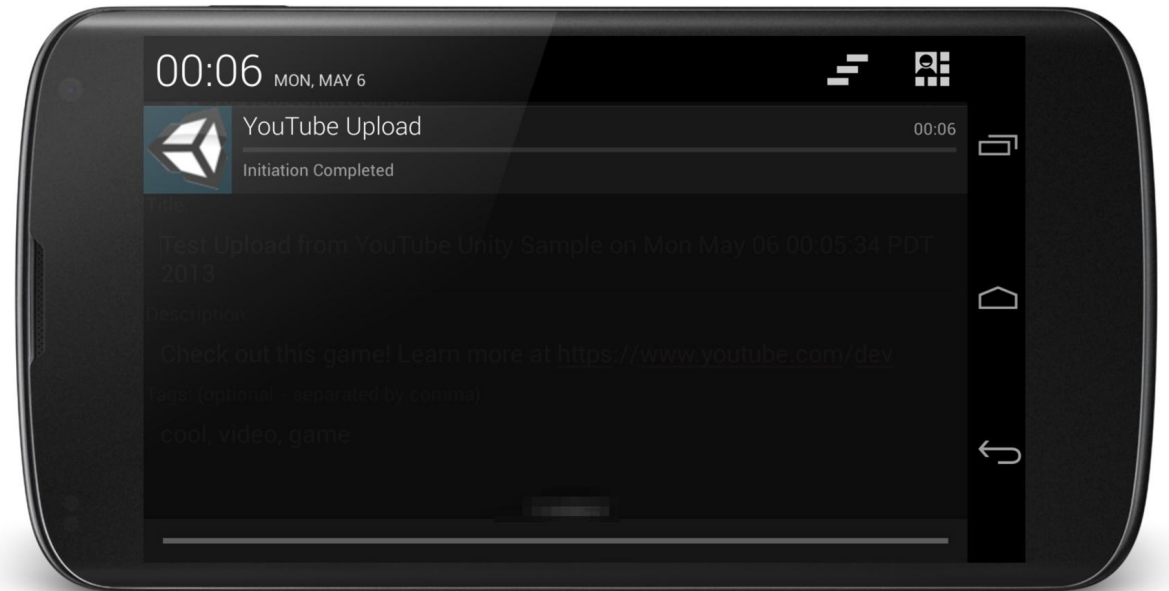
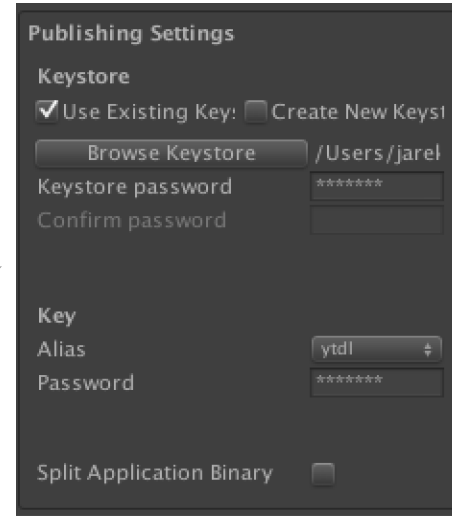
# Upload

The YouTube API

# YouTube Data API V3

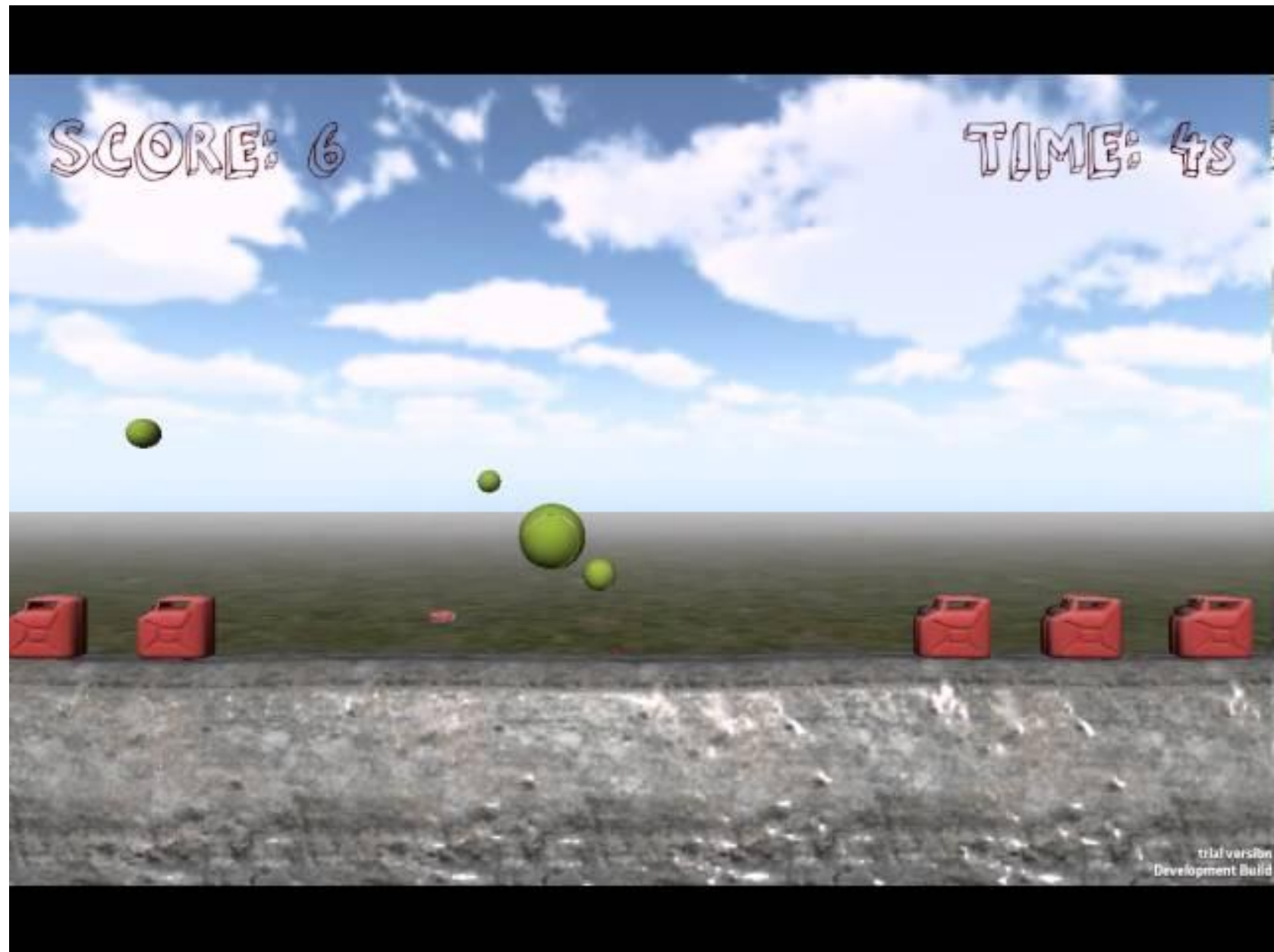
## Java Client Library

- Register Client ID at [code.google.com/apis/console](https://code.google.com/apis/console)
- Sign your app with the same key configured in the API console (SHA1 cert fingerprint)
- Use `IntentService` to handle uploads asynchronously
- Post updates using the `NotificationManager`
- Use resumable uploads uploader.  
`setDirectUploadEnabled(false)`



# Example

VP8+Vorbis





# Auth

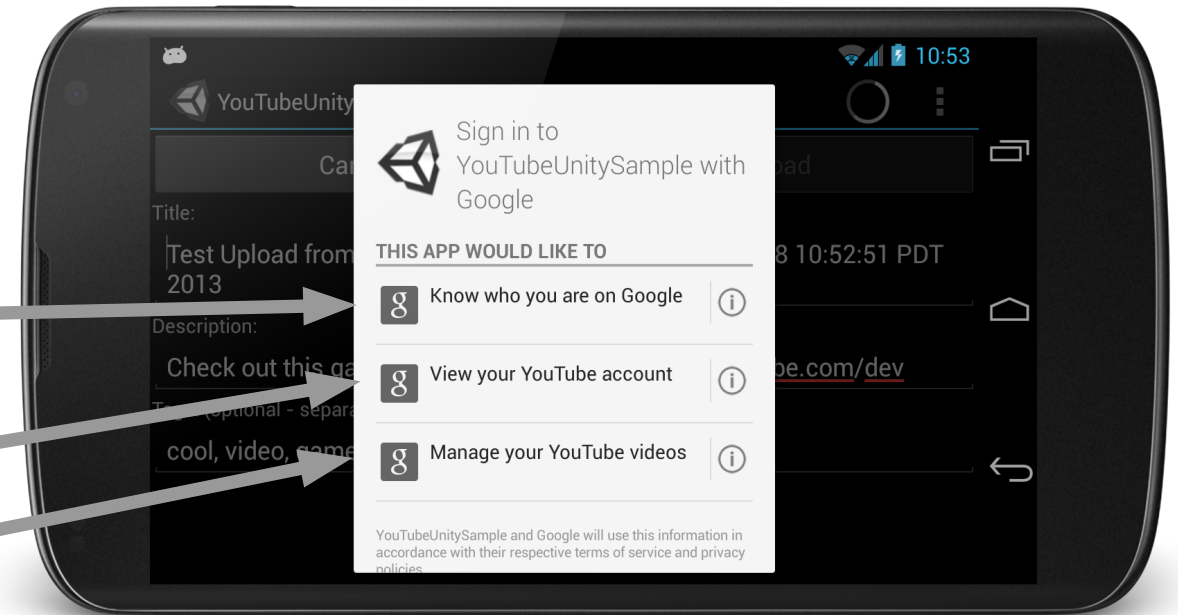
Google Play Services: Authorization

# Authorization

## GoogleAuthUtil

Java

```
mToken = GoogleAuthUtil.getToken(  
    UploadActivity.this,  
    mChosenAccountName,  
    "oauth2:" +  
    Scopes.PLUS_PROFILE +  
    " " +  
    YouTubeScopes.YOUTUBE_READONLY +  
    " " +  
    YouTubeScopes.YOUTUBE_UPLOAD);
```





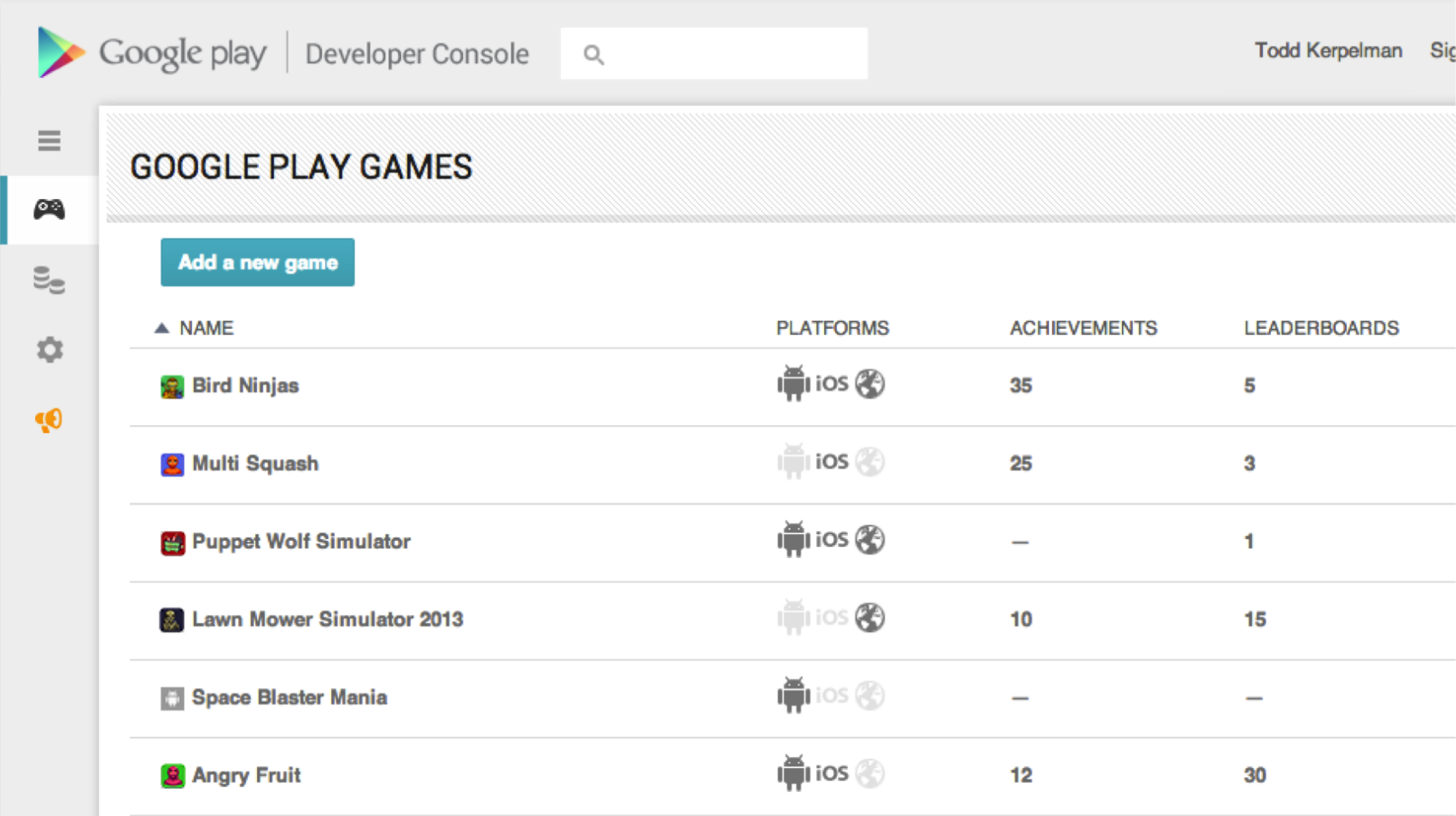


# Playback

























YouTube Android Player API

# Achievements and Leaderboards

## Google Play Games



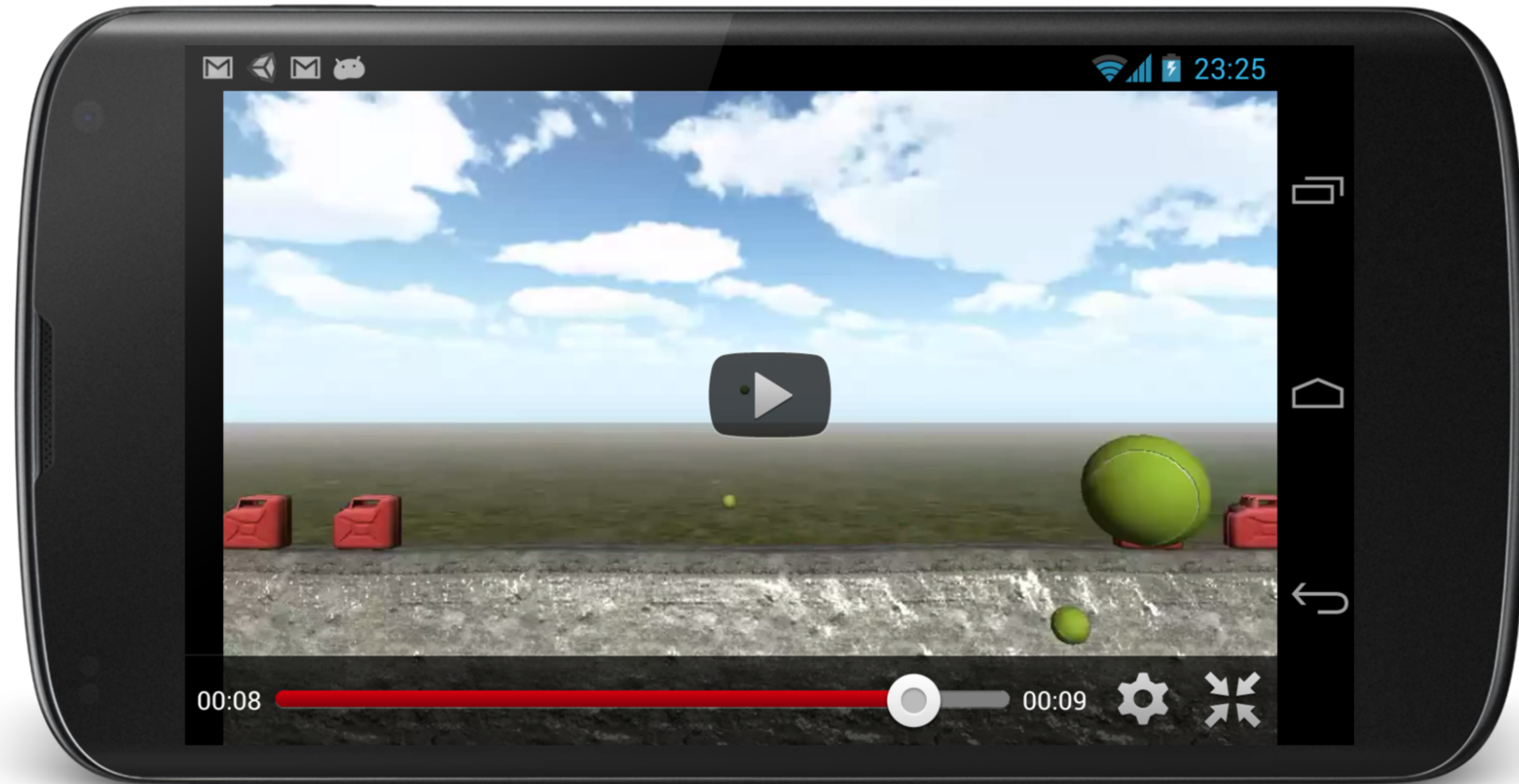
The screenshot shows the Google Play Games Developer Console interface. At the top, there is a search bar and the user's name, Todd Kerpelman. The main heading is "GOOGLE PLAY GAMES". Below this, there is a button labeled "Add a new game". A table lists several games with their respective platforms, achievements, and leaderboards.

NAME	PLATFORMS	ACHIEVEMENTS	LEADERBOARDS
 Bird Ninjas	  	35	5
 Multi Squash	  	25	3
 Puppet Wolf Simulator	  	—	1
 Lawn Mower Simulator 2013	  	10	15
 Space Blaster Mania	  	—	—
 Angry Fruit	  	12	30



# YouTube Android Player API

In-game Video Playback

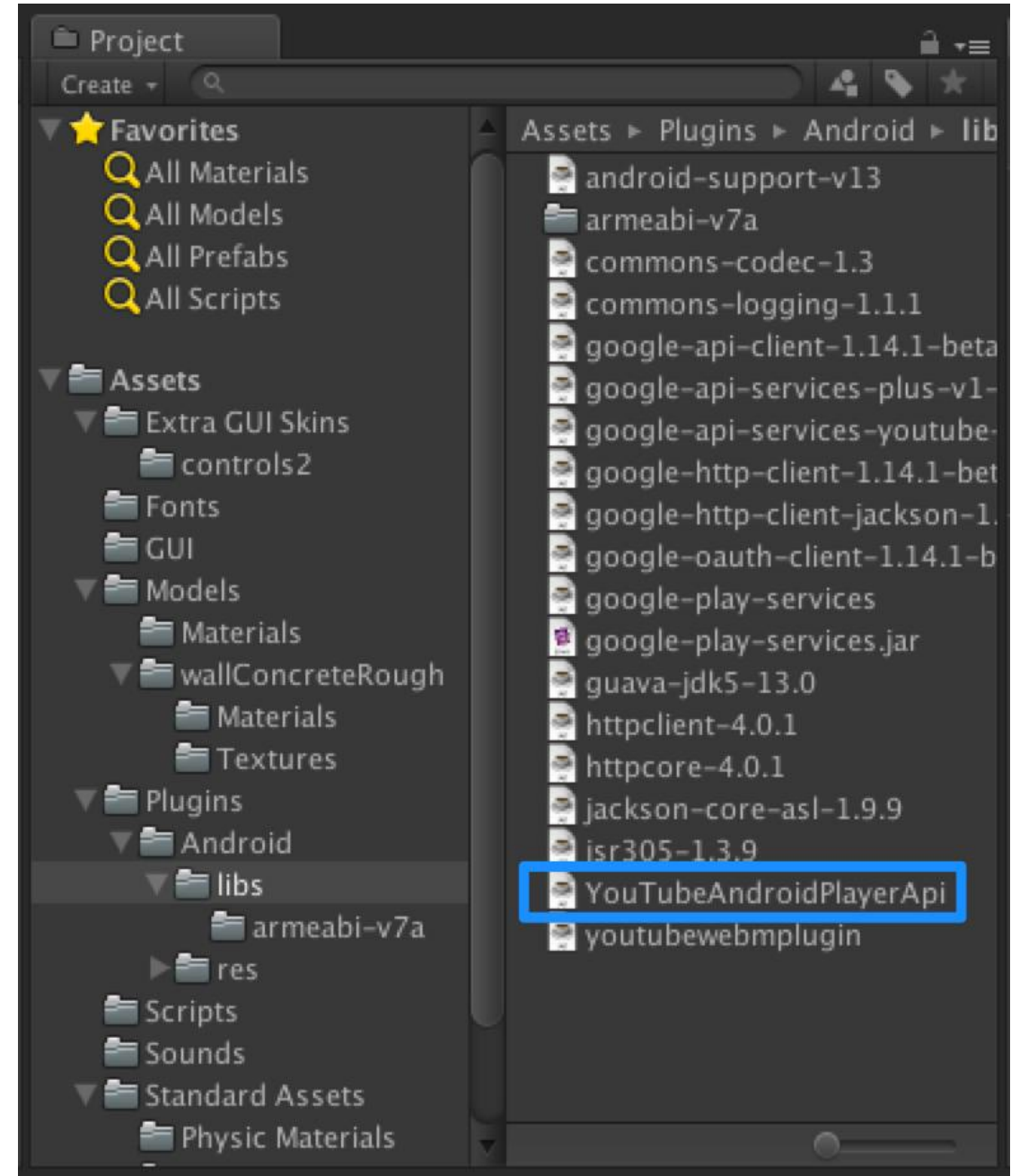


# YouTube Android Player API

## A Few Hints

- Need "Key for Android apps (with certificates)", the APK cert won't do
- May request YouTube app update due to `SERVICE_VERSION_UPDATE_REQUIRED`
- Video may not be ready for playback immediately after upload, use

```
processingDetails.processingProgress
```



# Developer Resources

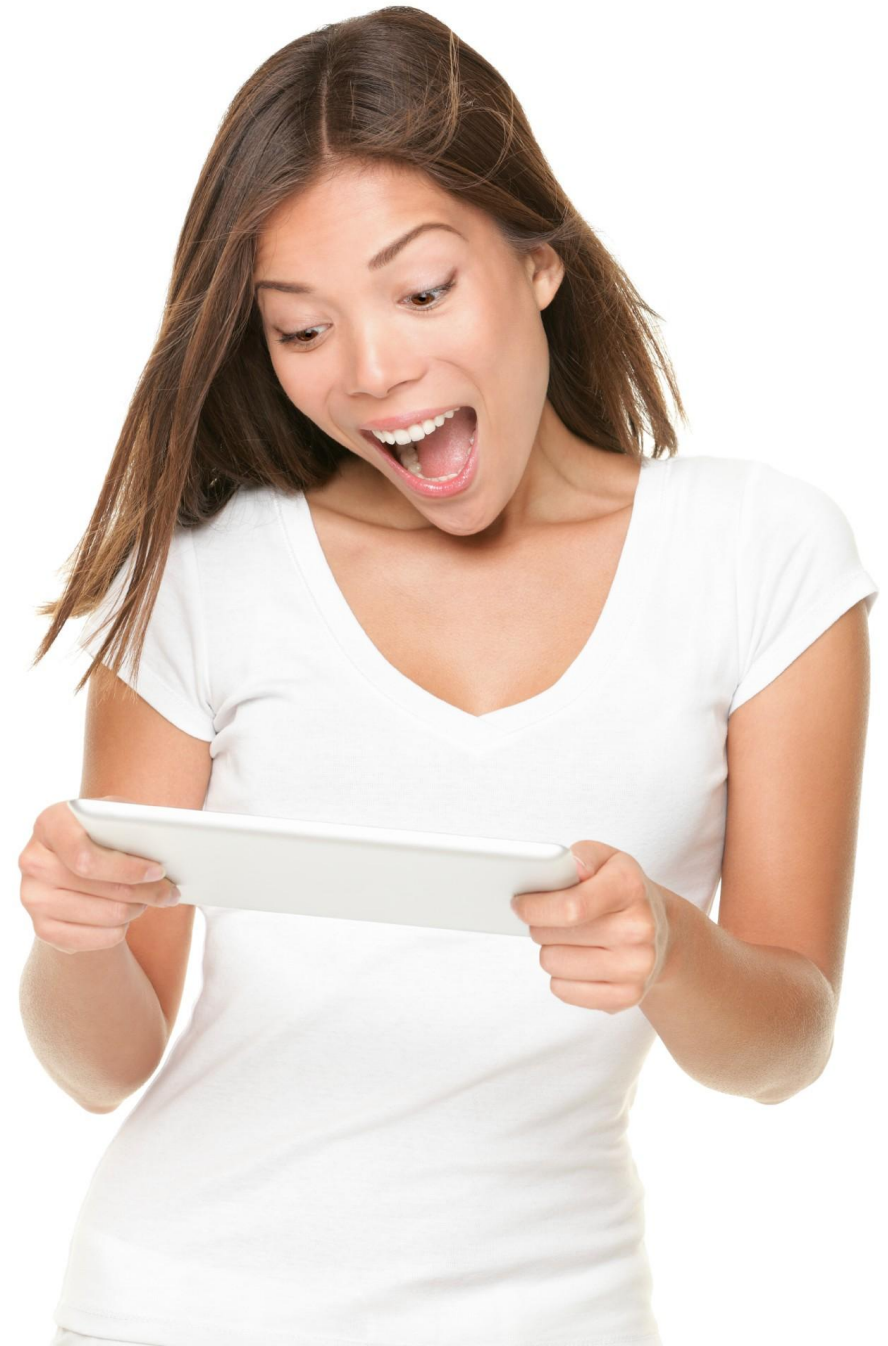
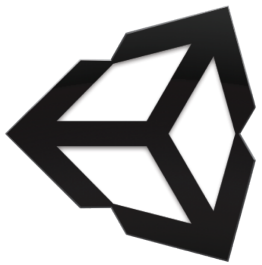
A Few Links

[youtube.com/dev](https://youtube.com/dev)

[code.google.com/p/webm/source/browse/?repo=bindings](https://code.google.com/p/webm/source/browse/?repo=bindings)



# Questions?



<Thank You!>

[youtube.com/dev](https://youtube.com/dev)

+YouTubeDev

@YouTubeDev

[google.com/+JarekWilkieiwcz](https://google.com/+JarekWilkieiwcz)

@wjarek







Google  
Developers

# Bullet Slide With Subtitle Placeholder

## Subtitle Placeholder

- Titles are formatted as Open Sans with bold applied and font size is set at 30pts
  - Vertical position for title is .3”
  - Vertical position for subtitle is 1.1”
  - Vertical position for bullet text is 2”
- Title capitalization is title case
- Subtitle capitalization is title case
- Titles and subtitles should never have a period at the end



# Color Palette

Flat Color



Secondary



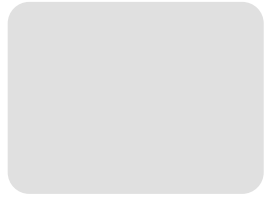
Gradient



Grays



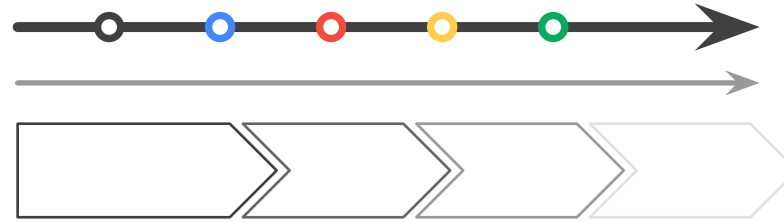
# Graphic Element Styles and Arrows



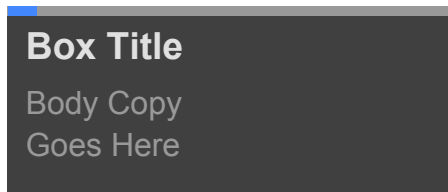
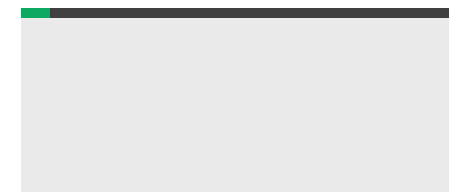
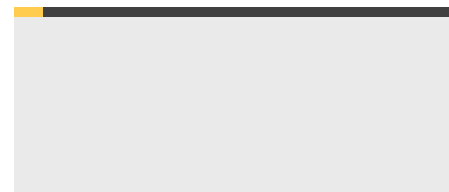
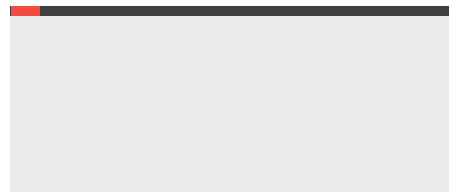
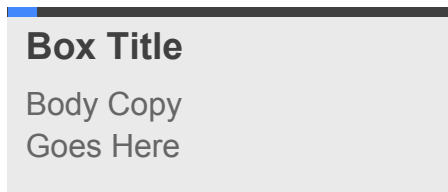
Rounded Boxes



Code Boxes



Arrows



Content Container Boxes



# Pie Chart Example

Subtitle Placeholder

Chart Title

source: place source info here



# Column Chart Example

Subtitle Placeholder

source: place source info here



# Line Chart Example

Subtitle Placeholder

source: place source info here





# Table Option A

Subtitle Placeholder

	Column 1	Column 2	Column 3	Column 4
Row 1	placeholder	placeholder	placeholder	placeholder
Row 2	placeholder	placeholder	placeholder	placeholder
Row 3	placeholder	placeholder	placeholder	placeholder
Row 4	placeholder	placeholder	placeholder	placeholder
Row 5	placeholder	placeholder	placeholder	placeholder
Row 6	placeholder	placeholder	placeholder	placeholder
Row 7	placeholder	placeholder	placeholder	placeholder



# Table Option B

Subtitle Placeholder

Header 1	placeholder	placeholder	placeholder
Header 2	placeholder	placeholder	placeholder
Header 3	placeholder	placeholder	placeholder
Header 4	placeholder	placeholder	placeholder
Header 5	placeholder	placeholder	placeholder





# Segue Slide

Subtitle Placeholder

“ This is an example of  
quote text. ”

Name

Company



# Code Slide With Subtitle Placeholder

## Subtitle Placeholder

```
<script type='text/javascript'>
  // Say hello world until the user starts questioning
  // the meaningfulness of their existence.
  function helloWorld(world) {
    for (var i = 42;--i >= 0;) {
      alert ('Hello' + String(world));
    }
  }
</script>
<style>
p { color: pink }
p { color: blue }
u { color: 'umber' }
</style>
```

HTML



# <Thank You!>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. In eget pretium augue.

Jon@website.com

#Jon

+Jon

