

Getting started with Android TV



lukasz_adamczak



<http://czak.pl>



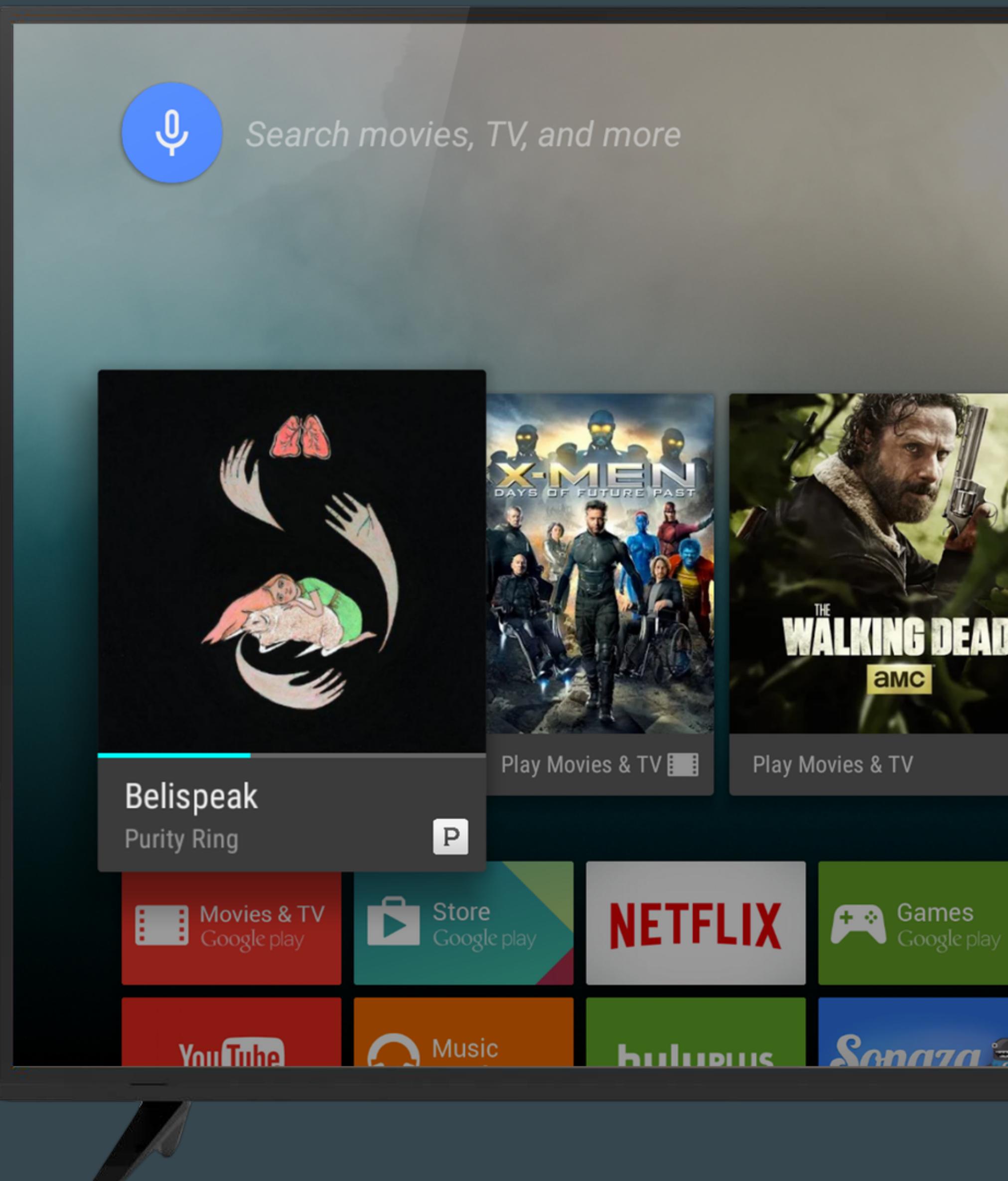
You?

Goals

1. What is Android TV?
2. How do I develop for it?

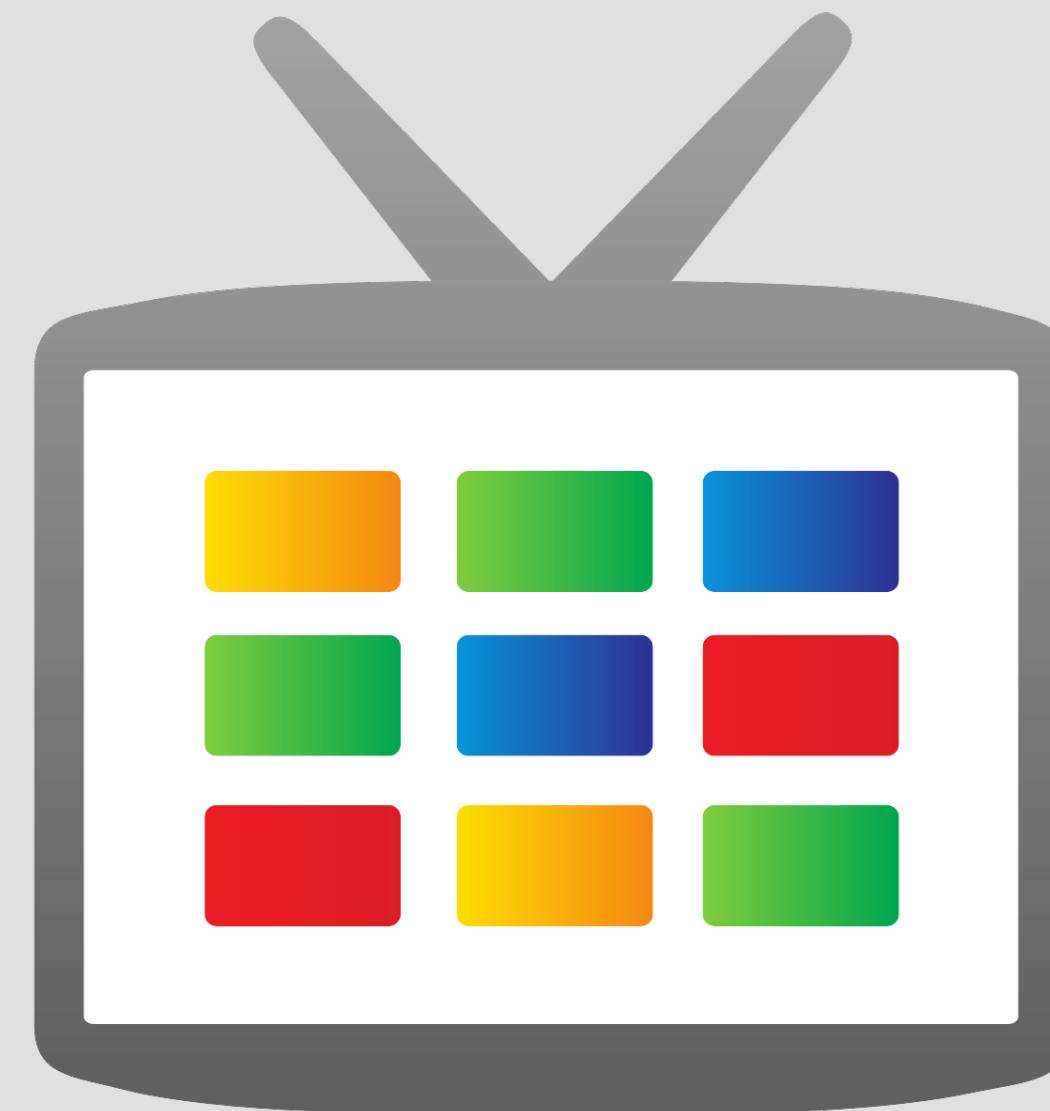
What is Android TV?

- Smart TV platform from Google
- Introduced in 2014 with 5.0 Lollipop
- Leanback UI
- Play Store for TV
- Google Cast built in



What is Android TV?





Google TV™

Google TV

Not to be confused with [Android TV](#).

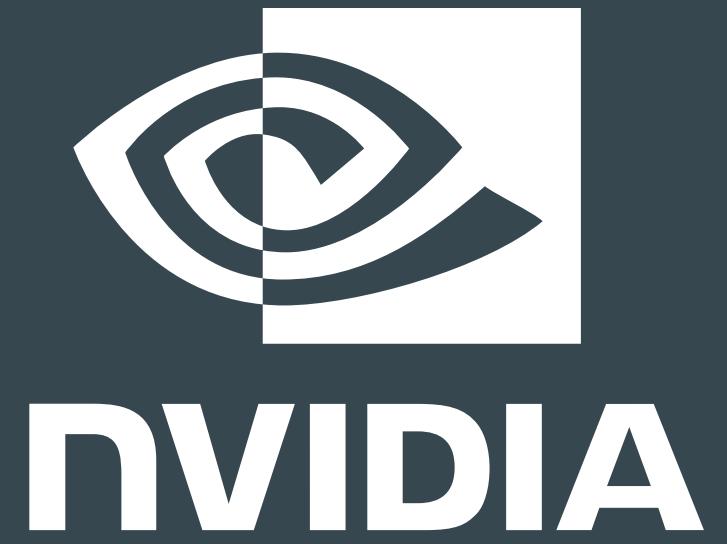
Google TV was a smart TV platform from [Google](#) co-developed by [Intel](#), [Sony](#), and [Logitech](#) that was launched in October 2010^[1] with official devices initially made by Sony and Logitech.^[2] Google TV integrates the [Android](#) operating system and the [Google Chrome](#) web browser to create an [interactive television overlay](#) on top of existing online video sites to add a [10-foot user interface](#), for a smart TV experience.

Google TV's first generation devices were all based on [x86 architecture](#) processors by Intel and were created and commercialized by [Sony](#) and [Logitech](#). The second generation of devices are all based on [ARM architecture](#) processors and with additional partners including [LG](#), [Samsung](#), [Vizio](#), and [Hisense](#). In 2013

SONY

PHILIPS

SHARP



NOW

E



“Talk is cheap. Show me the code.”

- ~~1. What is Android TV?~~
2. How do I develop for it?
 - Leanback library
 - ...and how to live without it
 - TV-specific considerations

android.support.v17.leanback



YouTube

Home

Subscriptions

Uploads

Playlists

History

Watch Later

Purchases

Subscriptions

Google

Recommended



Start Getting Used To Saying President Tr...
The Late Show with Stephen Colbert

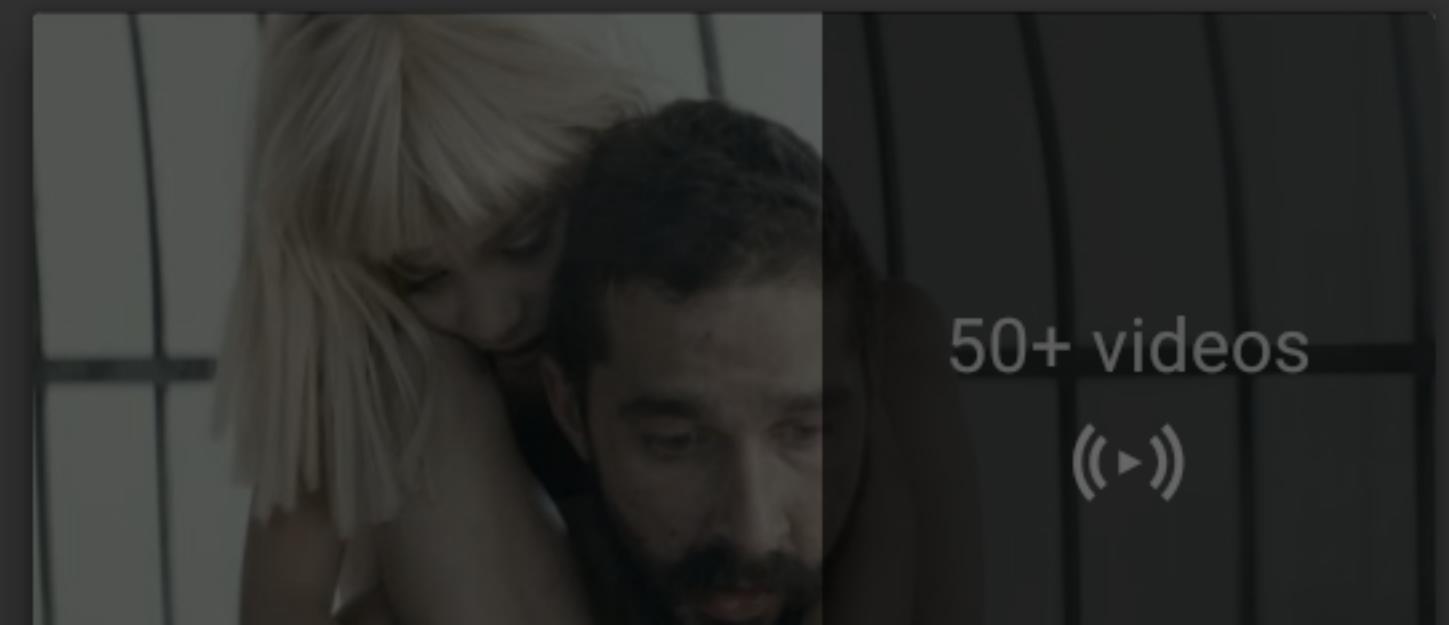


Donald Trump Has A Point, Incredibly
The Late Show with Stephen Colbert

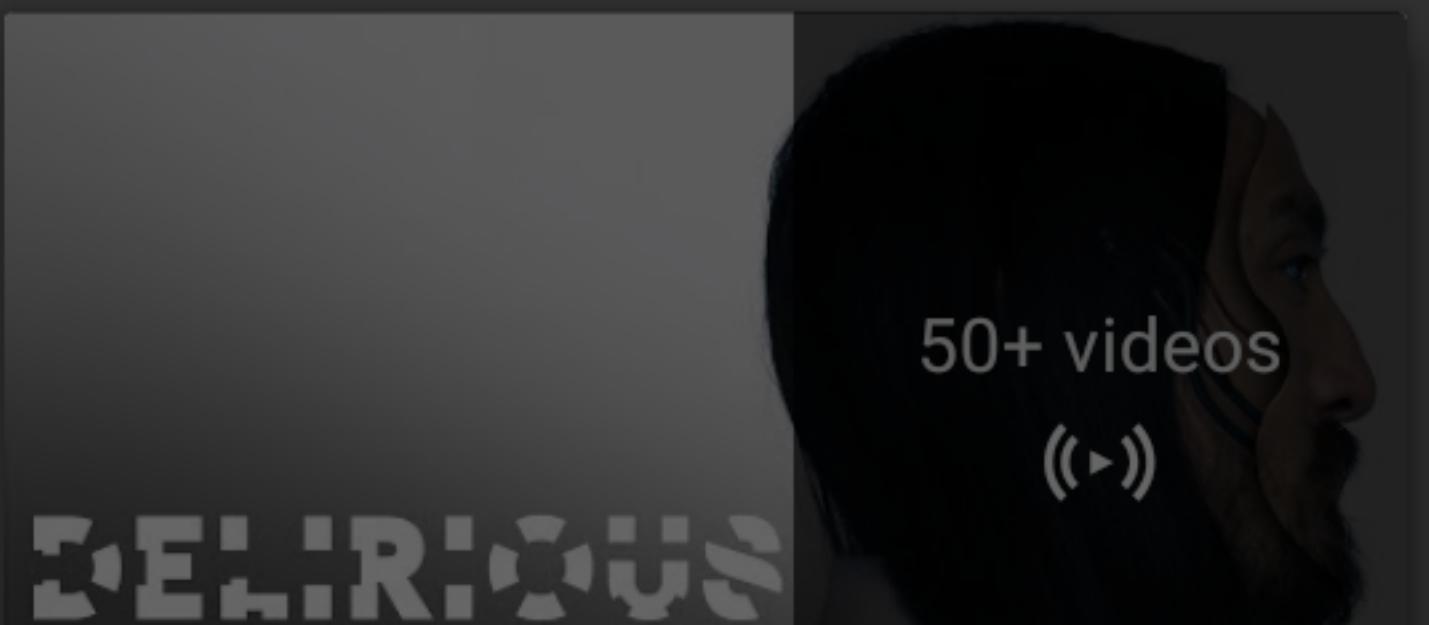


Jennifer Lawrence
The Tonight Show St...

YouTube Mixes

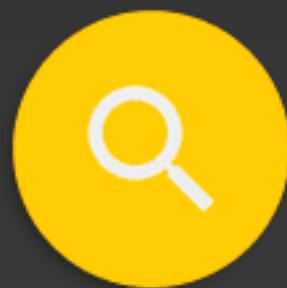


50+ videos
()



50+ videos
()





KITCHEN STOREIES

Recipes



Rhubarb meringue
cake



Chia fresca



Tarte tatin

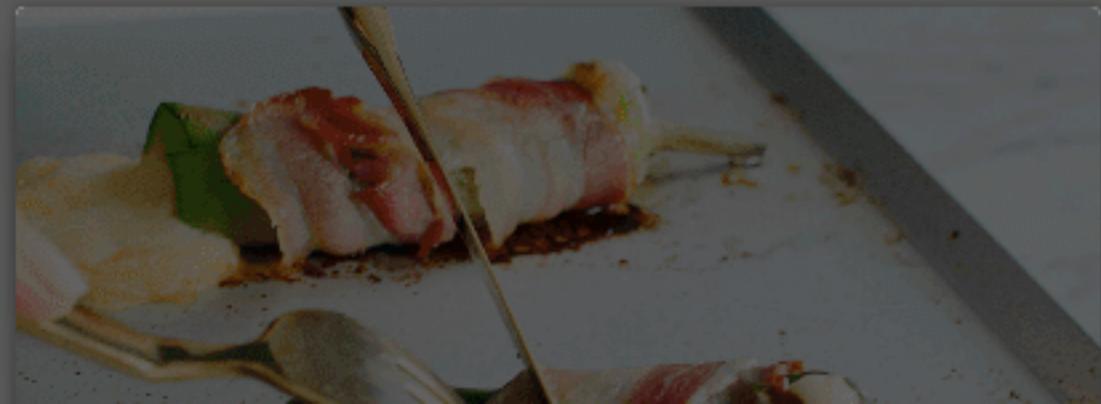


Braised pork belly

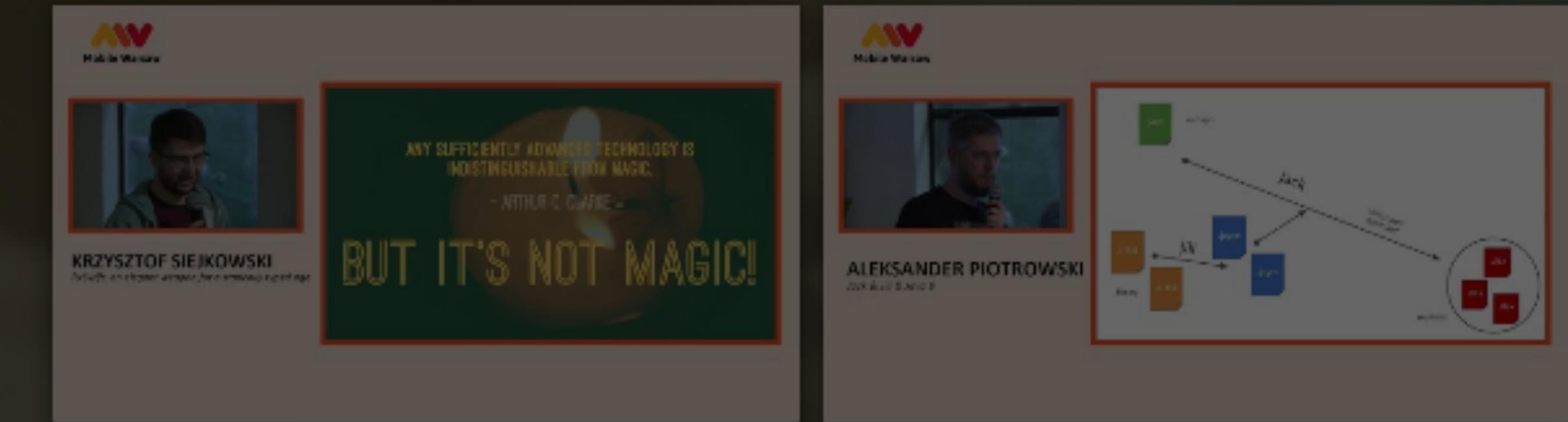


Fish g

Quick bites



Mobile Warsaw #36



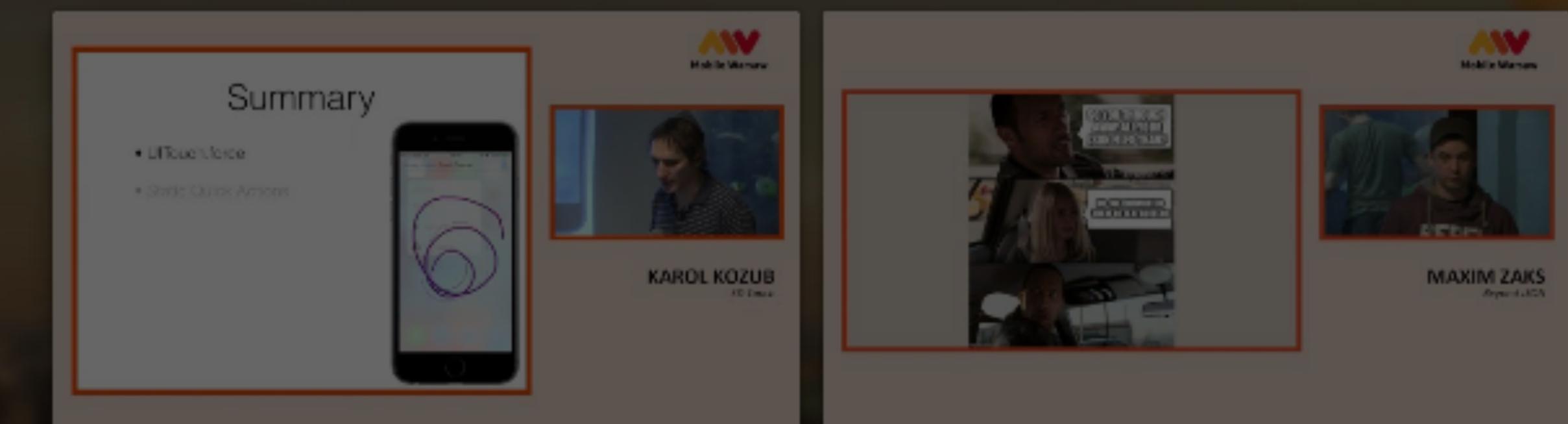
Mobile Warsaw #36 slide deck featuring two speakers: Krzysztof Siejkowski and Aleksander Piotrowski. The first slide shows a quote by Arthur C. Clarke: "ANY SUFFICIENTLY ADVANCED TECHNOLOGY IS INDISTINGUISHABLE FROM MAGIC." - ARTHUR C. CLARKE. The second slide shows a diagram illustrating a system architecture with components like API, DB, and Cache.

Mobile Warsaw #35



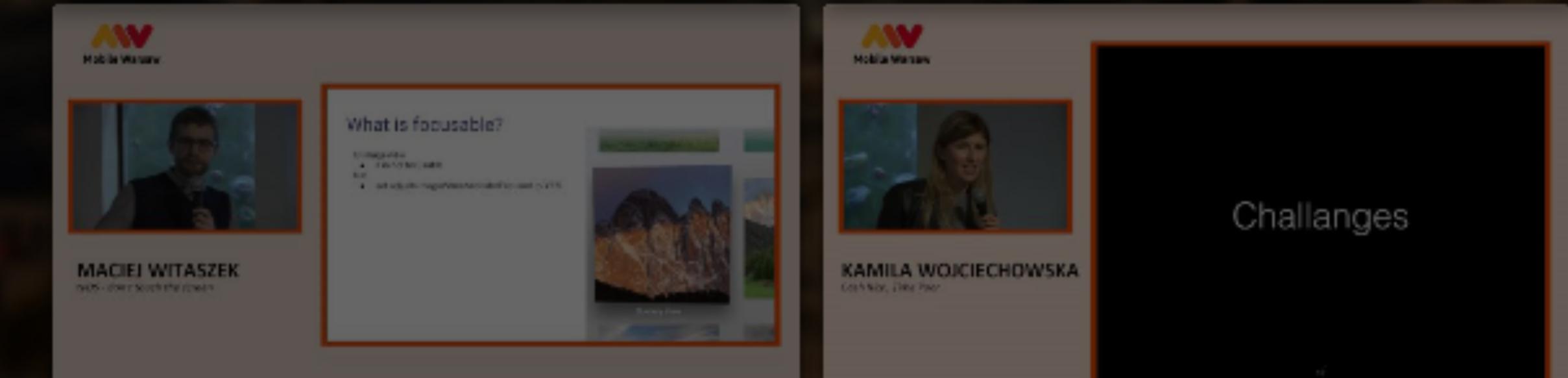
Mobile Warsaw #35 slide deck featuring three speakers: Maciek Górska, Kasper Kondzielski, Szymon Kucur, and Radek Pietruszewski. The slide for Szymon Kucur is titled "FUNCTIONAL REACTIVE PROGRAMMING". A separate slide on the right asks, "But is this really Swifty?"

Mobile Warsaw #34



Mobile Warsaw #34 slide deck featuring two speakers: Karol Kożub and Maksym Zaikin. The slide for Karol Kożub is titled "Summary" and lists "• iTouch/force" and "• Simple Gesture Actions".

Mobile Warsaw #32



Mobile Warsaw #32 slide deck featuring two speakers: Maciej Witaszek and Kamila Wojciechowska. The slide for Maciej Witaszek is titled "What is focusable?" and lists "• Focusable" and "• Focusable elements". The slide for Kamila Wojciechowska is titled "Challanges".

Mobile Warsaw #31

Mobile Warsaw #30

Mobile Warsaw #29

Mobile Warsaw #28

- Support classes for building Leanback user experiences
- Model-View-Presenter pattern

- android.support.v17.leanback.app
- android.support.v17.leanback.widget

VerticalGridFragment



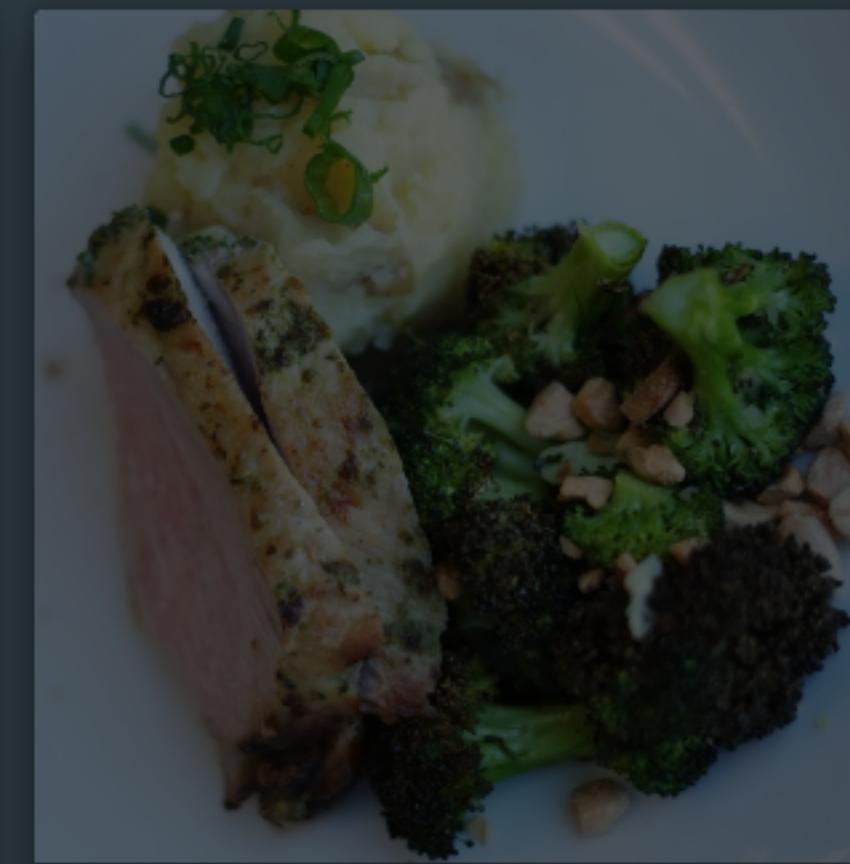
Nüsse

\$3.99/lb



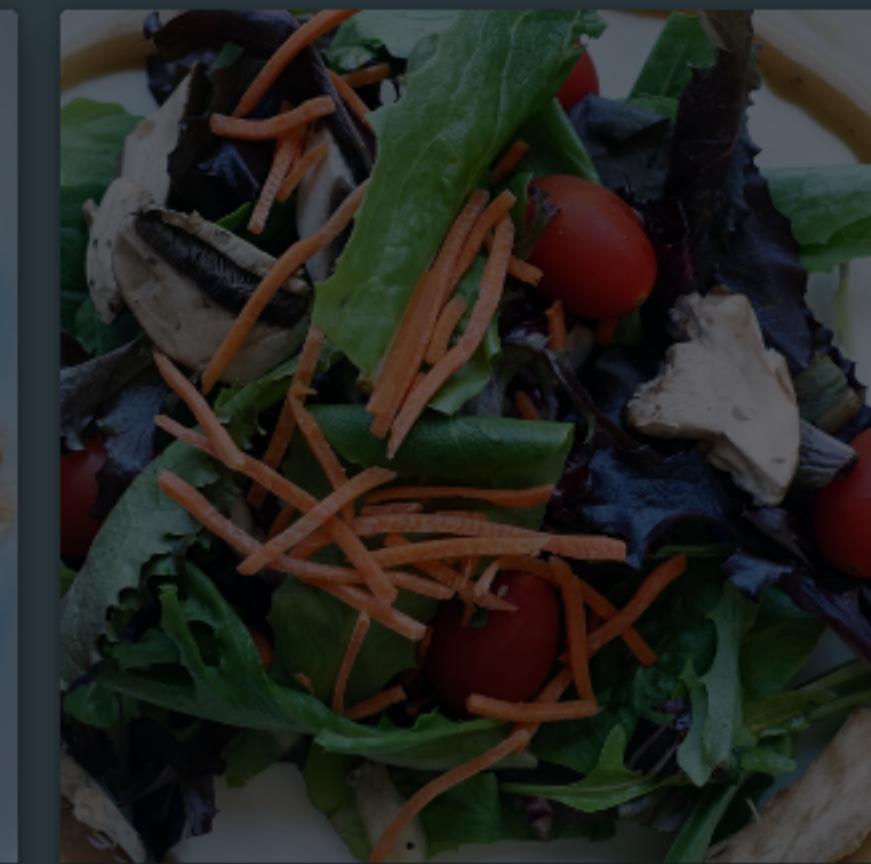
Undefinierbar

\$3.99/lb



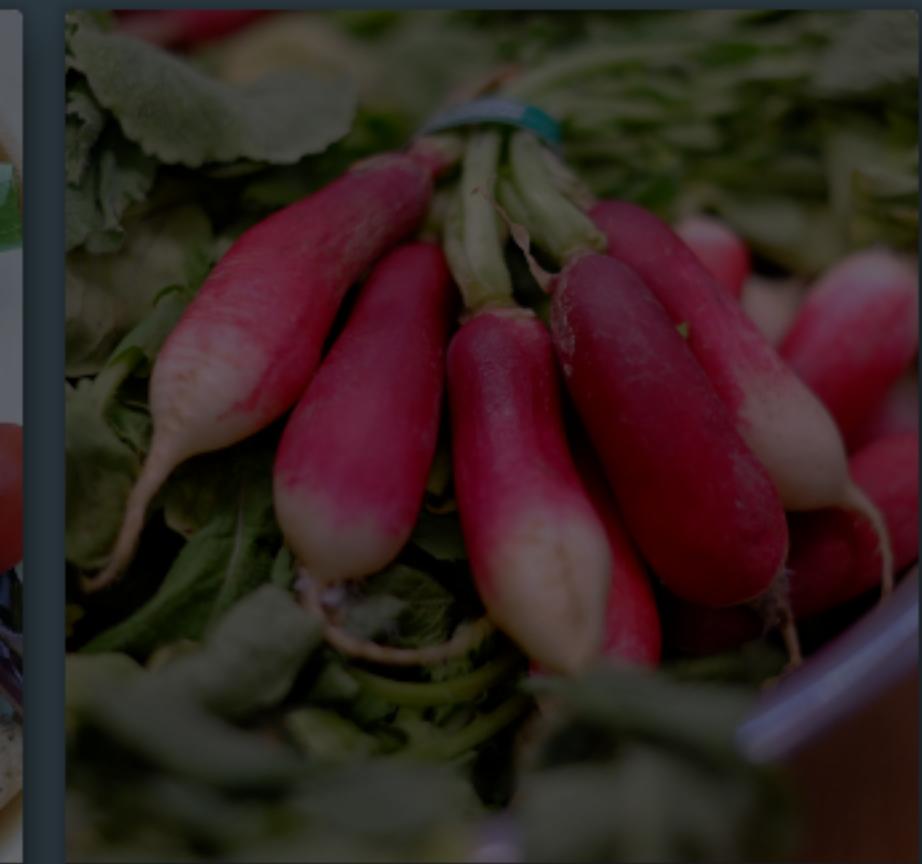
Brocoli

\$3.99/lb



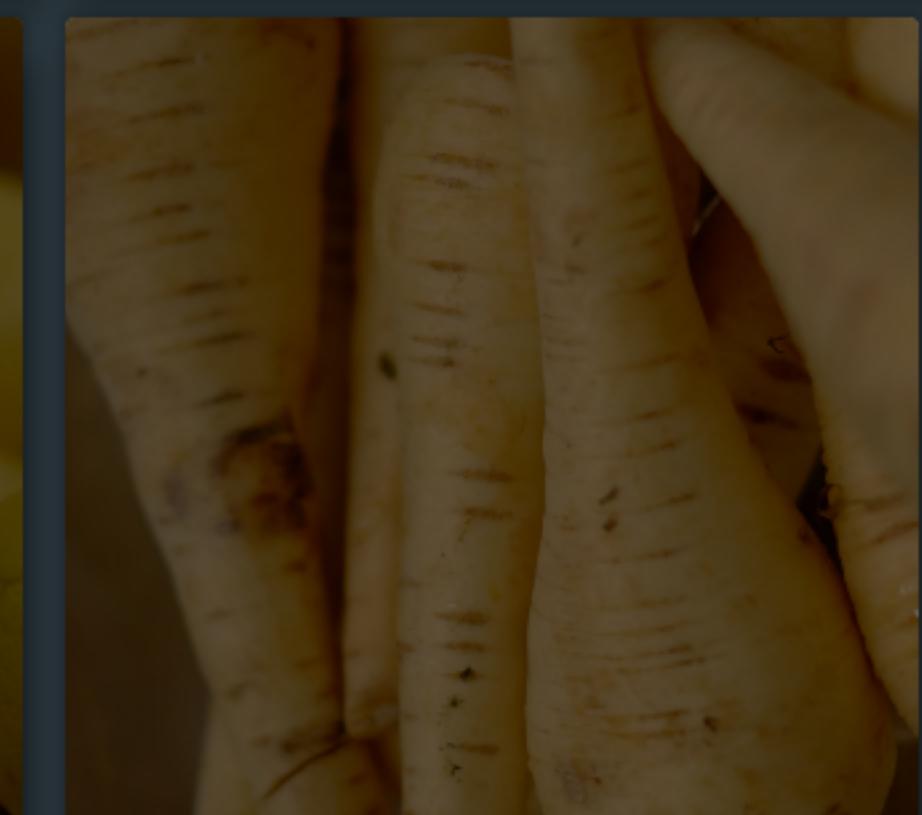
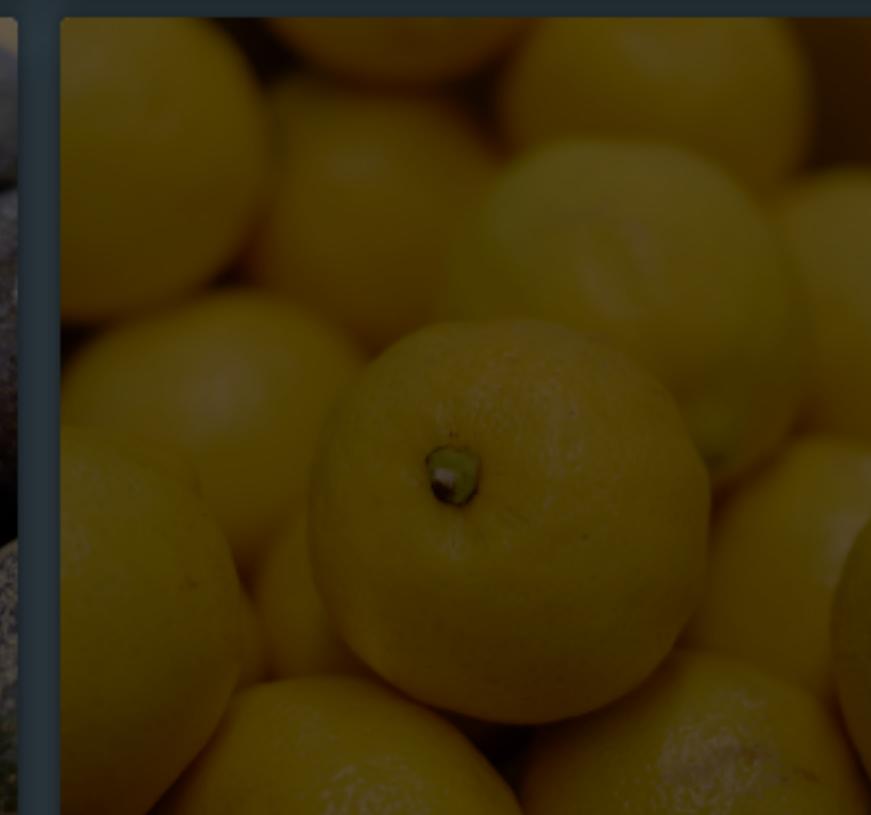
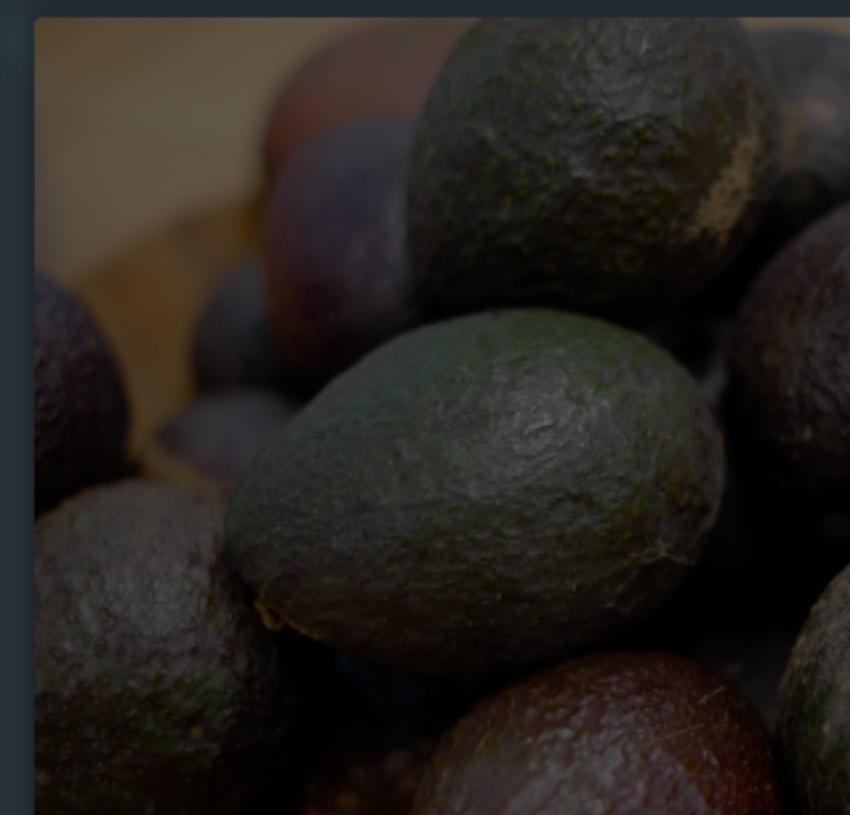
Salat

\$3.99/lb



Große Radischen

\$3.99/lb



RowsFragment

Vegetables



Salat
\$3.99/lb



Radish
\$3.99/lb



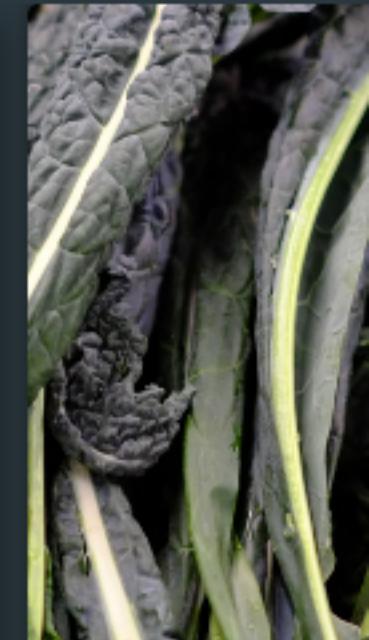
Red onion
\$3.99/lb



Celery
\$3.99/lb

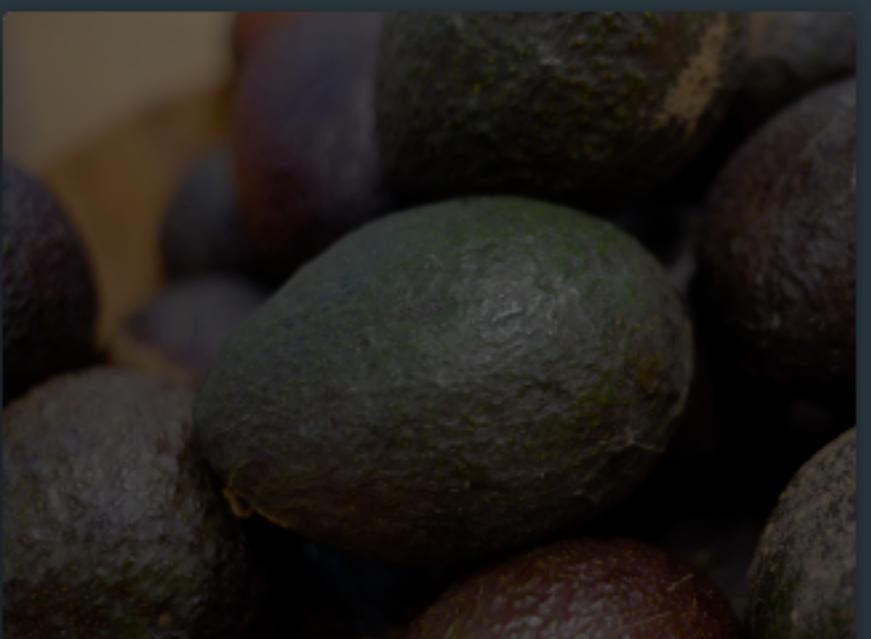


White vegetable
\$3.99/lb

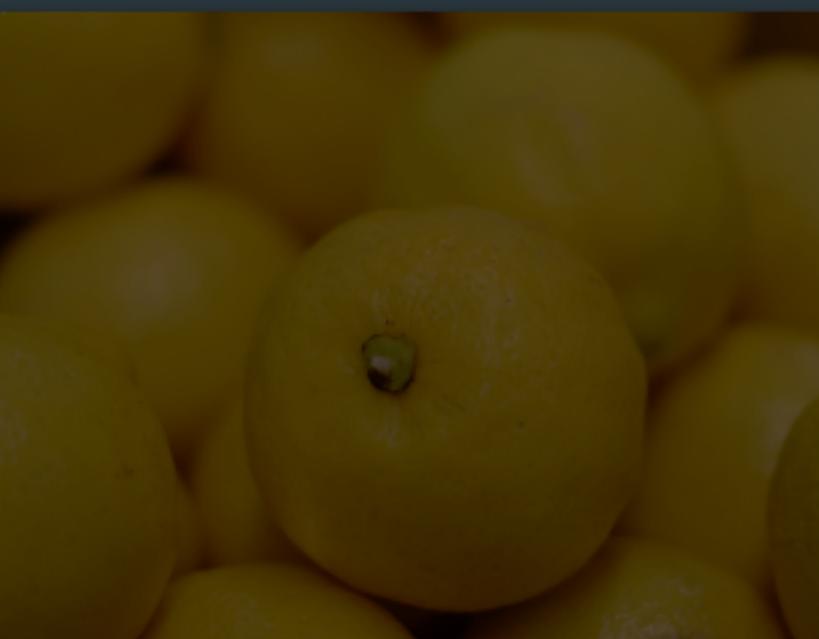


Cabbage?
\$3.99/lb

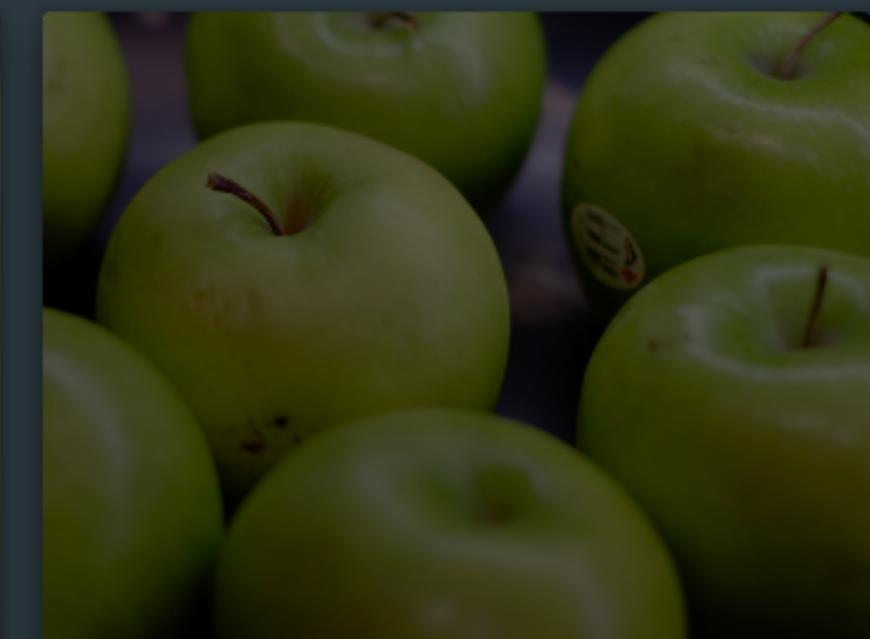
Fruit



Exotic somethings



Lemons



Apples



More Apples



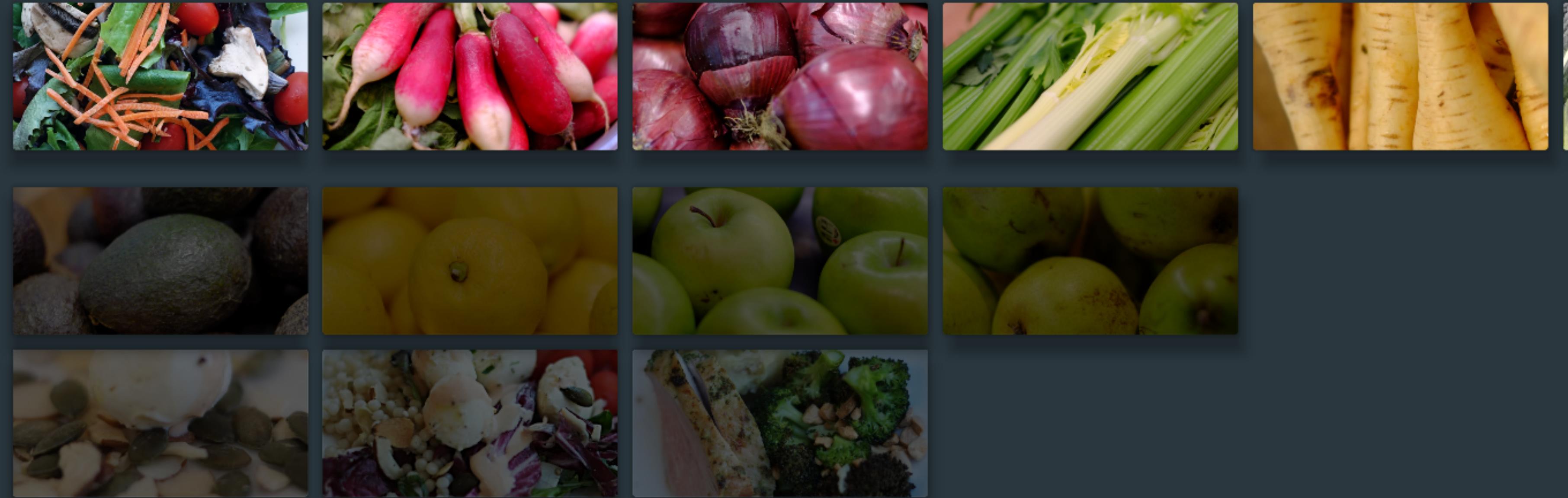
BrowseFragment

Menu

Vegetables

Fruit

Meat



DetailsFragment



BUY \$9.99

ADD TO WISHLIST

RELATED

A Summer in a Canyon

Kate Douglas Smith Wiggin 1914

It was nine o'clock one sunny California morning, and Geoffrey Strong stood under the live-oak trees in Las Flores Cañon, with a pot of black paint in one hand and a huge brush in the other. He could have handled these implements to better purpose and with better grace had not his arms been firmly held by three laughing girls, who pulled not wisely, but too well. He was further incommoded by the presence of a small urchin who lay on the dusty ground beneath his feet, fastening an upward clutch on the legs of his trousers.

GuidedStepFragment

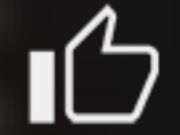
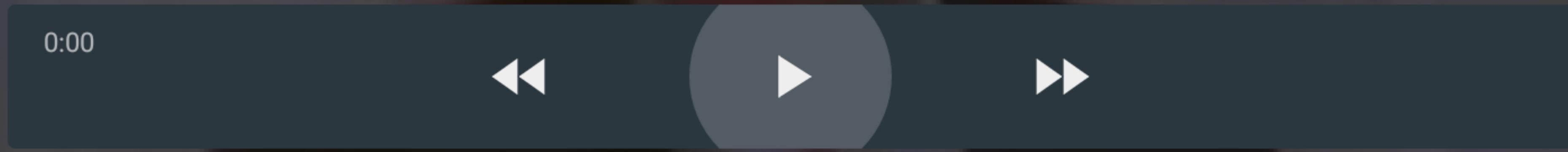
Used to prompt for user input and create wizards.

Awsum

Sux

PlaybackOverlayFragment

Definitely not Rick Astley



Using leanback fragments

1. Subclass fragment
2. Place in layout
3. Fill with data
4. Implement Presenter
5. Customize via styles

Using leanback fragments

```
class Pizza {  
    String photo;  
    String name;  
    int price;  
}
```

Title



Pizza 4

1. Subclass fragment

```
package pl.czak.pizzeria;

class PizzaGridFragment
    extends VerticalGridFragment {

}
```

2. Place in layout

```
<FrameLayout>

    <fragment
        android:id="@+id/pizza_grid_fragment"
        android:name="pl.czak.pizzeria.PizzaGridFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

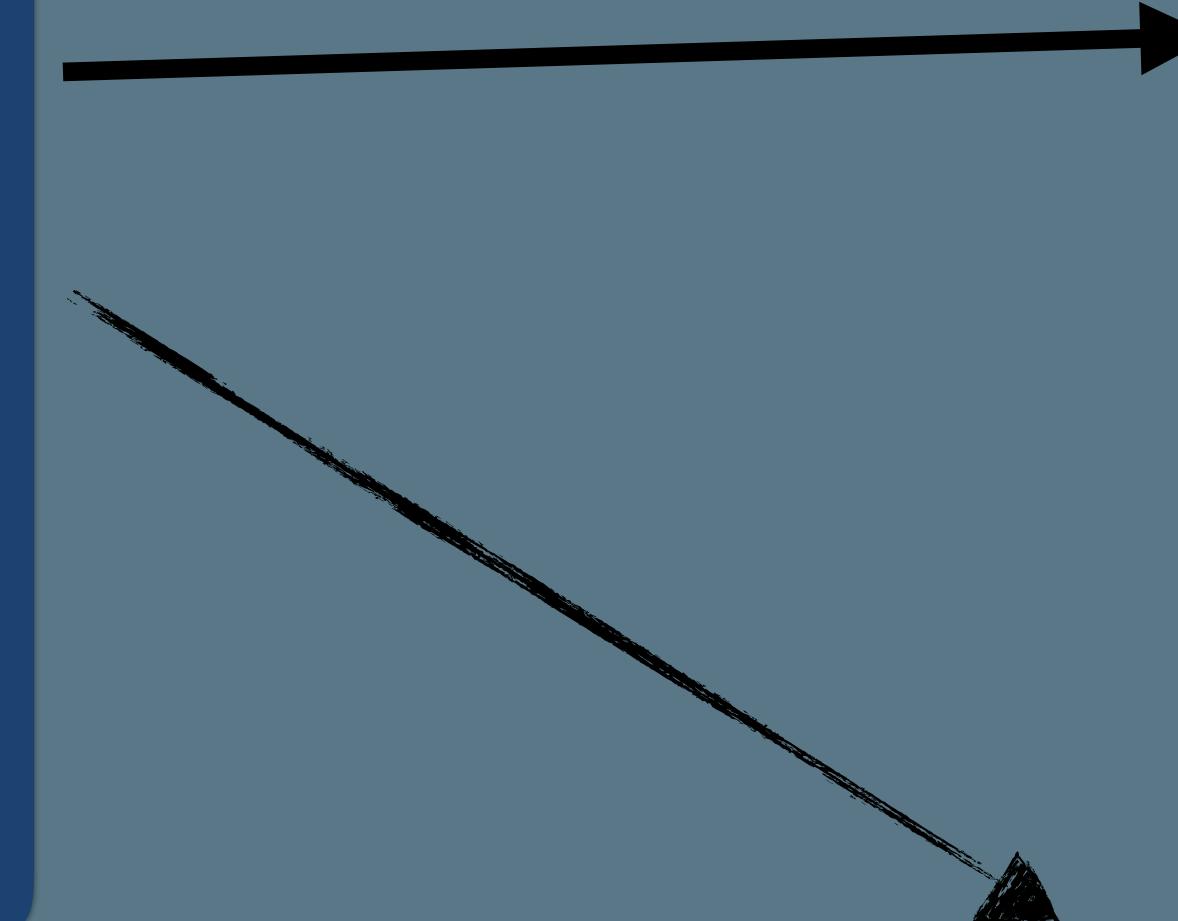
</FrameLayout>
```

3. Fill with data

```
ArrayObjectAdapter adapter =  
    new ArrayObjectAdapter(new PizzaCardPresenter());  
  
adapter.add(new Pizza("Margherita", 30, "margherita.jpg"));  
adapter.add(new Pizza("Pepperoni", 35, "pepperoni.jpg"));  
  
setAdapter(adapter);
```

4. Implement presenter

```
class Pizza {  
    String photo;  
    String name;  
    int price;  
}
```



Pizza 1

4. Implement presenter

```
package android.support.v17.leanback.widget;

public abstract class Presenter {
    public abstract ViewHolder onCreateViewHolder(ViewGroup parent);
    public abstract void onBindViewHolder(ViewHolder viewHolder, Object item);
    public abstract void onBindViewHolder(ViewHolder viewHolder);
}
```

4. Implement presenter

```
class PizzaCardPresenter extends Presenter {  
    /* ... */  
  
    @Override  
    public ViewHolder onCreateViewHolder(ViewGroup parent) {  
        View view = new ImageCardView(parent.getContext());  
        return new ViewHolder(view);  
    }  
  
    /* ... */  
}
```

4. Implement presenter

```
class PizzaCardPresenter extends Presenter {  
    @Override  
    public void onBindViewHolder(ViewHolder viewHolder, Object item) {  
        Pizza pizza = (Pizza) item;  
        ImageCardView view = (ImageCardView) viewHolder.view;  
  
        view.setTitleText(pizza.getName());  
        view.setMainImage(/*...*/);  
    }  
}
```

4. Implement presenter

```
class PizzaCardPresenter extends Presenter {  
  
    @Override  
    public void onUnbindViewHolder(ViewHolder viewHolder) {  
        // nothing to do here...  
    }  
  
}
```

5. Customize via styles

```
<style name="PizzaCardImage"
      parent="Widget.Leanback.ImageCardView.ImageStyle">
    <item name="android:layout_width">200dp</item>
    <item name="android:layout_height">200dp</item>
</style>

<style name="AppTheme" parent="@style/Theme.Leanback">
    <item name="imageCardViewImageStyle">@style/PizzaCardImage</item>
</style>
```

Using leanback fragments

1. Subclass fragment
2. Place in layout
3. Fill with data
4. Implement Presenter
5. Customize via styles

MACIEK GÓRSKI,
KASPER KONDZIELSKI
Livecoding: family meeting (iPhone talk)

SYMFONIEN
Szymon Kuczur
RTE... MVVM and Reactive Programming

REACTIVE
PROGRAMMING

RADEK PIETRUSZEWSKI
Statically-Typed Swift APIs

really Swifty?

Livecoding: family meeting
Maciek Górski & Kasper

Rrrrrr... MVVM and Reactiv...
Szymon Kuczur

Statically-Typed Swifty APIs
Radek Pietruszewski

Mobile Warsaw #34

Summary

- UITapGestureRecognizer
- Static Quick Actions



KAROL KOZUB
3D Touch

MW
Mobile Warsaw

Mobile Warsaw



MAXIM ZAKS
Beyond JSON

MW
Mobile Warsaw

Mobile Warsaw

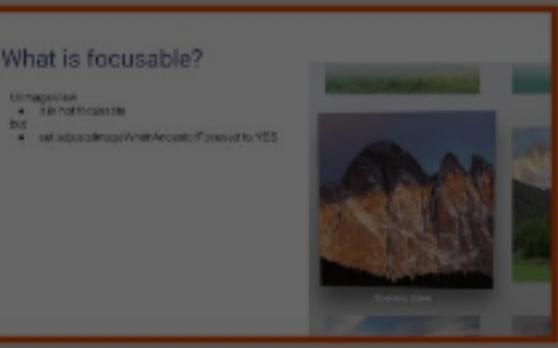
3D Touch
Karol Kozub

Beyond JSON
Maxim Zaks

Mobile Warsaw #33

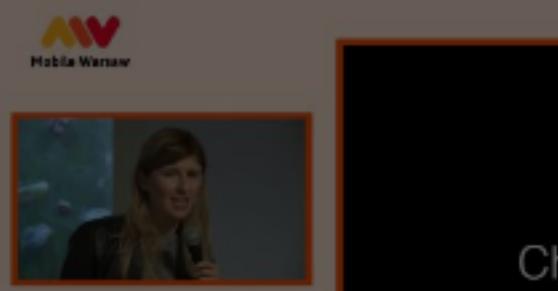


MACIEJ WITASZEK
tvOS: don't touch the screen



What is focusable?

Imagine... it's not possible
not all elements are focusable to YES



KAMILA WOJCIECHOWSKA
Cash Rich, Time Poor

Challanges

12

tvOS: Don't touch the screen
Maciej Witaszek

Cash Rich, Time Poor
Kamila Wojciechowska



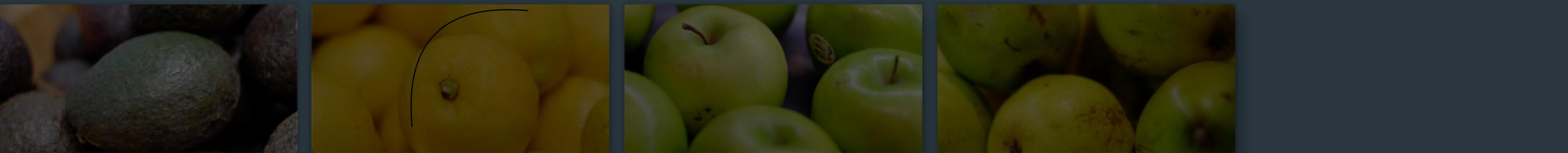
Menu

Vegetables

Fruit

Meat

Row



Diving with Sharks

A Googler

0:02



0:05



android.support.v17.leanback

- ~NN fragment classes in total
- still changing
- <https://github.com/googlesamples/leanback-showcase>

1. ~~What is Android TV?~~
2. How do I develop for it?
 - ~~Leanback library~~
 - ...and how to live without it
 - TV-specific considerations

duolingoTV

- Navigation & Feedback
- User Interface
- Device Capabilities

Navigation



- No touch
- 7 buttons
 - UP/DOWN/LEFT/RIGHT
 - SELECT
- BACK
- HOME

Navigation

“The Android framework handles directional navigation between layout elements automatically, so you typically do not need to do anything extra for your app.”

– <https://developer.android.com/training/tv/start/navigation.html>

UI: Feedback

- UI must reflect what's in focus
- focusable & focused View attributes
- `<item android:state_focused="true">`
for your state list drawables

UI: Best practices

- Use Theme.NoTitleBar
- Prefer GridView over ListView

Device capabilities

- Don't require
- Check at runtime

1. ~~What is Android TV?~~
2. How do I develop for it?
 - ~~Leanback library~~
 - ~~...and how to live without it~~
 - TV-specific considerations

Declare a TV activity

```
<activity  
    android:name="com.example.android.TvActivity"  
    android:label="@string/app_name"  
    android:theme="@style/Theme.Leanback">  
  
    <intent-filter>  
        <action android:name="android.intent.action.MAIN" />  
        <category android:name="android.intent.category.LEANBACK_LAUNCHER" />  
    </intent-filter>  
  
</activity>
```

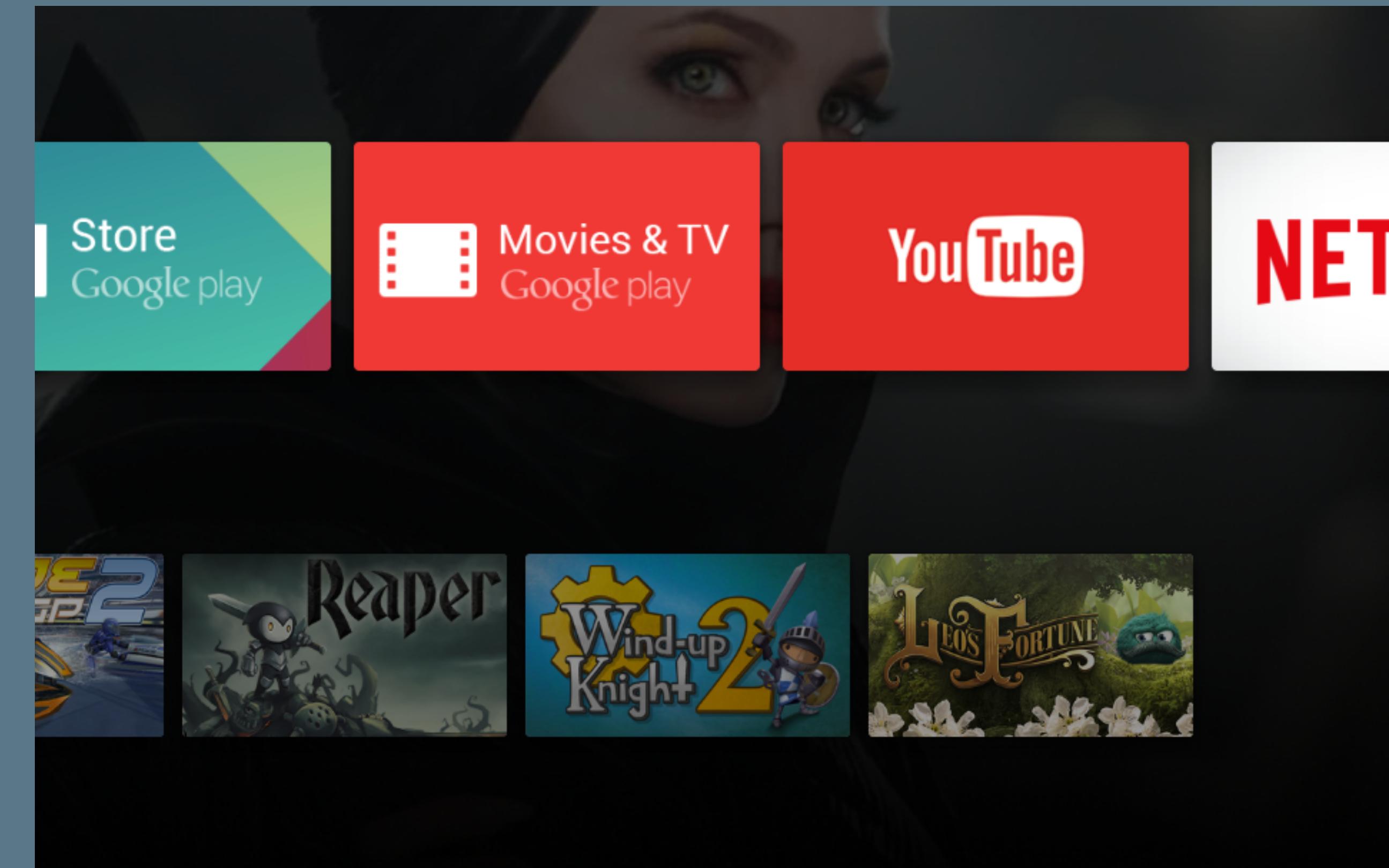
Declare leanback support

```
<manifest>
    <uses-feature android:name="android.software.leanback"
                  android:required="false" />
    ...
</manifest>
```

Touchscreen not required

```
<manifest>
    <uses-feature android:name="android.hardware.touchscreen"
                  android:required="false" />
    ...
</manifest>
```

Provide a home screen banner



Resource variants for TV

- -tvdpi at 720p
-xhdpi at 1080p
- 960x540 dp effective dimensions
- -sw540dp
- -television

Deployment

- Common APK vs Dual APKs

1. ~~What is Android TV?~~
2. How do I develop for it?
 - ~~Leanback library~~
 - ~~...and how to live without it~~
 - ~~TV specific considerations~~

1. ~~What is Android TV?~~
2. ~~How do I develop for it?~~
 - ~~Leanback library~~
 - ~~...and how to live without it~~
 - ~~TV specific considerations~~

Questions?