

IMMOBILIEN
SCOUT24

Navigation and the Single Activity

Learnings from a Skeptic

Droidcon Lisbon 

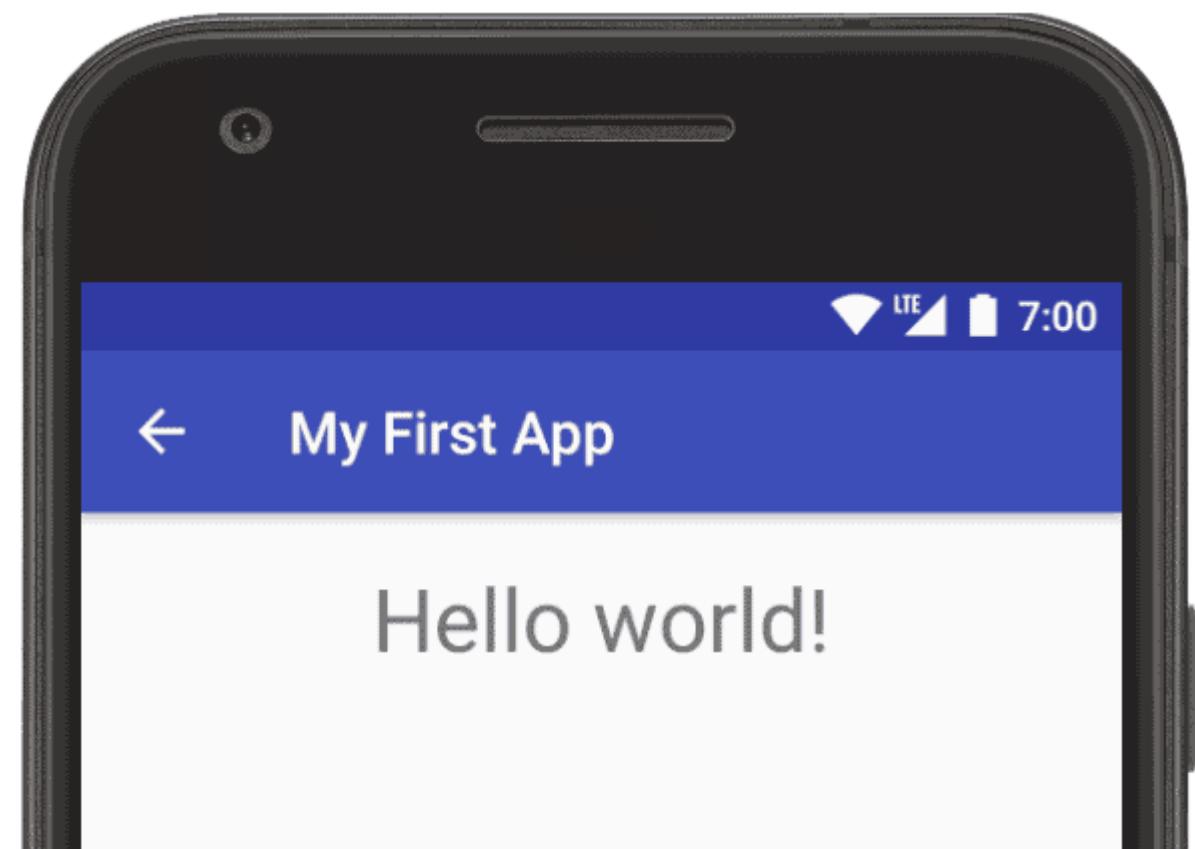
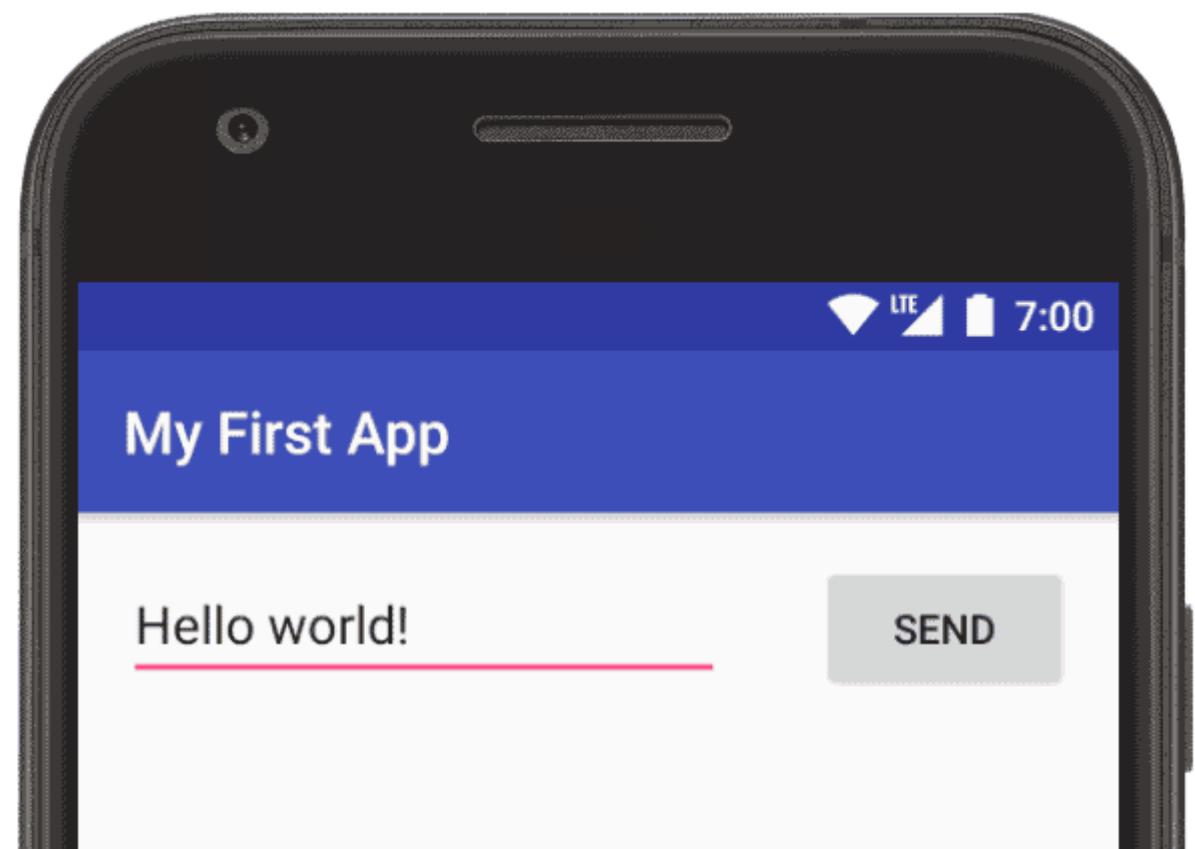


@askashdavies





@askashdavies



```
class MainActivity : AppCompatActivity {

    private val message: EditText
        get() = findViewById(R.id.message)

    override fun onCreate(savedInstanceState: Bundle) {
        /* ... */
    }

    fun sendMessage() {
        val intent = Intent(this, DisplayMessageActivity::class.java)
        intent.putExtra(DisplayMessageActivity.EXTRA_MESSAGE, message.text.toString())
        startActivity(intent)
    }
}
```



@askashdavies

"Once we have gotten in to this entry-point to your UI,
we really don't care how you organise the flow
inside."

— *Dianne Hackborn, Android Framework team, 2016*



@askashdavies



Family



Find contacts

New



4 contacts

**H**
Home**K****T**
Torin**東洸****★ K**

Phone

(415) 555-1212

Home

Email

example@gmail.com

Home

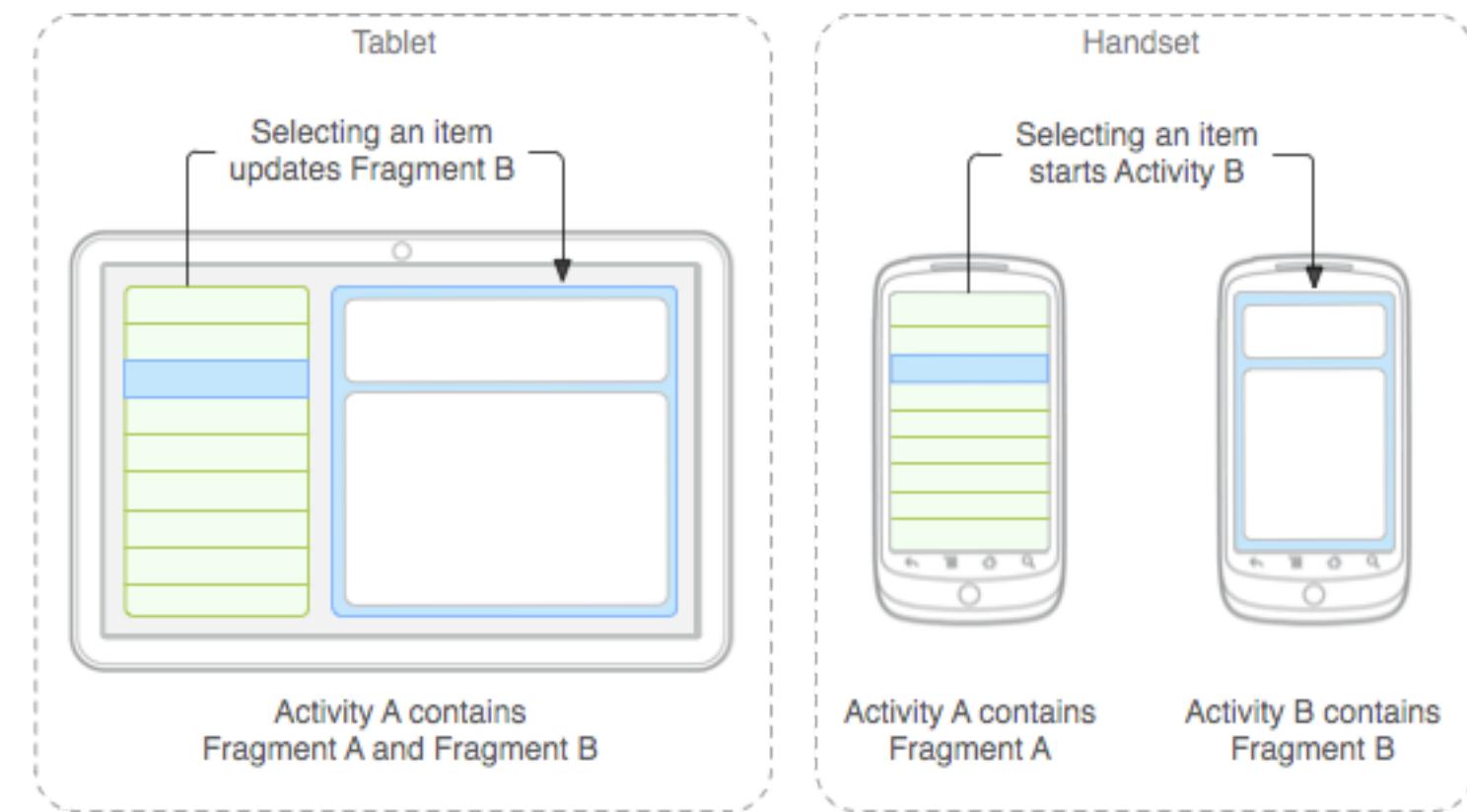


Groups

Family



9:41 *



```
public class MainActivity extends FragmentActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.news_articles);

        // Check that the activity is using the fragment_container FrameLayout
        if (findViewById(R.id.fragment_container) != null) {

            // However, if we're being restored from a previous state, then we don't need to do anything,
            // and should return or else we could end up with overlapping fragments.
            if (savedInstanceState != null) {
                return;
            }

            // Create a new Fragment to be placed in the activity layout
            HeadlinesFragment firstFragment = new HeadlinesFragment();

            // In case this activity was started with special instructions from an
            // Intent, pass the Intent's extras to the fragment as arguments
            firstFragment.setArguments(getIntent().getExtras());

            // Add the fragment to the 'fragment_container' FrameLayout
            getSupportFragmentManager()
                .beginTransaction()
                .add(R.id.fragment_container, firstFragment)
                .commit();
        }
    }
}
```

@askashdavies

```
public class MainActivity extends FragmentActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.news_articles);

        // Check that the activity is using the fragment_container FrameLayout
        if (findViewById(R.id.fragment_container) != null) {

            // However, if we're being restored from a previous state, then we don't need to do anything,
            // and should return or else we could end up with overlapping fragments.
            if (savedInstanceState != null) {
                return;
            }

            // Create a new Fragment to be placed in the activity layout
            HeadlinesFragment firstFragment = new HeadlinesFragment();

            // In case this activity was started with special instructions from an
            // Intent, pass the Intent's extras to the fragment as arguments
            firstFragment.setArguments(getIntent().getExtras());

            // Add the fragment to the 'fragment_container' FrameLayout
            getSupportFragmentManager()
                .beginTransaction()
                .add(R.id.fragment_container, firstFragment)
                .commit();
        }
    }
}
```

@askashdavies

```
public class MainActivity extends FragmentActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.news_articles);

        // Check that the activity is using the fragment_container FrameLayout
        if (findViewById(R.id.fragment_container) != null) {

            // However, if we're being restored from a previous state, then we don't need to do anything,
            // and should return or else we could end up with overlapping fragments.
            if (savedInstanceState != null) {
                return;
            }

            // Create a new Fragment to be placed in the activity layout
            HeadlinesFragment firstFragment = new HeadlinesFragment();

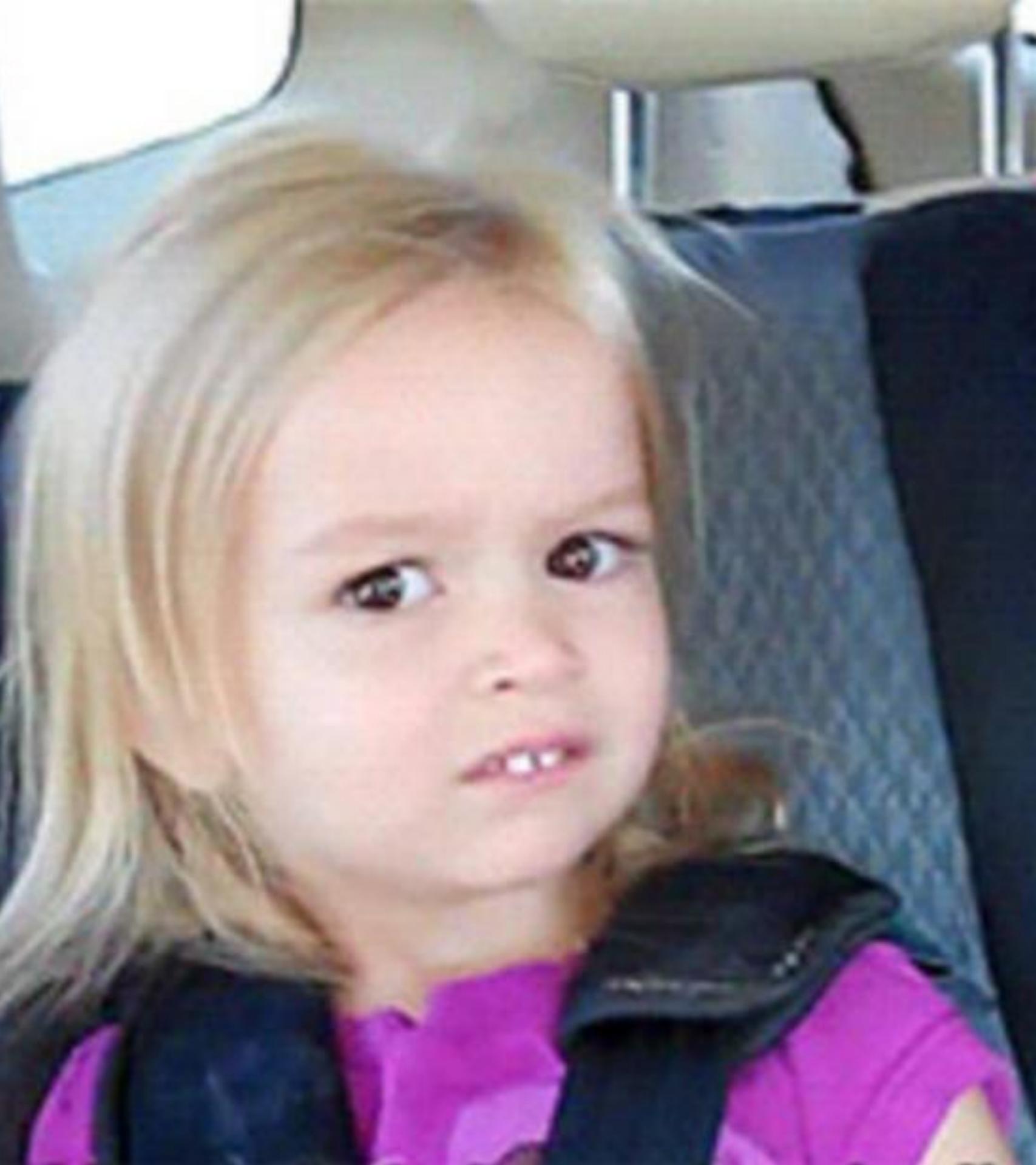
            // In case this activity was started with special instructions from an
            // Intent, pass the Intent's extras to the fragment as arguments
            firstFragment.setArguments(getIntent().getExtras());

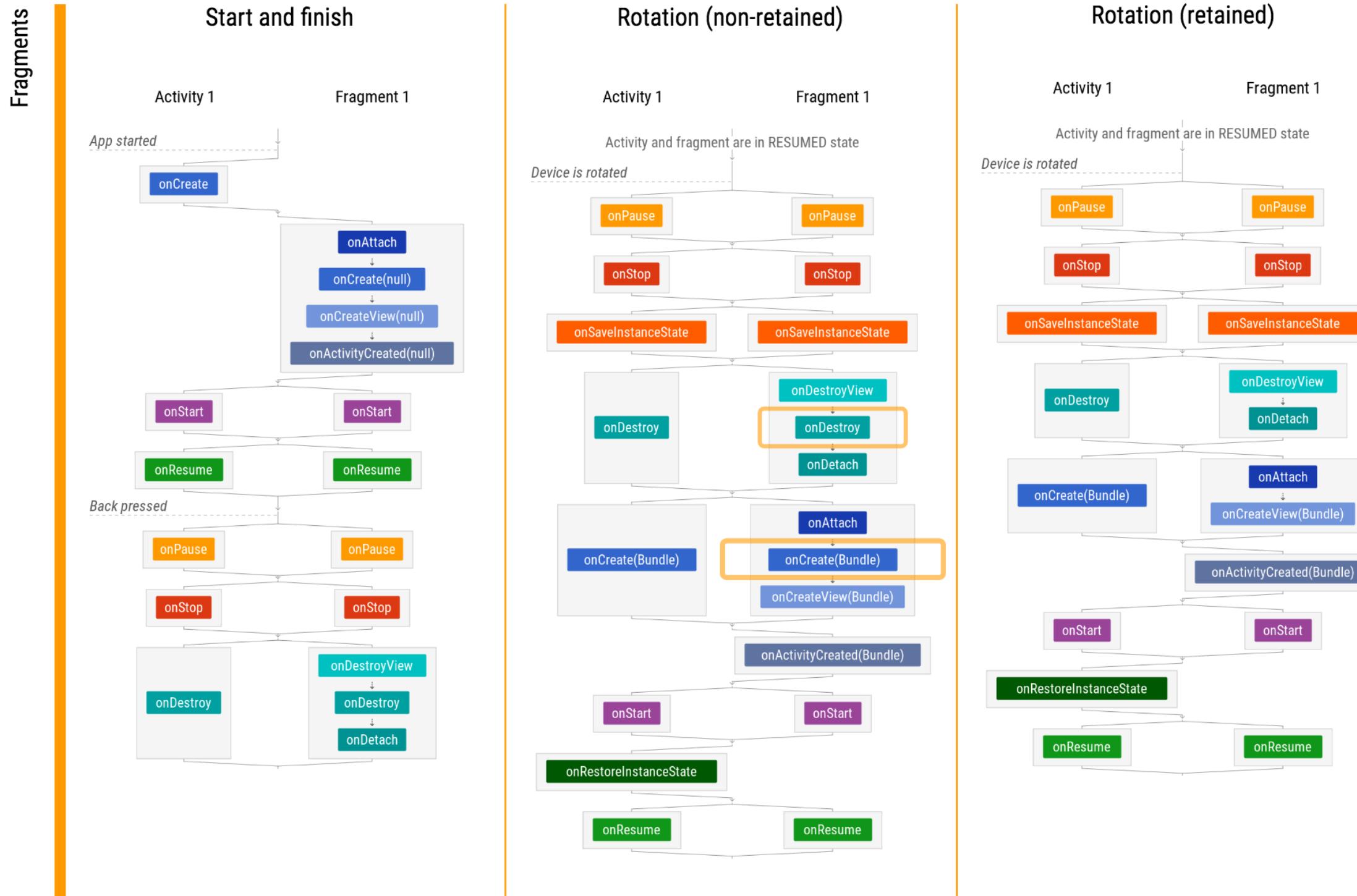
            // Add the fragment to the 'fragment_container' FrameLayout
            getSupportFragmentManager()
                .beginTransaction()
                .add(R.id.fragment_container, firstFragment)
                .commit();
        }
    }
}
```

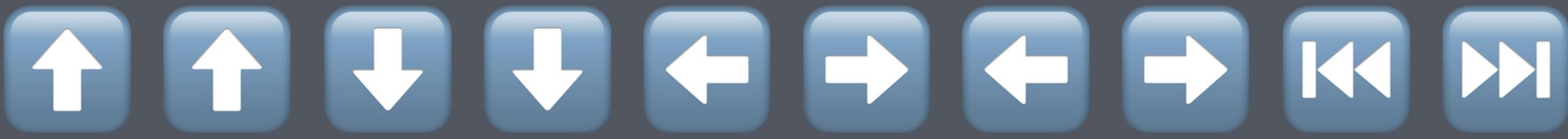
@askashdavies

Activity Lifecycle != Fragment Lifecycle

Activity Lifecycle !=
Fragment Lifecycle !=
View Lifecycle







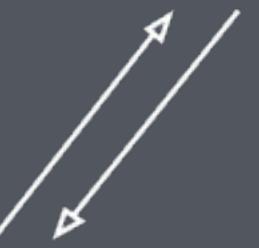
@askashdavies

@askashdavies



(supportFragmentManager || childFragmentManager)?

Activity



Fragment 2

Bundles



FragmentFactory

androidx.fragment:fragment:1.1.0

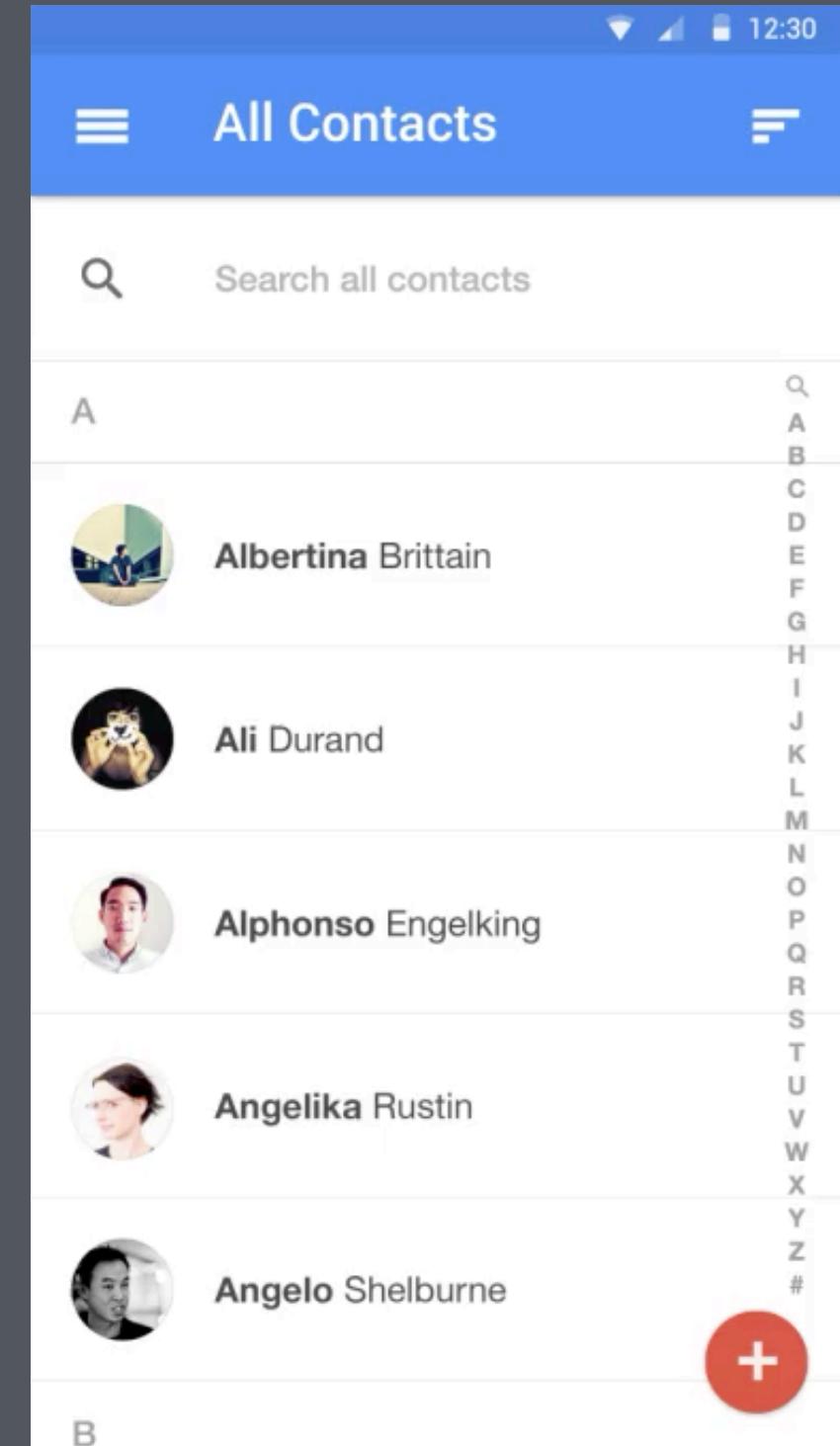
Fragments? 🐱

Fragments? 🤔

Activity Capabilities



Shared Element Transition (API 21+)



CompatShims < Enough

~~android.app.Fragment~~
androidx.fragment.app.Fragment

Scopes



Scopes

Activity 1

Activity 2

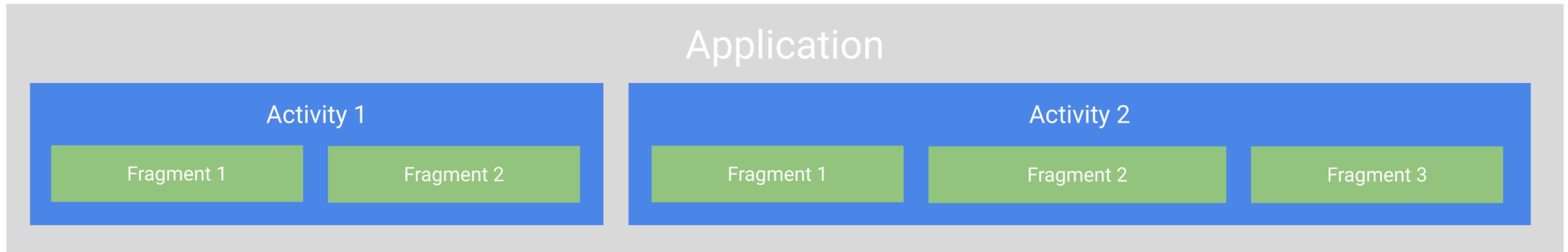
Scopes

Application

Activity 1

Activity 2

Scopes





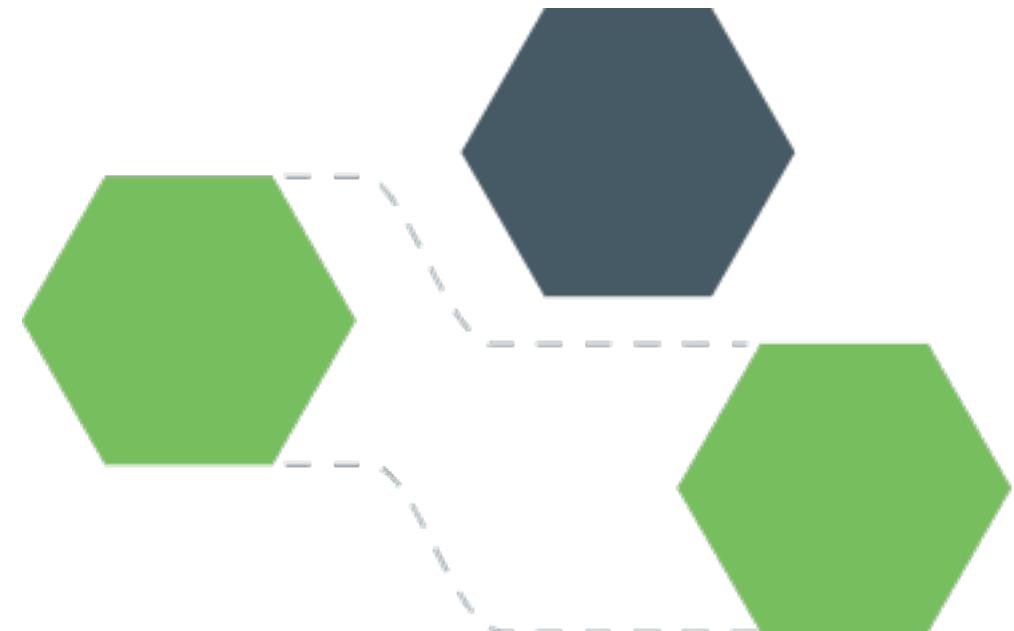
@askashdavies



@askashdavies

Android JetPack

Foundation Components



Android JetPack

Architecture Components



Android JetPack

Behaviour Components



Android JetPack

UI Components



Android JetPack

Navigation



Android JetPack: Navigation

Libraries



Android JetPack: Navigation

Plugin 

Android JetPack: Navigation

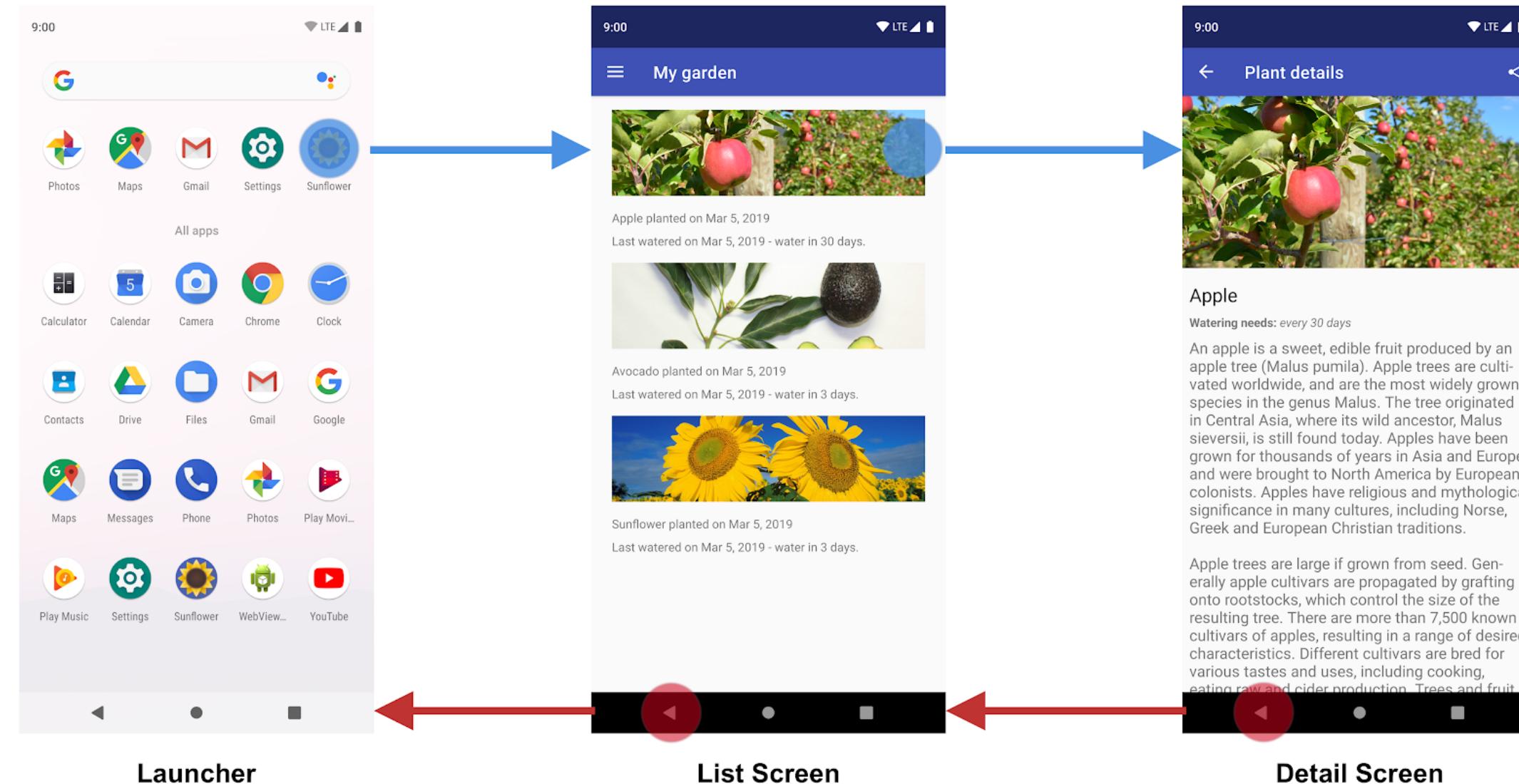
Tooling 

Android JetPack: Navigation

- Fragment transactions
- Up and back actions
- Standardised transition resources
- Deep link implementation
- Easy navigation patterns
- Type safe arguments

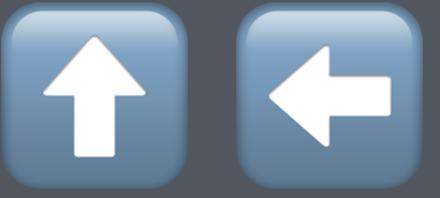
Principles of Navigation

Fixed Start Destination



State as Stack





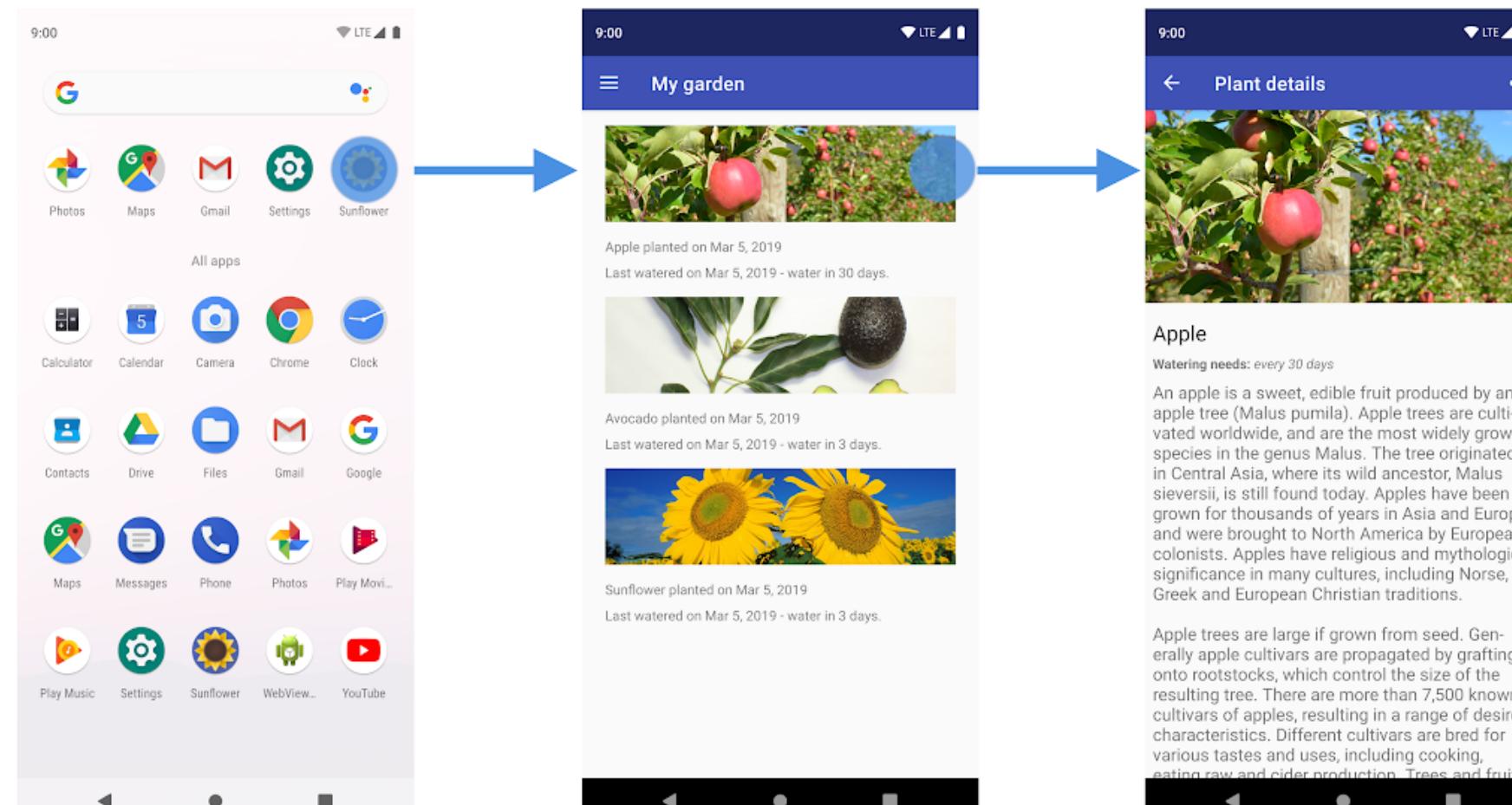
@askashdavies



@askashdavies

Deep Link Simulates Manual Navigation

Organic Navigation through Sunflower (Browsing to apple details)

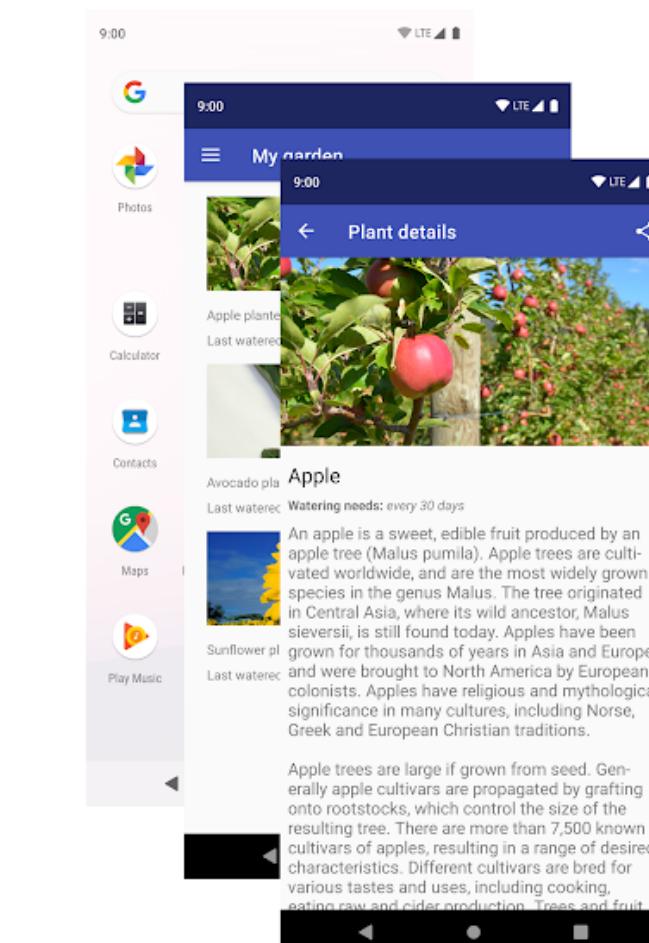


Launch Screen

List Screen

Detail Screen

Resulting Sunflower Task Back Stack

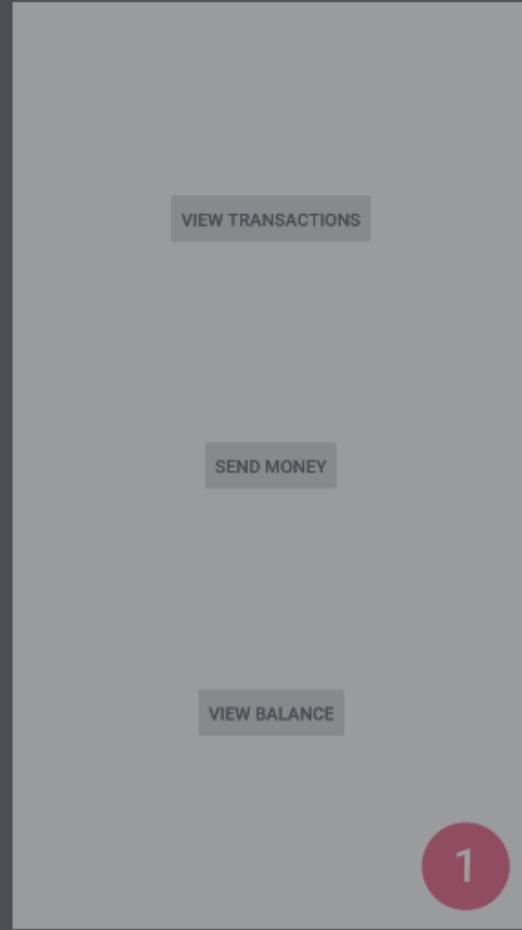


Android JetPack

Navigation



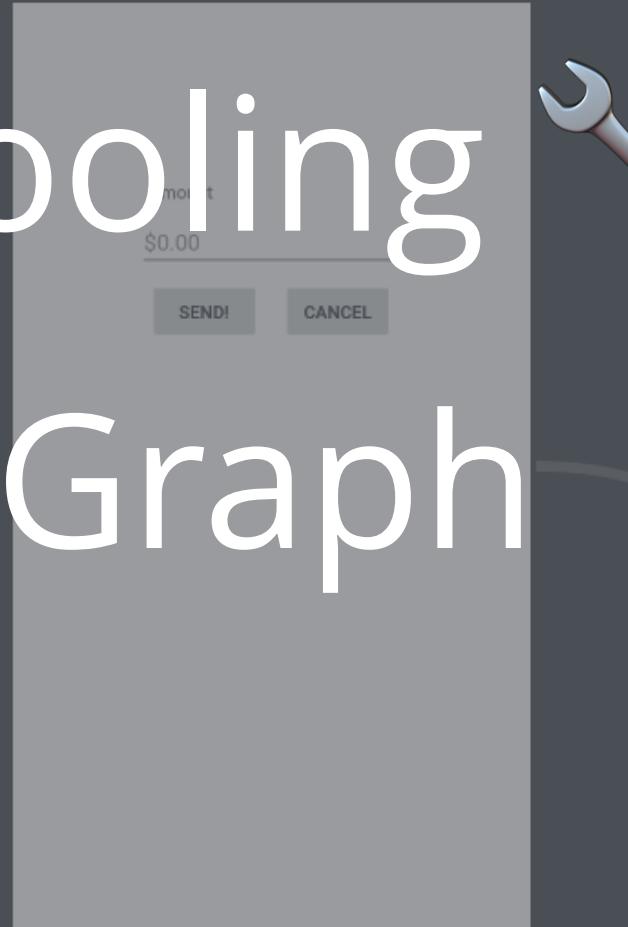
fragmentMain



fragmentChooseRecipient...



fragmentSpecifyAmount

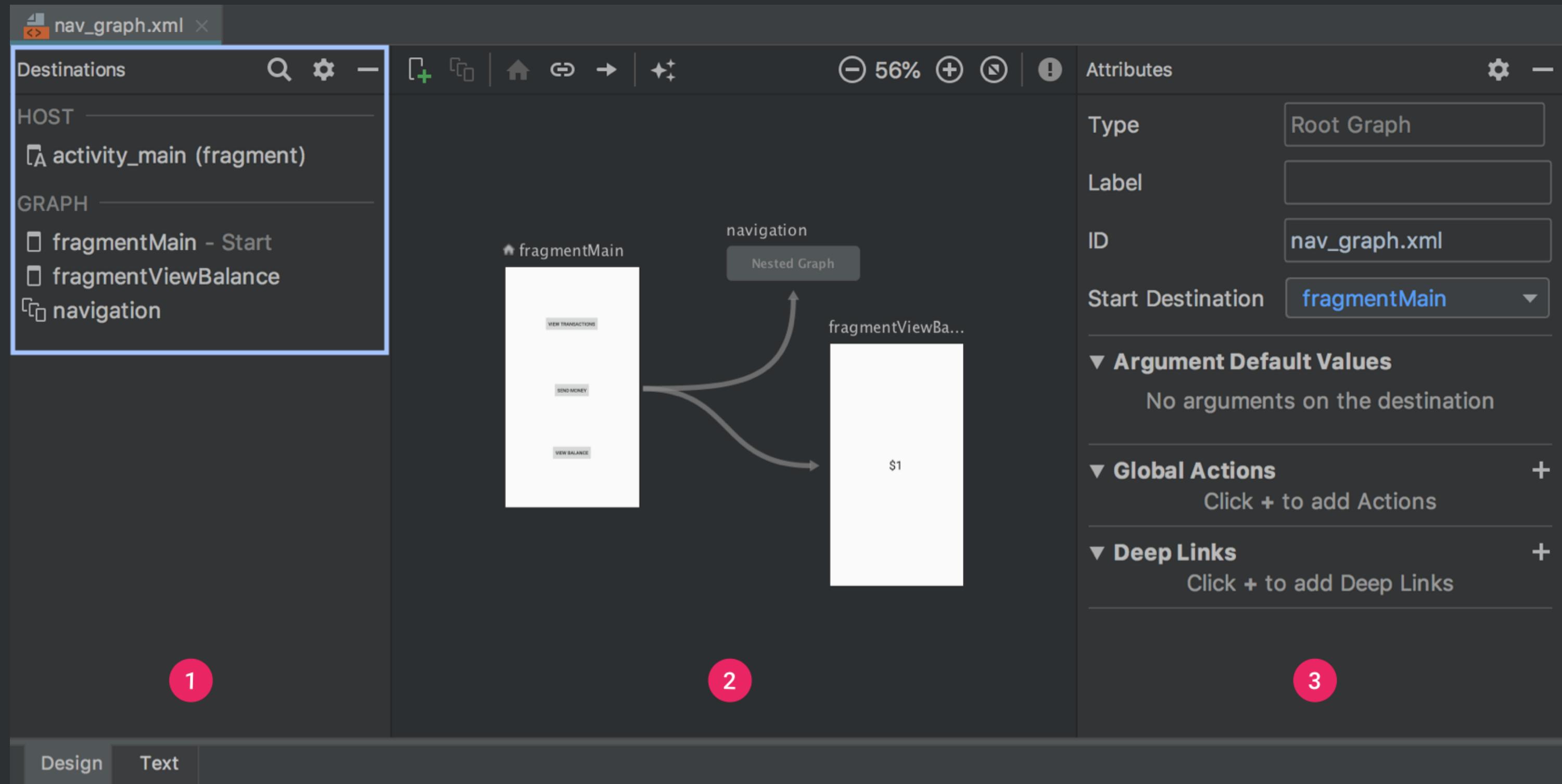


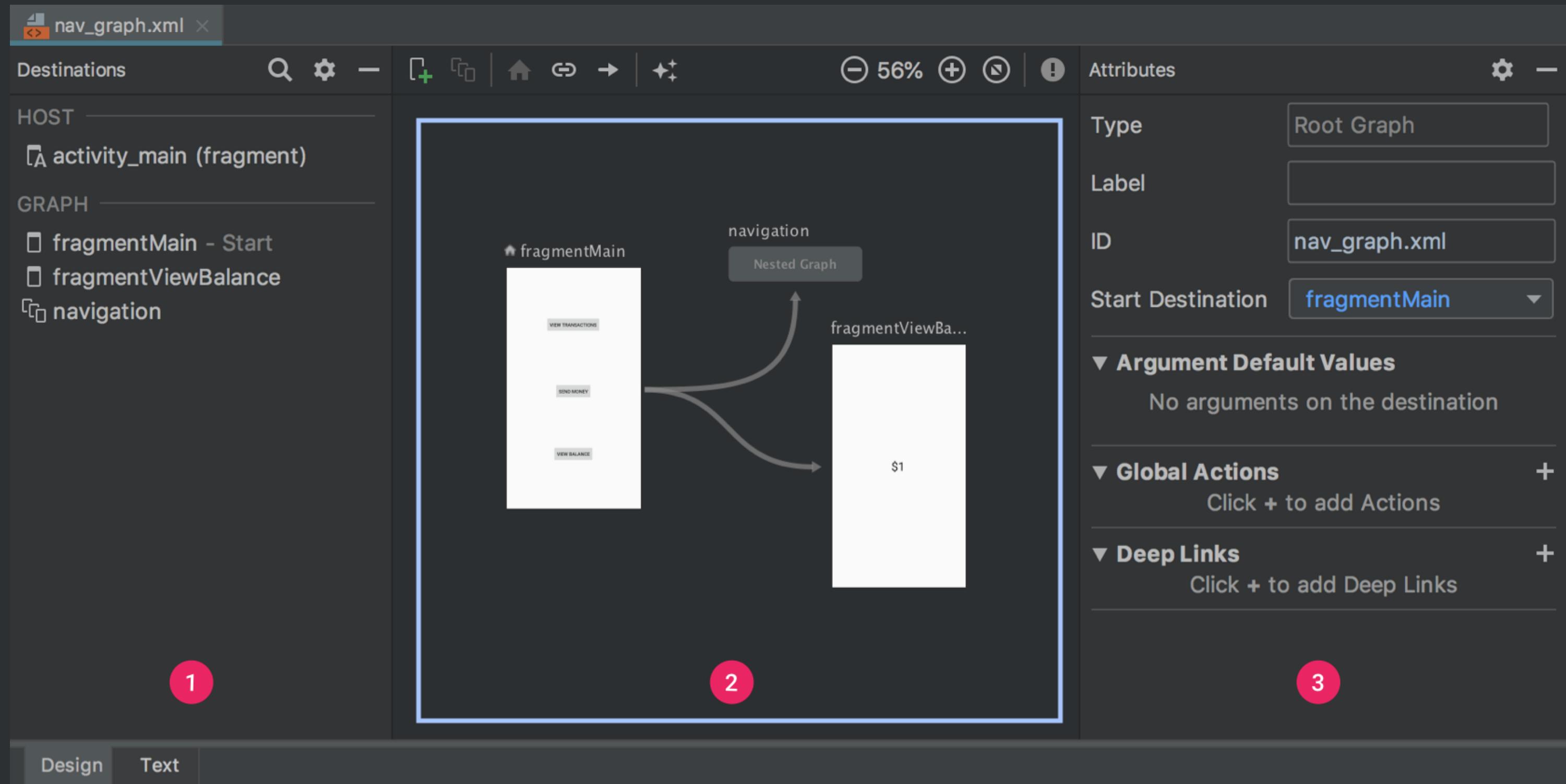
fragmentConfirmation

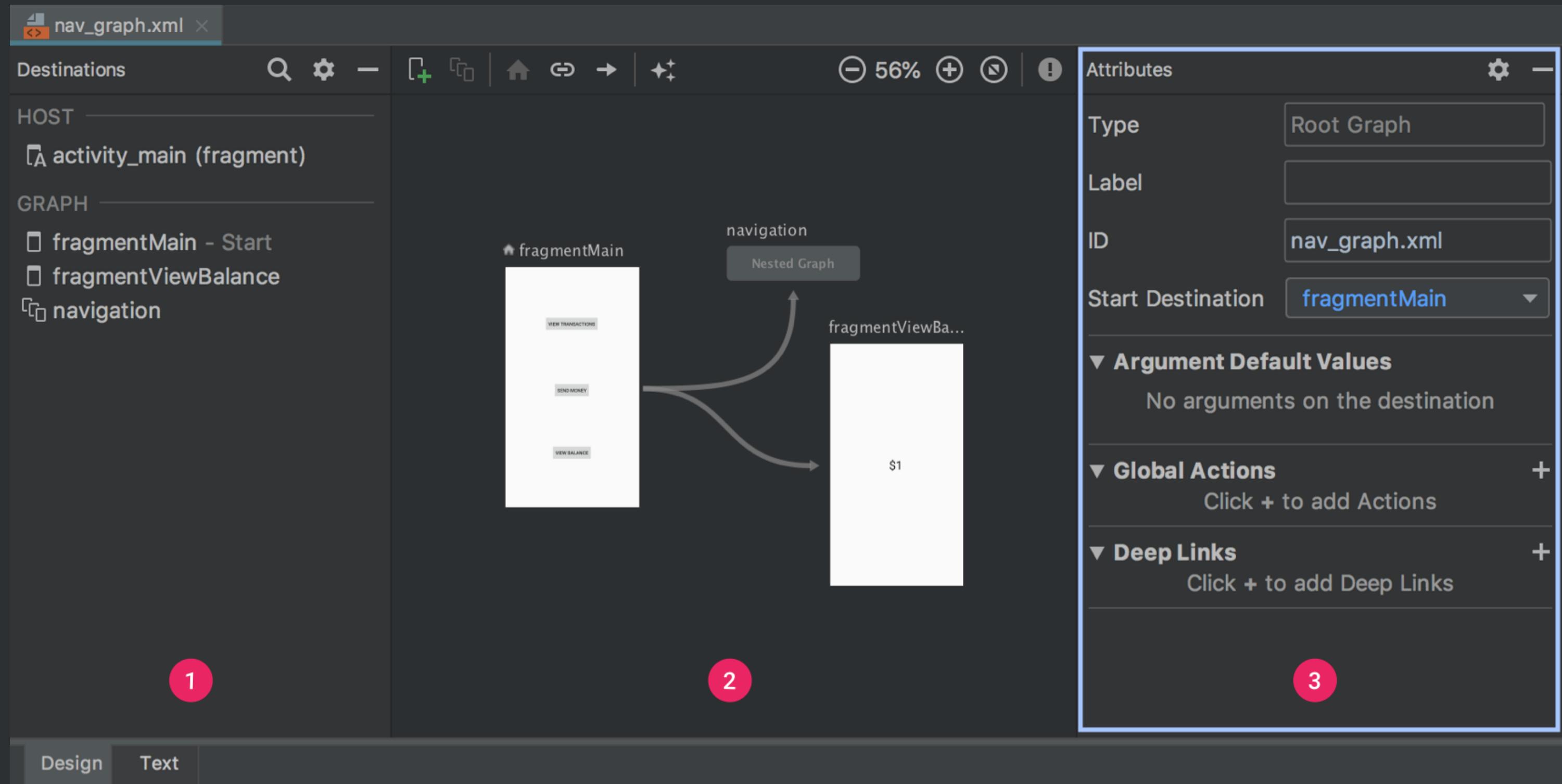


Navigation: Tooling

Navigation Graph







nav_graph.xml

```
<navigation xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/navigation"
    app:startDestination="@+id/fragmentMain">

    <fragment
        android:id="@+id/fragmentMain"
        android:name="com.google.sample.MainFragment"
        android:label="@string/main_title"
        tools:layout="@layout/main_fragment">

        <action
            android:id="@+id/mainToViewBalance"
            app:destination="@+id/fragmentViewBalance"/>

    </fragment>

    <fragment
        android:id="@+id/fragmentViewBalance"
        android:name="com.google.sample.ViewBalanceFragment"
        android:label="@string/view_balance_title"
        tools:layout="@layout/view_balance_fragment" />

</navigation>
```

Destination Types

- Activity <activity>
- Fragment <fragment>
- Dialog <dialog> (2.1.0+)

Destination Types (Custom)

developer.android.com/guide/navigation/navigation-add-new

- Extend Navigator<T> with your type
- Provide Destination implementation
- Configure arguments after inflation
- Augment NavController

Resource Inflation



Navigation: Libraries



NavHostFragment

```
<fragment  
    android:id="@+id/nav_host_fragment"  
    android:name="androidx.navigation.fragment.NavHostFragment"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    app:defaultNavHost="true"  
    app:navGraph="@navigation/nav_graph" />
```

Navigation: Libraries



NavController

- Fragment.findNavController()
- View.findNavController()
- Activity.findNavController(viewId: Int)

Actions

```
viewTransactionsButton.setOnClickListener { view ->  
    view.findNavController().navigate(R.id.viewTransactionsAction)  
}
```

```
button.setOnClickListener(  
    Navigation.createNavigateOnClickListener(R.id.next_fragment, null)  
)
```

```
<activity
    android:name="com.example.android.GizmosActivity"
    android:label="@string/title_gizmos" >
    <intent-filter android:label="@string/filter_view_http_gizmos">
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="android.intent.category.BROWSABLE" />
        <!-- Accepts URIs that begin with "http://www.example.com/gizmos" -->
        <data android:scheme="http"
            android:host="www.example.com"
            android:pathPrefix="/gizmos" />
        <!-- note that the leading "/" is required for pathPrefix-->
    </intent-filter>
    <intent-filter android:label="@string/filter_view_example_gizmos">
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="android.intent.category.BROWSABLE" />
        <!-- Accepts URIs that begin with "example://gizmos" -->
        <data android:scheme="example"
            android:host="gizmos" />
    </intent-filter>
</activity>
```



HOW ABOUT

Deep Links



@askashdavies



OH, REALLY?

Deep Links

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapplication">

    <application ... >

        <activity name=".ProfileActivity" ...>
            ...
            <nav-graph android:value="@navigation/main.xml" />
            ...
        </activity>
    </application>
</manifest>
```

Deep Links

```
<fragment  
    android:id="@+id/profile"  
    android:name=".ProfileActivity">  
  
    <deepLink android:autoVerify="true"  
        app:uri="www.example.com/profile/{userId}" />  
  
</fragment>
```

Deep Links

```
<fragment  
    android:id="@+id/profile"  
    android:name=".ProfileActivity">  
  
<deepLink android:autoVerify="true"  
    app:uri="www.example.com/profile/?userId={userId}" />  
  
</fragment>
```



Deep Links

Optional Args

2.2.0-alpha02

Deep Links

Implicit Links

Deep Links

Explicit Links

Deep Links

Explicit Links

```
val pendingIntent = NavDeepLinkBuilder(context)
    .setGraph(R.navigation.nav_graph)
    .setDestination(R.id.android)
    .setArguments(args)
    .createPendingIntent()
```

Deep Links

Explicit Links

```
val pendingIntent = findNavController()  
    .createDeepLink()  
    .setDestination(R.id.android)  
    .setArguments(args)  
    .createPendingIntent()
```

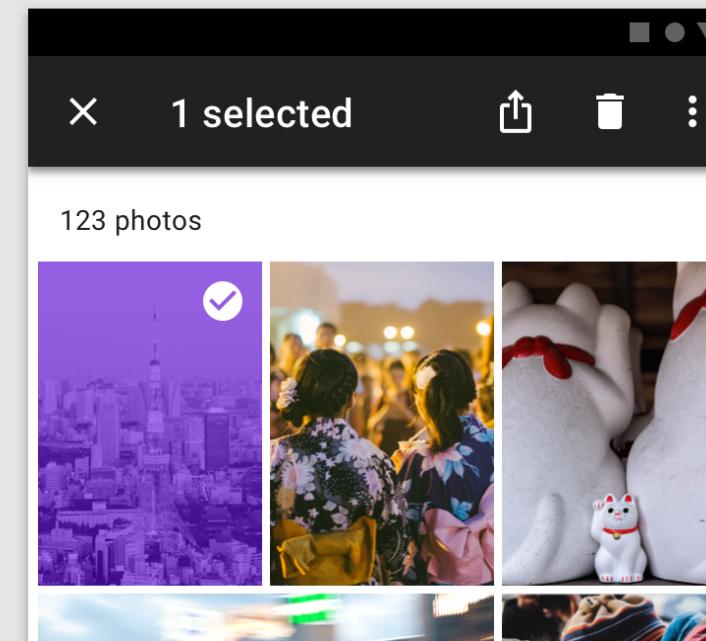
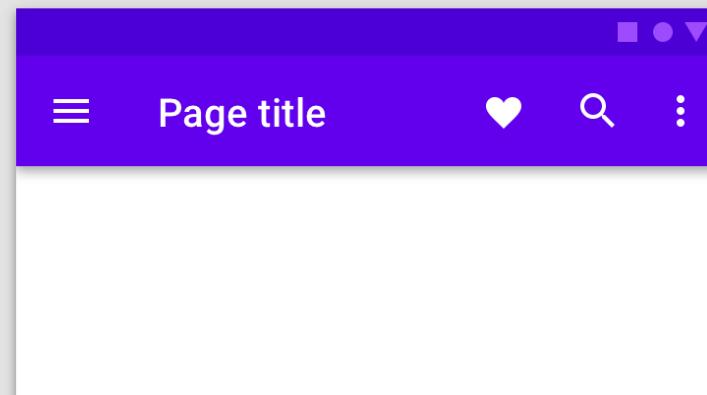
Navigation Styles



Navigation Styles



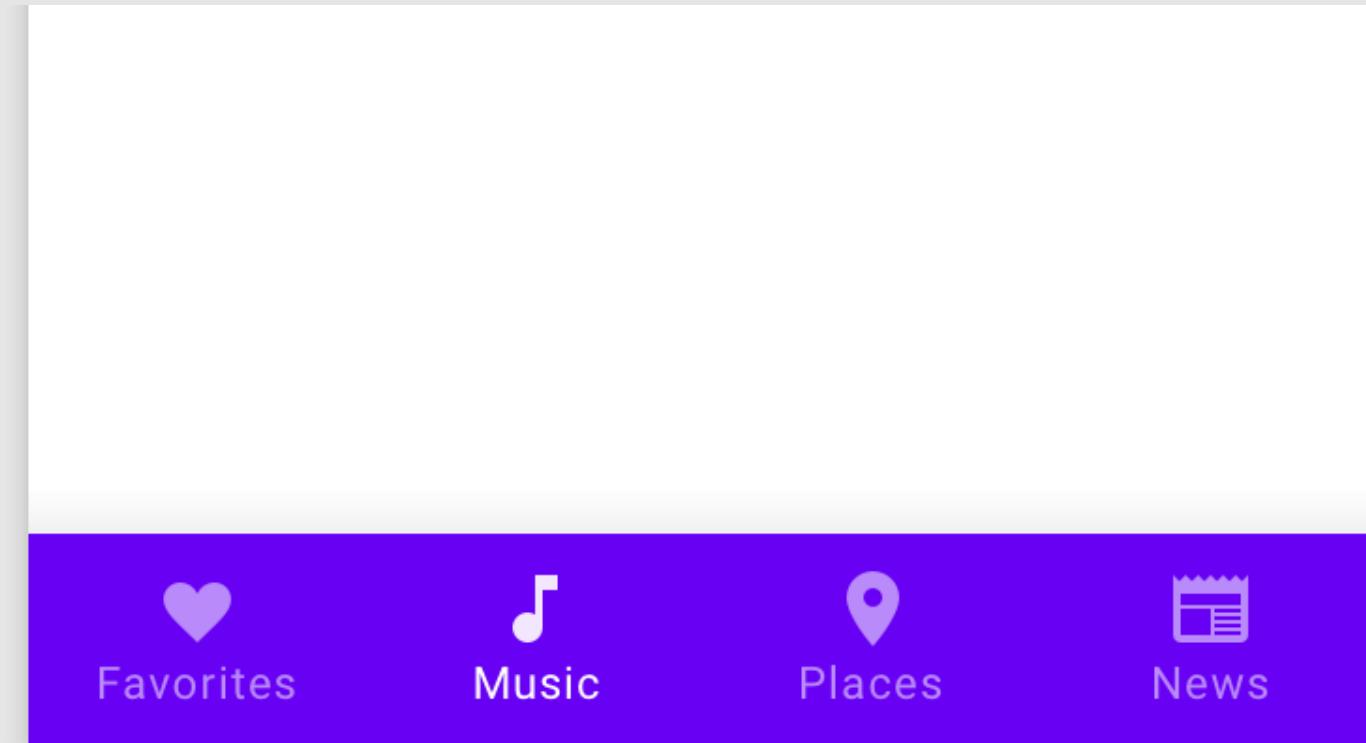
Toolbar



Navigation Styles



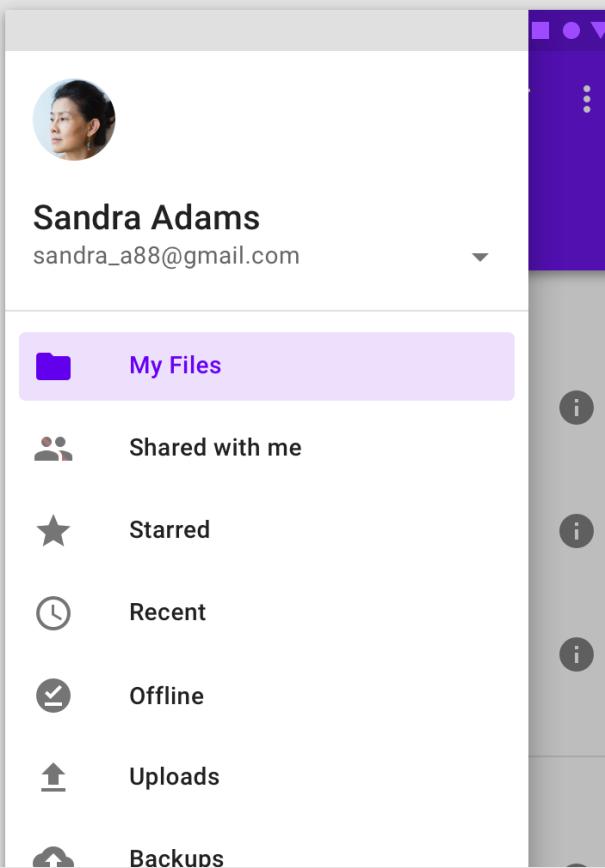
Bottom



Navigation Styles



Drawer



NavigationUI

setupWithNavController()

```
class AwesomeActivity : AppCompatActivity() {

    private val controller: NavController by lazy(NONE) { findNavController(this, R.id.host) }
    private val drawer: DrawerLayout by lazy(NONE) { findViewById<DrawerLayout>(R.id.drawer) }
    private val toolbar: Toolbar by lazy(NONE) { findViewById<Toolbar>(R.id.toolbar) }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.awesome_activity)

        toolbar.setupWithNavController(controller, drawer)
    }
}
```

NavigationUI

```
NavigationUI.setupActionBarWithNavController(  
    activity: AppCompatActivity,  
    controller: NavController,  
    configuration: AppBarConfiguration  
)
```

```
NavigationUI.setupWithNavController(  
    toolbar: Toolbar,  
    controller: NavController,  
    configuration: AppBarConfiguration  
)
```

NavigationUI: AppBarConfiguration

- Top level destinations
- Drawer layout
- Fallback "up" listener

OnDestinationChangedListener

NavController.addOnDestinationChangedListener

```
interface OnDestinationChangedListener {  
  
    fun onDestinationChanged(  
        controller: NavController,  
        destination: NavDestination,  
        arguments: Bundle?  
    );  
}
```

Bundles



Honourable Mention



Eugenio Marletti: Android Extras Delegates

github.com/Takhion/android-extras-delegates

Android Extras Delegates

```
class SomeActivity : Activity() {  
  
    companion object : ActivityCompanion<IntentOptions>(  
        intentOptions = IntentOptions,  
        kclass = SomeActivity::class  
    )  
  
    object IntentOptions {  
  
        var Intent.someExtra by IntentExtra.String()  
    }  
}
```

Plugin 

SafeArgs 

SafeArgs: Directions

MainFragmentDirections.mainToViewBalance()

```
<fragment
    android:id="@+id/fragmentMain"
    android:name="com.google.sample.MainFragment"
    android:label="@string/main_title"
    tools:layout="@layout/main_fragment">

    <action
        android:id="@+id/mainToViewBalance"
        app:destination="@+id/fragmentViewBalance"/>

</fragment>
```

SafeArgs: Directions

```
<fragment
    android:id="@+id/fragmentViewBalance"
    android:name="com.google.sample.ViewBalanceFragment"
    android:label="@string/view_balance_title"
    tools:layout="@layout/view_balance_fragment">

<argument
    android:name="balanceAmount"
    app:argType="integer"/>

</fragment>
```

SafeArgs: Args

```
class MainFragment: Fragment() {  
  
    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
        super.onViewCreated(view, savedInstanceState)  
  
        viewTransactionsButton.setOnClickListener { view ->  
            MainFragmentDirections.mainToViewBalance(100)  
        }  
    }  
}  
  
class ViewBalanceFragment : Fragment() {  
  
    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
        super.onViewCreated(view, savedInstanceState)  
  
        val args = ViewBalanceFragmentArgs.fromBundle(arguments!!)  
        balance.setText(String.format("%.2d", args.balanceAmount))  
    }  
}
```

SafeArgs: Generated Args



```
data class ViewBalanceFragmentArgs(val balanceAmount: Int) : NavArgs {
    @Suppress("CAST_NEVER_SUCCEEDS")
    fun toBundle(): Bundle {
        val result = Bundle()
        result.putInt("balanceAmount", this.balanceAmount)
        return result
    }

    companion object {
        @JvmStatic
        fun fromBundle(bundle: Bundle): FinanceBorrowerFragmentArgs {
            bundle.setClassLoader(FinanceBorrowerFragmentArgs::class.java.classLoader)
            val __balanceAmount : Int
            if (bundle.containsKey("balanceAmount")) {
                __balanceAmount = bundle.getInt("balanceAmount")
            } else {
                throw IllegalStateException("Required argument \\"balanceAmount\\" is missing and does not have an android:defaultValue")
            }
            return FinanceBorrowerFragmentArgs(__balanceAmount)
        }
    }
}
```

SafeArgs: Generated Directions



```
class MainFragmentDirections private constructor() {
    private data class MainToViewBalance(val balanceAmount: Int) : NavDirections {
        override fun getActionId(): Int = R.id.mainToViewBalance

        @Suppress("CAST_NEVER_SUCCEEDS")
        override fun getArguments(): Bundle {
            val result = Bundle()
            result.putInt("balanceAmount", this.index)
            return result
        }
    }

    companion object {
        fun mainToViewBalance(balanceAmount: Int): NavDirections = BorrowerToBorrower(index, financeBorrower)
    }
}
```

Single Source of Truth

SafeArgs: Activities



SafeArgs: Activities

```
class MainActivity : Activity {

    override fun onCreate(savedInstanceState: Bundle) {
        super.onCreate(savedInstanceState)

        val args = ViewBalanceActivityArgs(100)
        val intent = Intent(this, ViewBalanceActivity::class.java)

        startActivity(intent, bundle.toBundle())
    }
}
```

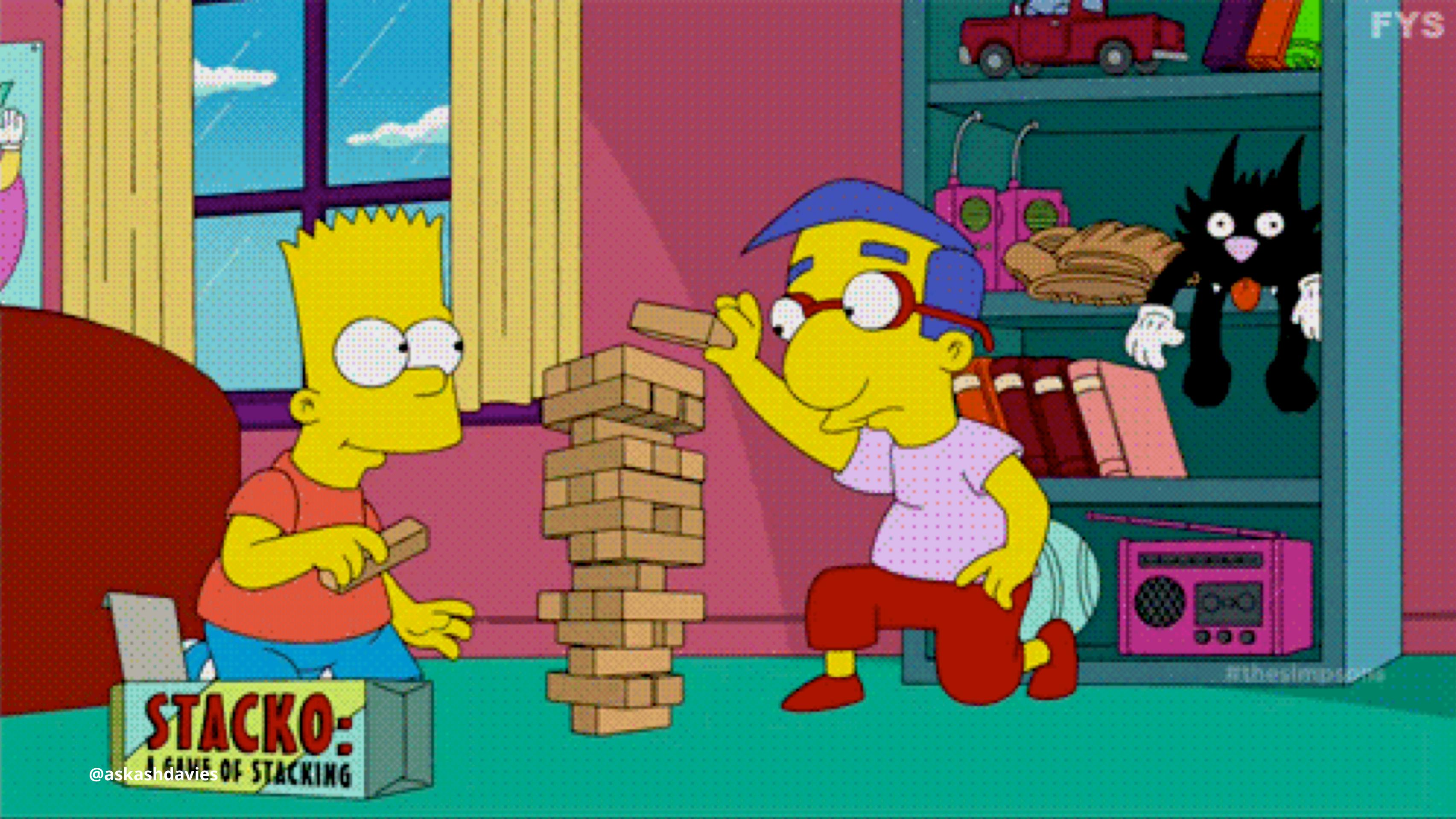
SafeArgs: Getting Started

```
buildscript {  
    repositories {  
        google()  
    }  
  
    dependencies {  
        classpath "androidx.navigation:navigation-safe-args-gradle-plugin:2.1.0"  
    }  
}  
  
apply plugin: 'androidx.navigation.safeargs.kotlin'
```



Migrating

FYS



@askashdavies



Migrating



Processed Cheddar



Aragon



Blue Vein Cheeses



Castigliano



Allgauer Emmentaler



Schloss



Roncal



Fondant de Brebis



White Stilton

@askashdavies



Camembert



Migrating



Stilton



Info

Jarlsberg lancashire edam. Dolcelatte hard cheese brie st. agur blue cheese caerphilly bavarian bergkase cheese and biscuits mascarpone. Cheeseburger swiss bavarian bergkase cream cheese fromage frais cheesy feet port-salut airedale. St. agur blue cheese rubber cheese caerphilly cheddar cheesecake cream cheese manchego lancashire. Roquefort squirty cheese the big cheese.

Friends

Jarlsberg lancashire edam. Dolcelatte hard cheese brie st. agur blue cheese caerphilly bavarian bergkase cheese and biscuits mascarpone. Cheeseburger swiss bavarian bergkase cream cheese fromage frais cheesy feet port-salut airedale. St. agur blue cheese rubber cheese caerphilly cheddar cheesecake cream cheese manchego lancashire. Roquefort squirty cheese the big cheese.



Migrating

Move screen behaviour away from activities



Migrating

Create new activity for Fragment's

 Migrating

Move existing activity logic to fragment

- FragmentBindings -> ActivityBindings
- onCreate() -> onCreateView()
- onViewCreated()
- getViewLifecycleOwner()

 Migrating

Initialise fragment in host activity

Initialise fragment in host activity

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.main_activity)  
  
    if (savedInstanceState == null) {  
        supportFragmentManager  
            .beginTransaction()  
            .add(R.id.main_content, CheeseListFragment())  
            .commit()  
    }  
  
    fun navigateToDetails(productId: String) {  
        /* ... */  
    }  
}
```

Pass arguments as necessary

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.main_activity)

    if (savedInstanceState == null) {
        val fragment = CheeseListFragment() // 🧀
        fragment.arguments = intent.extras

        supportFragmentManager
            .beginTransaction()
            .add(R.id.main_content, fragment)
            .commit()
    }

    fun navigateToDetails(productId: String) {
        /* ... */
    }
}
```



Migrating

Create navigation graph

Create navigation graph

```
<navigation xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    app:startDestination="@+id/cheeseGraph">  
  
    <fragment  
        android:id="@+id/cheeseListFragment"  
        android:name="com.sample.CheeseListFragment"  
        android:label="@string/cheese_list_title"  
        tools:layout="@layout/cheese_list_fragment" />  
  
    <fragment  
        android:id="@+id/cheeseDetailsFragment"  
        android:name="com.sample.CheeseDetailsFragment"  
        android:label="@string/cheese_details_title"  
        tools:layout="@layout/cheese_details_fragment" />  
  
</navigation>
```

Create navigation host

```
<fragment  
    android:id="@+id/main_content"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:name="androidx.navigation.fragment.NavHostFragment"  
    app:navGraph="@navigation/main_graph"  
    app:defaultNavHost="true" />
```



Migrating

```
class MainHostActivity : AppCompatActivity() {  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.main_activity)  
    }  
}
```

Navigation Directions

```
<navigation xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:startDestination="@+id/cheeseListFragment">

    <fragment
        android:id="@+id/cheeseListFragment"
        android:name="com.sample.CheeseListFragment"
        android:label="@string/cheese_list_title"
        tools:layout="@layout/cheese_list_fragment">

        <action
            android:id="@+id/navigateToDetails"
            app:destination="@+id/cheeseDetailsFragment"/>

    </fragment>

    <fragment
        android:id="@+id/cheeseDetailsFragment"
        android:name="com.sample.CheeseDetailsFragment"
        android:label="@string/cheese_details_title"
        tools:layout="@layout/cheese_details_fragment">

        <argument
            android:name="itemId"
            app:argType="string" />

    </fragment>
</navigation>
```

CheeseListFragment 🧀

```
class CheeseListFragment : Fragment() {  
    ...  
    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
        binding  
            .productsList  
            .adapter = ListAdapter(clickCallback)  
    }  
    ...  
  
    // The callback makes the call to the activity to make the transition.  
    private val clickCallback = ClickCallback { item ->  
        val directions = MainDirections.navigateToDetails(item.id)  
        findNavController().navigate(directions)  
    }  
}
```



Profit

IMMOBILIEN
SCOUT24

Random Slides

Cheese



@askashdavies



@askashdavies

onActivityResult?

onActivityResult
issuetracker.google.com/issues/79672220

Jose Alcérreca: LiveData<Event<T>>

bit.ly/2YuSYXi

Dynamic Features

Coming Soon? ™

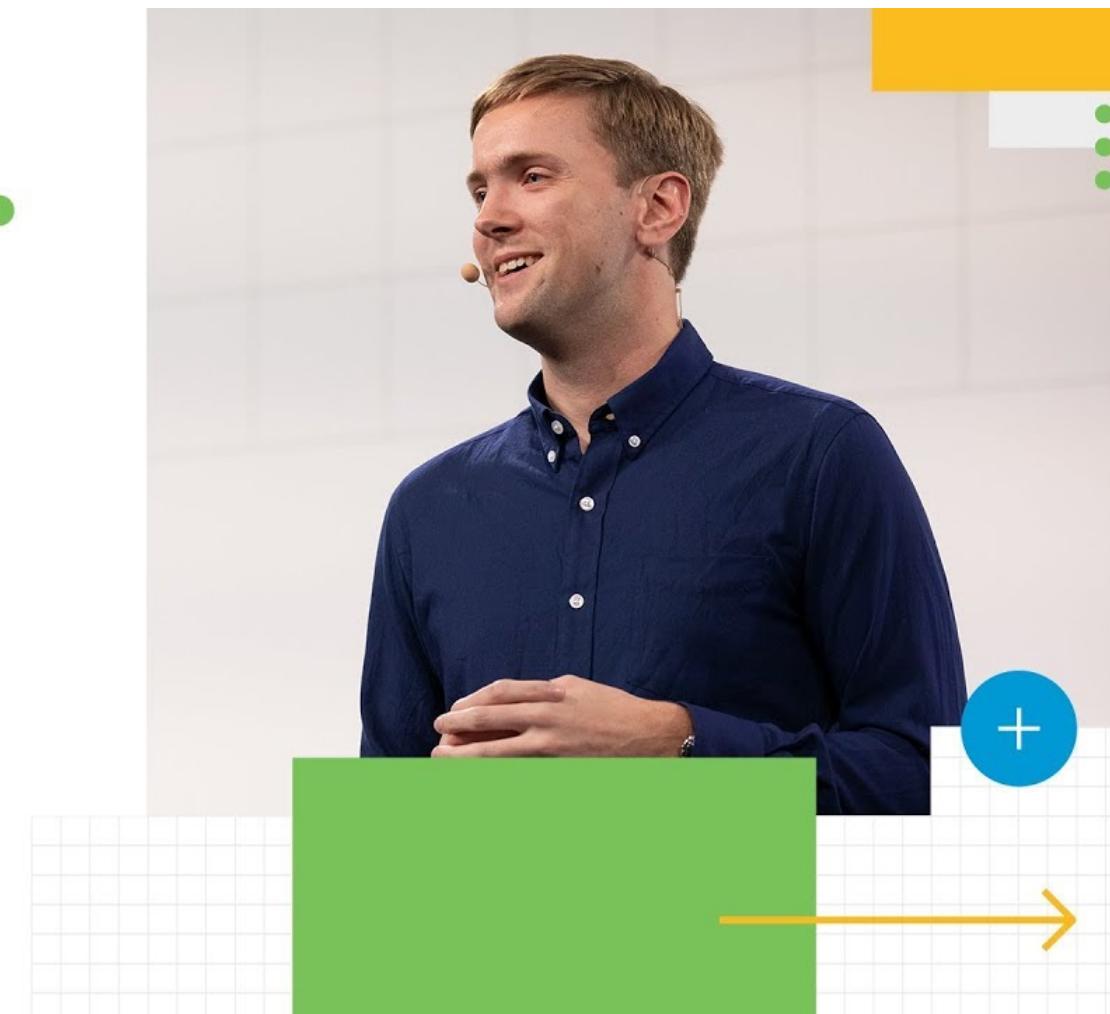
Ian Lake: Single Activity: Why, When, and How

bit.ly/2Jo94x9



 Android Dev Summit
Mountain View 2018

Android
Jetpack
Navigation



Thanks! 🍺

@askashdavies



Droidcon Lisbon



13:20 - RxJava & Coroutines: A Practical Analysis

17:40 - Implementing the Paging Library