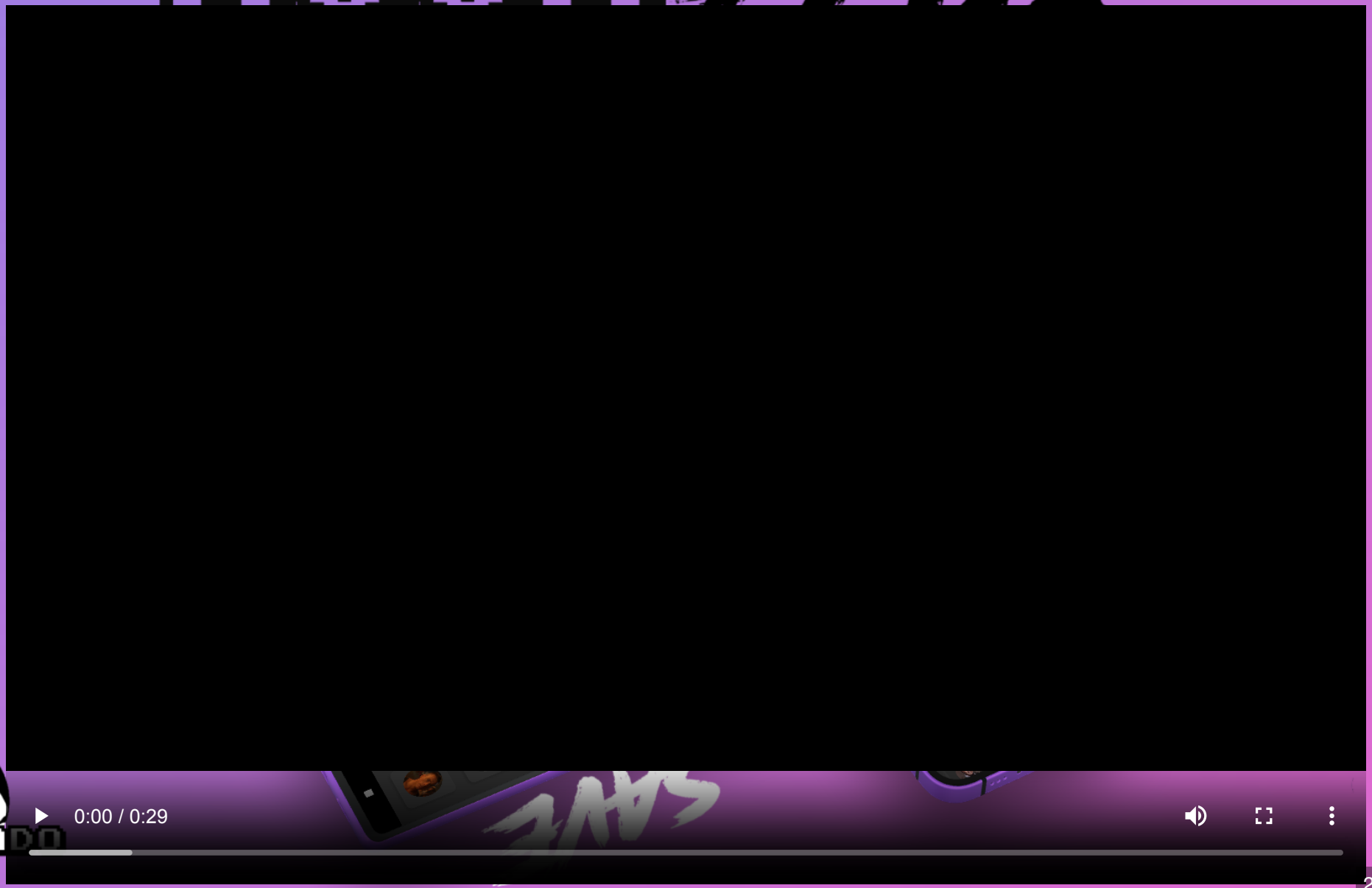
MVVM FLUX


LOAD
SEND
UPDATE
SAVE
CollectionView
Stars Dudes
Will Ferrell
Father Ted
Knights of Ni
Jean-Claude


0:00 / 0:29

SHARPNADO

Pragmatic components for the sad real world:

- CollectionView
- Tabs
- MaterialFrame
- MetroLog.Maui
- TaskLoaderView
- GridView with Drag and Drop

2024 CONSULTING SERVICES

- Best practices training
- MAUI Migration
- App audit and optimization
- Project startup

WHAT IS IT ABOUT?

- Immutability
- State propagation
- Async handling
- Error handling
- TaskLoaderView

MVVM (PRESENTATION)

- Model-View-ViewModel
- data binding and decoupling

What about view model loading and properties mutation ?

SIMPLIFIED DDD (BUSINESS)

- Entities
- Repositories
- Services
- ViewModels
- Views

What about entities mutation and data retrieval ?

MVVM.FLUX: A STATE ORCHESTRATION

All about filling the gaps!

- Coherent UI: the ui reflects the app state
- Coherent updates: the app state is always coherent
- A single source of truth

MVVM.FLUX: PRINCIPLES

- Composition over inheritance
- Single source of truth
- Immutability
- One way data flow

MVVM.FLUX: AN IMPLEMENTATION

- Composition: `TaskLoaderNotifier`
- Single source of truth: domain layer
- Immutability: `records`
- One way data flow: events/messages

Let's put it to the test!

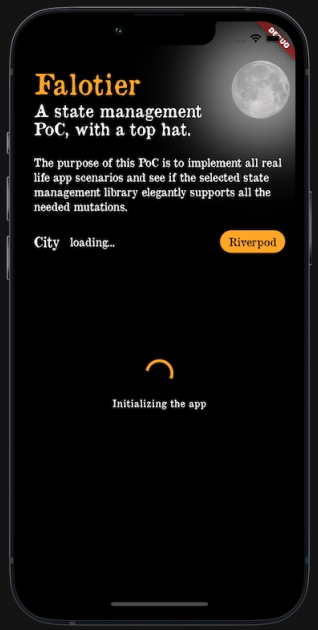
NOMINAL USE CASES

- Loading from scratch
- Refreshing
- Item update

Coherent UI: the ui reflects the app state

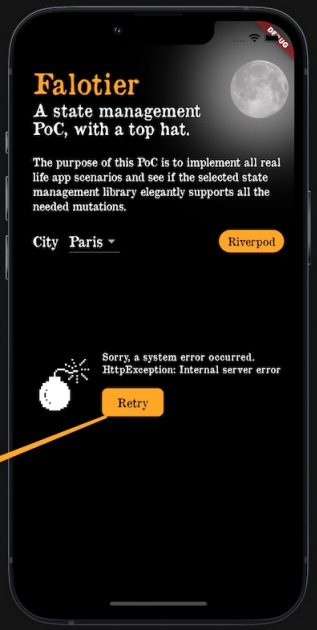
LOADING FROM SCRATCH

App Initialization

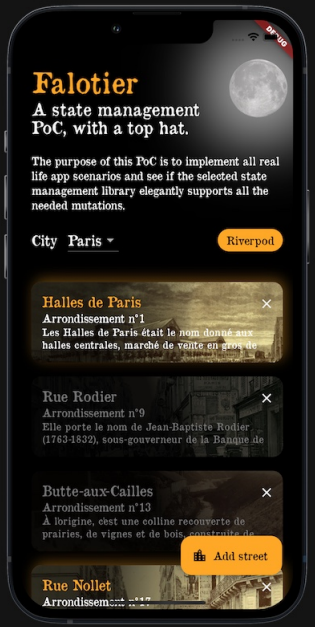


List Loading

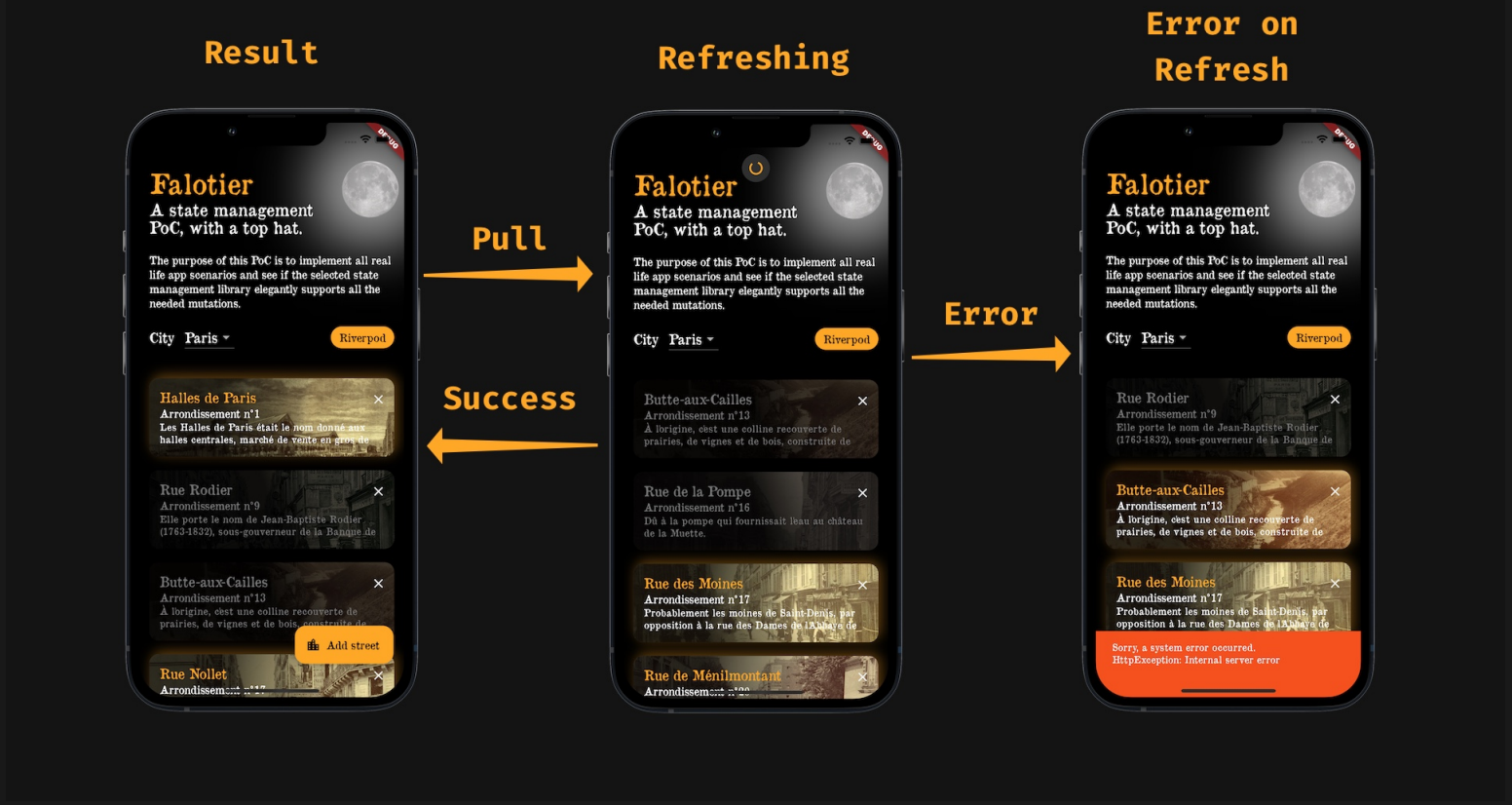
Error



Success



REFRESHING


Result
Refreshing
Error on Refresh
Pull
Success
Error
Falotier
A state management PoC, with a top hat.
The purpose of this PoC is to implement all real life app scenarios and see if the selected state management library elegantly supports all the needed mutations.
City Paris
Riverpod
Halles de Paris
Arrondissement n°1
Les Halles de Paris était le nom donné aux halles centrales, marché de vente en gros de
Rue Rodier
Arrondissement n°9
Elle porte le nom de Jean-Baptiste Rodier (1763-1832), sous-gouverneur de la Banque de
Butte-aux-Cailles
Arrondissement n°13
À l'origine, c'est une colline recouverte de prairies, de vignes et de bois, construite de
Add street
Rue Nollet
Rue de la Pompe
Arrondissement n°16
Dû à la pompe qui fournissait l'eau au château de la Muette.
Rue des Moines
Arrondissement n°17
Probablement les moines de Saint-Denis, par opposition à la rue des Dames de l'Abbaye de
Rue de Ménilmontant
Sorry, a system error occurred.
HttpException: Internal server error

UPDATING ITEM

LOADING FROM SCRATCH WITH MVVM.FLUX

- Sharpnado's TaskLoaderView
- Stop using `IsBusy=true` and all this nonsense
- Takes our `Task` state and create UI feedback
- Composition over inheritance

ISBUSY WAY IN RETRONADO

```
public override void OnNavigated(object parameter)
{
    _platform = (GamePlatform)parameter;

    Load();
}
```

```
private async void Load(bool isRefreshing = false)
{
    IsBusy = !isRefreshing;
    IsRefreshing = isRefreshing;
    HasError = false;
    HasRefreshError = false;
    ErrorMessage = string.Empty;

    ...
```

```
try
{
    Games = await GetGamesAsync();
}
catch (NetworkException)
{
    ErrorImageUrl = "Sample.Images.the_internet.png";
    ErrorMessage = SampleResources.Error_Network;
}
catch (ServerException)
{
    ErrorImageUrl = "Sample.Images.server.png";
    ErrorMessage = SampleResources.Error_Business;
}
...
```

```
    catch (Exception)
    {
        ErrorImageUrl = "Sample.Images.richmond.png";
        ErrorMessage = SampleResources.Error_Unknown;
    }
    finally
    {
        IsBusy = false;
        IsRefreshing = false;
        HasError = !isRefreshing && ErrorMessage != string.Em
        HasRefreshError = isRefreshing && ErrorMessage != str
    }
}
```

HOW TO REUSE CODE?

- Bad idea: tackle the issue with inheritance
- Good idea: create a component

WITH TASKLOADERNOTIFIER

```
public TaskLoaderNotifier<List<Game>> Loader { get; }

public override void OnNavigated(object parameter)
{
    _platform = (GamePlatform)parameter;

    // TaskStartMode = Manual (Default mode)
    Loader.Load(_ => GetGamesAsync());
}
```

LOADING FROM SCRATCH

`Retronado` Demo

Featuring: TaskLoaderView

REFRESHING

`Retronado` Demo

Featuring: TaskLoaderView

UPDATING ITEM

0:00 / 0:23

UPDATING: BAD IDEAS

- Share a entity by reference between VMs and just modify it
- VM to VM communication

BAD IDEAS

```
light.IsOn = true;
var updatedLight = await _lightService.Update(light);
_spectacularBus.Send(new UpdatedLightMessage(updatedLight));
```

- What if light instance is shared (caching)?
- What if light is updated in several places?
- How do we handle errors while updating?

UPDATING ITEM WITH MVVM.FLUX

- Use records
- Only the domain layer can propagate updates
- Use the `TaskLoaderCommand` to get UI feedback
- Group your UI states with a `CompositeTaskLoaderNotifier`

UPDATING ITEM WITH MVVM.FLUX

"Exclusive Lights" Demo

Featuring: records, CompositeTaskLoaderNotifier

TO TAKE AWAY

- Use `TemplatedTaskLoader` for async loading
- Use `TaskLoaderCommand` for updates
- Send update message from your services
- Use records for entities and the `with` syntax