Project 4 Task 2 – Search Youtube Video App

By Bonan Cao – bonanc@andrew.cmu.edu

Description:

My application searches a video with the keyword user gives and display the information and thumbnail of the first result. User can also click the title to watch the video.
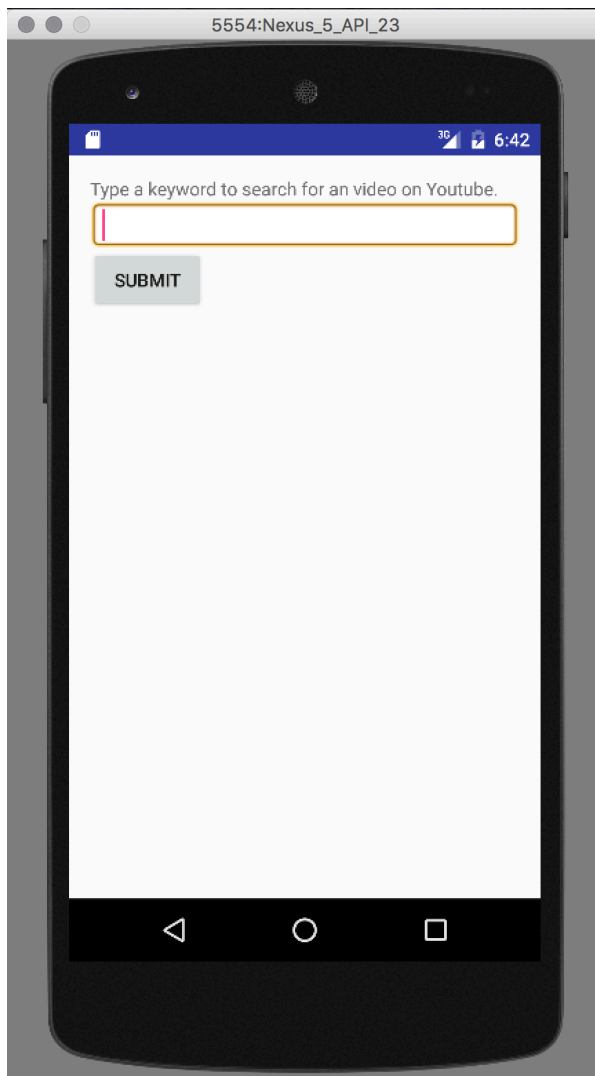
1. Implement a native Android application

The name of my native Android application project in Android Studio is SearchYoutubeVideo.

1.1   Has at least two different kinds of views in your Layout (TextView, EditText, ImageView, etc.)
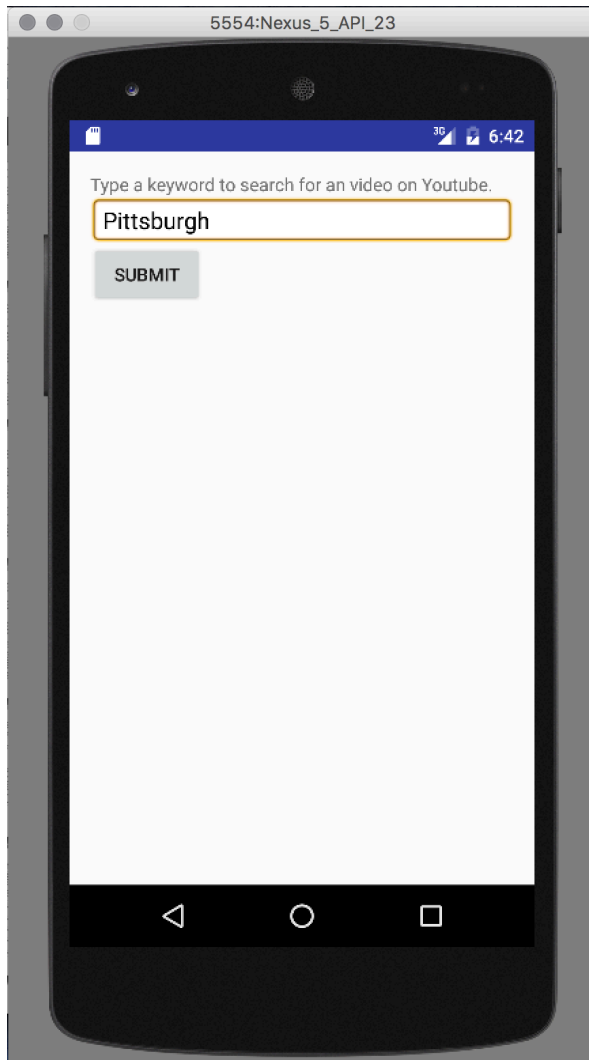
My application uses TextView, EditText, Button and ImageView. See content_main.xml for details. I also use the RelativeLayout for alignment.

Here is a screenshot of the layout before the information and thumbnail are displayed.

1.2   Requires input from the user

Here is a screenshot of the user searching for a video from Youtube

1.3 Makes an HTTP request (using an appropriate HTTP method) to your web service

My application does an HTTP GET request in MainActivity.java. The HTTP request is "https://fast-peak-64814.herokuapp.com/?searchWord=" + searchTerm where searchTerm is the user's search term.

The search method makes this request of my web application, parses the returned JSON to find the information and the thumbnail url of the video, fetches the thumbnail and returns the YoutubeVideo object built.

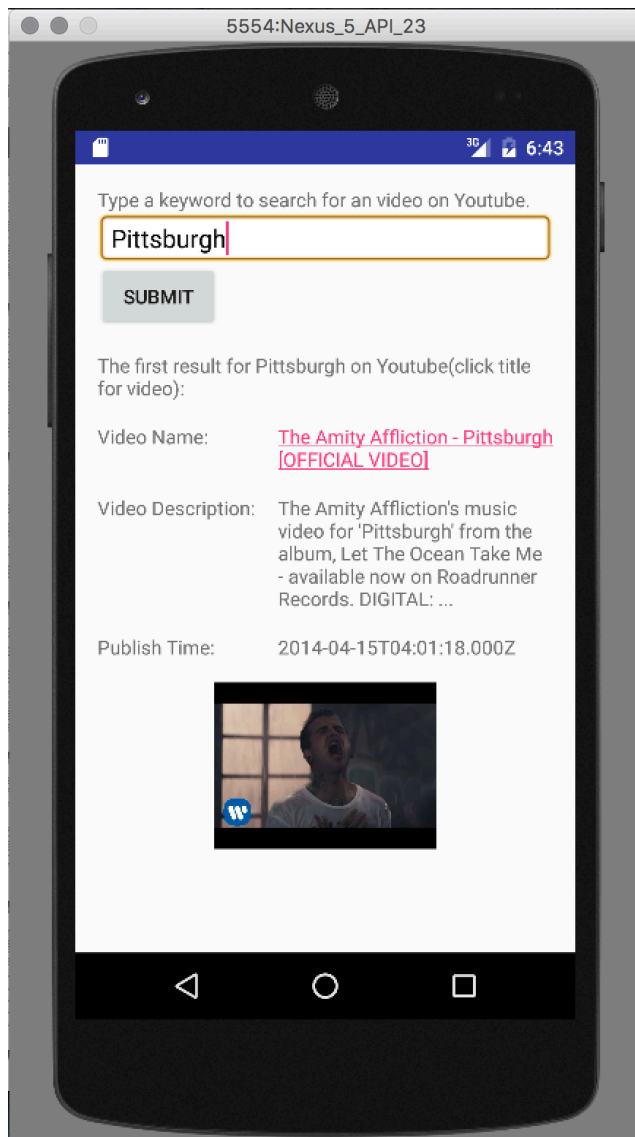## 1.4 Receives and parses an XML or JSON formatted reply from the web service

An example of the JSON reply is:

{"videoTitle":"The Amity Affliction - Pittsburgh [OFFICIAL VIDEO]","videoDesc":"The Amity Affliction's music video for 'Pittsburgh' from
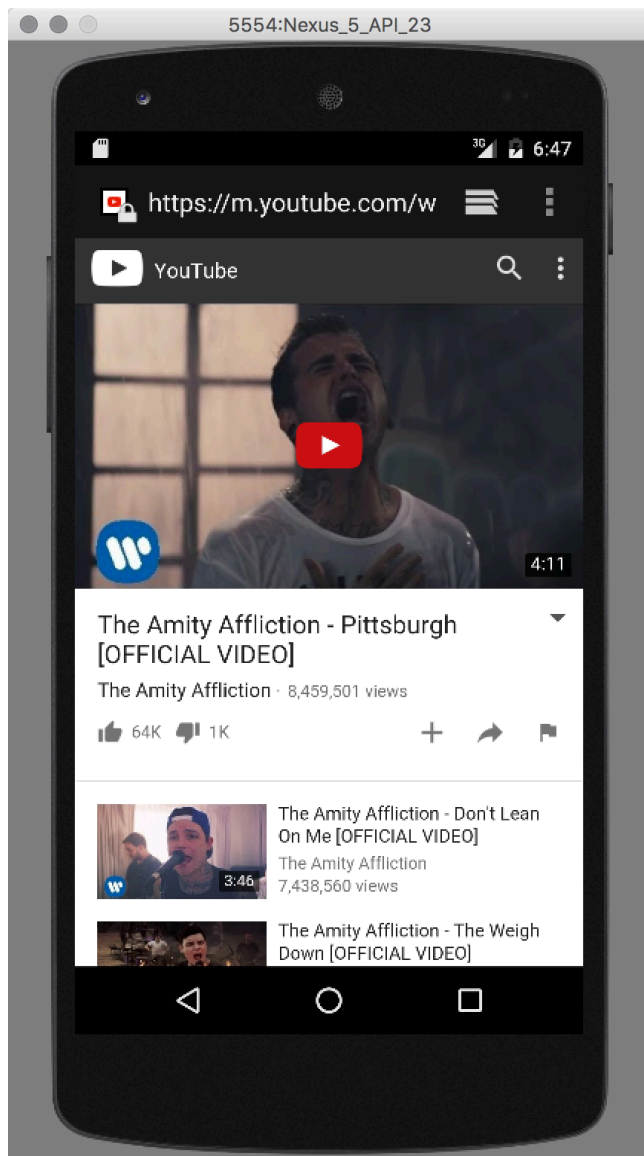
the album, Let The Ocean Take Me - available now on Roadrunner Records. DIGITAL: ...","thumbnailUrl":"https://i.ytimg.com/vi/vu3xGr-lNVI/hqdefault.jpg","publishTime":"2014-04-15T04:01:18.000Z","videoUrl":"https://www.youtube.com/watch?v=vu3xGr-lNVI"}

1.5   Displays new information to the user

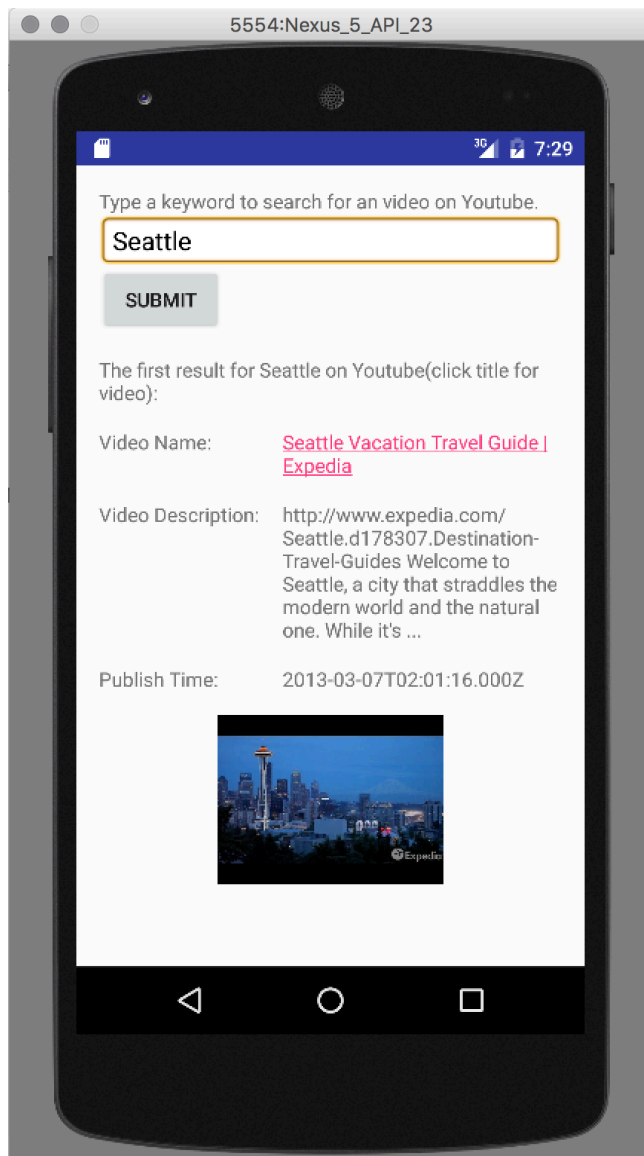Here is the screenshot after the information and thumbnail are returned.

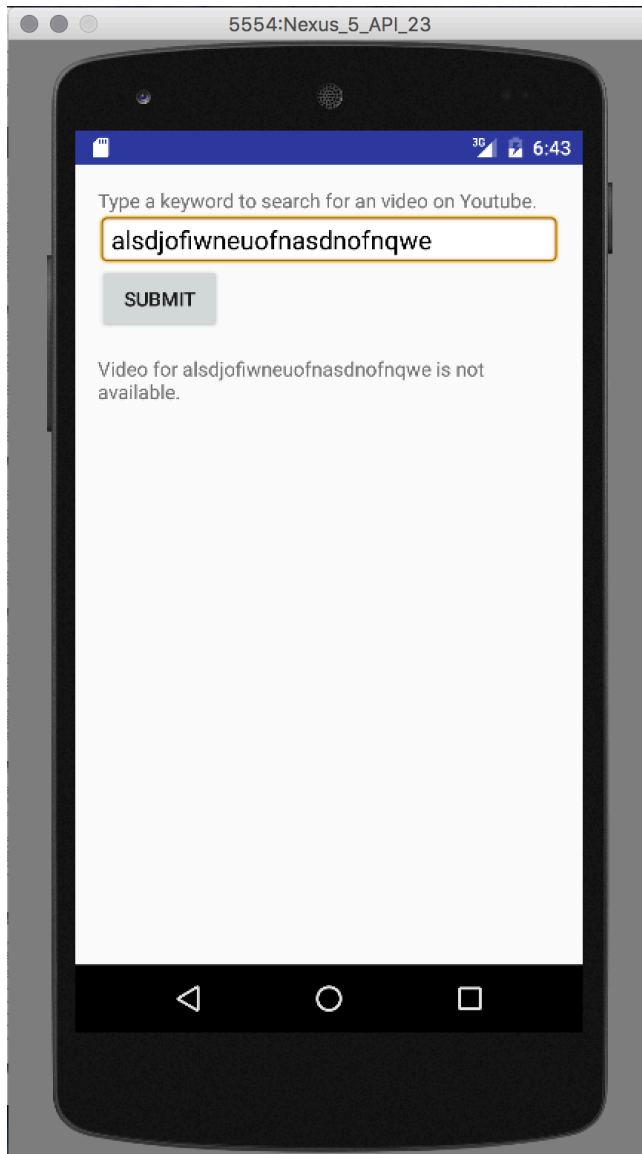You can click the title to watch the video in browser.

1.6 Is repeatable (I.e. the user can repeatedly reuse the application without restarting it.)

The user can type in another search term and hit submit. Here is an example.

The application will give notification if such video is not available.

2. Implement a web application, deployed to Heroku

The name of the Heroku project is YoutubeVideo

2.1   Using an HttpServlet to implement a simple (can be a single path) API

In my web app project:

Model: YoutubeVideoModel.java

Controller: YoutubeVideoServlet.java

The servlet will return the JSON string for the HTTP GET request.

2.2   Receives an HTTP request from the native Android application

YoutubeVideoServlet.java receives the HTTP GET request with the argument "search". It passes this search string on to the model.

2.3 Executes business logic appropriate to your application

YoutubeVideoModel.java makes an HTTP request to "https://www.googleapis.com/youtube/v3/search?part=snippet&q=" + searchTag + "&key=" + apiKey where searchTag is the search term and apiKey is the API key. It parses the JSON response and extracts the parts it needs to respond to the Android application.

2.4 Replies to the Android application with an XML or JSON formatted response.

YoutubeVideoModel.java then generates the JSON string with the extracted parts and sends it to the servlet and servlet sends the JSON string to the Android application:

```
String jsonString = Json.createObjectBuilder()

        .add("videoTitle", videoTitle)

        .add("videoDesc", videoDesc)

        .add("thumbnailUrl", thumbnailUrl)

        .add("publishTime", publishTime)

        .add("videoUrl", videoUrl)

        .build()

        .toString();
```

Here's an example:

{"videoTitle":"The Amity Affliction - Pittsburgh [OFFICIAL VIDEO]", "videoDesc":"The Amity Affliction's music video for 'Pittsburgh' from the album, Let The Ocean Take Me - available now on Roadrunner Records. DIGITAL: ...", "thumbnailUrl":"https://i.ytimg.com/vi/vu3xGr-lNVI/hqdefault.jpg", "publishTime":"2014-04-15T04:01:18.000Z", "videoUrl":"https://www.youtube.com/watch?v=vu3xGr-lNVI"}