

Zoom transitions

Douglas Hill

douglashill.co

iOS team lead at PSPDFKit

pspdfkit.com

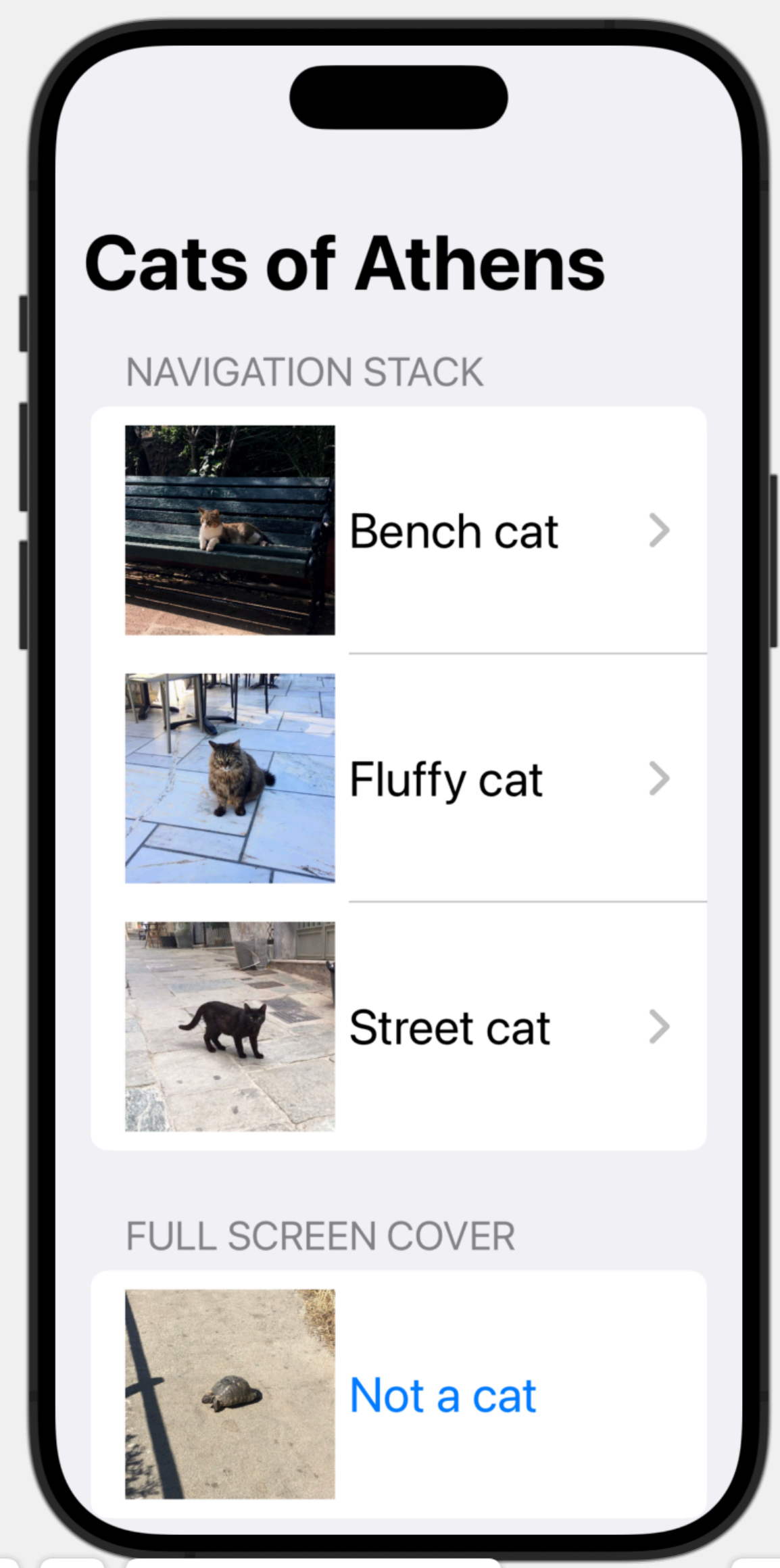
NSLondon organiser

nslondon.com

```
struct ContentView: View {
    let catPhotos: [Photo]
    let nonCatPhotos: [Photo]
    @State private var presentedPhoto: Photo?

    var body: some View {
        NavigationStack {
            List {
                Section(header: Text("Navigation Stack")) {
                    ForEach(catPhotos) { photo in
                        NavigationLink {
                            PhotoDetail(photo: photo)
                        } label: {
                            HStack {
                                PhotoThumbnail(photo: photo)
                                Text(photo.caption)
                            }
                        }
                    }
                }
                Section(header: Text("Full Screen Cover")) {
                    ForEach(nonCatPhotos) { photo in
                        Button {
                            presentedPhoto = photo
                        } label: {
                            HStack {
                                PhotoThumbnail(photo: photo)
                                Text(photo.caption)
                            }
                        }
                    }
                }
            }
        }
        .fullScreenCover(item: $presentedPhoto) { photo in
            NavigationStack {
```

ContentView



Zoomy

Zoomy > iPhone 15

Zoomy: Ready | Today at 11:35

Zoomy > Zoomy > ContentView > body

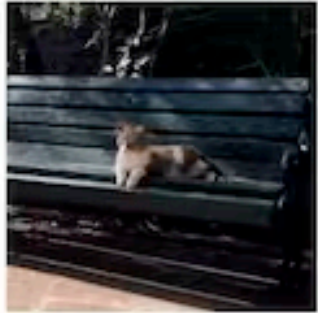
ContentView

```
struct ContentView: View {
    let catPhotos: [Photo]
    let nonCatPhotos: [Photo]
    @State private var presentedPhoto: Photo?

    var body: some View {
        NavigationStack {
            List {
                Section(header: Text("Navigation Stack")) {
                    ForEach(catPhotos) { photo in
                        NavigationLink {
                            PhotoDetail(photo: photo)
                        } label: {
                            HStack {
                                PhotoThumbnail(photo: photo)
                                Text(photo.caption)
                            }
                        }
                    }
                }
            }
            Section(header: Text("Full Screen Cover")) {
                ForEach(nonCatPhotos) { photo in
                    Button {
                        presentedPhoto = photo
                    } label: {
                        HStack {
                            PhotoThumbnail(photo: photo)
                            Text(photo.caption)
                        }
                    }
                }
            }
        }
        .fullScreenCover(item: $presentedPhoto) { photo in
            NavigationStack {
```


Cats of Athens

NAVIGATION STACK




Bench cat

>



Fluffy cat


>



Street cat

>

FULL SCREEN COVER



Not a cat

Automatic – iPhone 15 Pro

Zoomy

Zoomy > iPhone 15

Zoomy: Ready | Today at 11:38

Zoomy > Zoomy > ContentView > PhotoThumbnail

ContentView

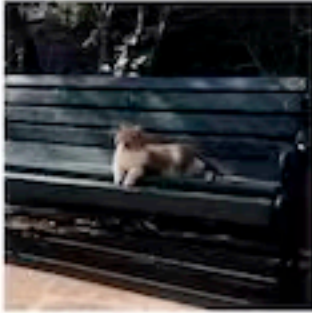
```
struct ContentView: View {
    let catPhotos: [Photo]
    let nonCatPhotos: [Photo]
    @State private var presentedPhoto: Photo?
    @Namespace private var namespace


    var body: some View {
        NavigationStack {
            List {
                Section(header: Text("Navigation Stack")) {
                    ForEach(catPhotos) { photo in
                        NavigationLink {
                            PhotoDetail(photo: photo)
                                .navigationTransition(.zoom(sourceID: photo, in: namespace))
                        } label: {
                            HStack {
                                PhotoThumbnail(photo: photo)
                                    .matchedTransitionSource(id: photo, in: namespace)
                                Text(photo.caption)
                            }
                        }
                    }
                }
            }
            Section(header: Text("Full Screen Cover")) {
                ForEach(nonCatPhotos) { photo in
                    Button {
                        presentedPhoto = photo
                    } label: {
                        HStack {
                            PhotoThumbnail(photo: photo)
                                .matchedTransitionSource(id: photo, in: namespace)
                            Text(photo.caption)
                        }
                    }
                }
            }
        }
    }
}
```


ContentView

Cats of Athens


NAVIGATION STACK

Bench cat >

Fluffy cat >

Street cat >

FULL SCREEN COVER

Not a cat

Automatic - iPhone 15 Pro

Line: 62 Col: 19

Zoomy

Zoomy > iPhone 15


Zoomy: Ready | Today at 11:38

Zoomy > Zoomy > ContentView > PhotoThumbnail


ContentView

Cats of Athens


NAVIGATION STACK



Bench cat >




Fluffy cat >



Street cat >

FULL SCREEN COVER



Not a cat

```
struct ContentView: View {
    let catPhotos: [Photo]
    let nonCatPhotos: [Photo]
    @State private var presentedPhoto: Photo?
    @Namespace private var namespace

    var body: some View {
        NavigationStack {
            List {
                Section(header: Text("Navigation Stack")) {
                    ForEach(catPhotos) { photo in
                        NavigationLink {
                            PhotoDetail(photo: photo)
                                .navigationTransition(.zoom(sourceID: photo, in: namespace))
                        } label: {
                            HStack {
                                PhotoThumbnail(photo: photo)
                                    .matchedTransitionSource(id: photo, in: namespace)
                                Text(photo.caption)
                            }
                        }
                    }
                }
            }
            Section(header: Text("Full Screen Cover")) {
                ForEach(nonCatPhotos) { photo in
                    Button {
                        presentedPhoto = photo
                    } label: {
                        HStack {
                            PhotoThumbnail(photo: photo)
                                .matchedTransitionSource(id: photo, in: namespace)
                            Text(photo.caption)
                        }
                    }
                }
            }
        }
    }
}
```

Automatic – iPhone 15 Pro

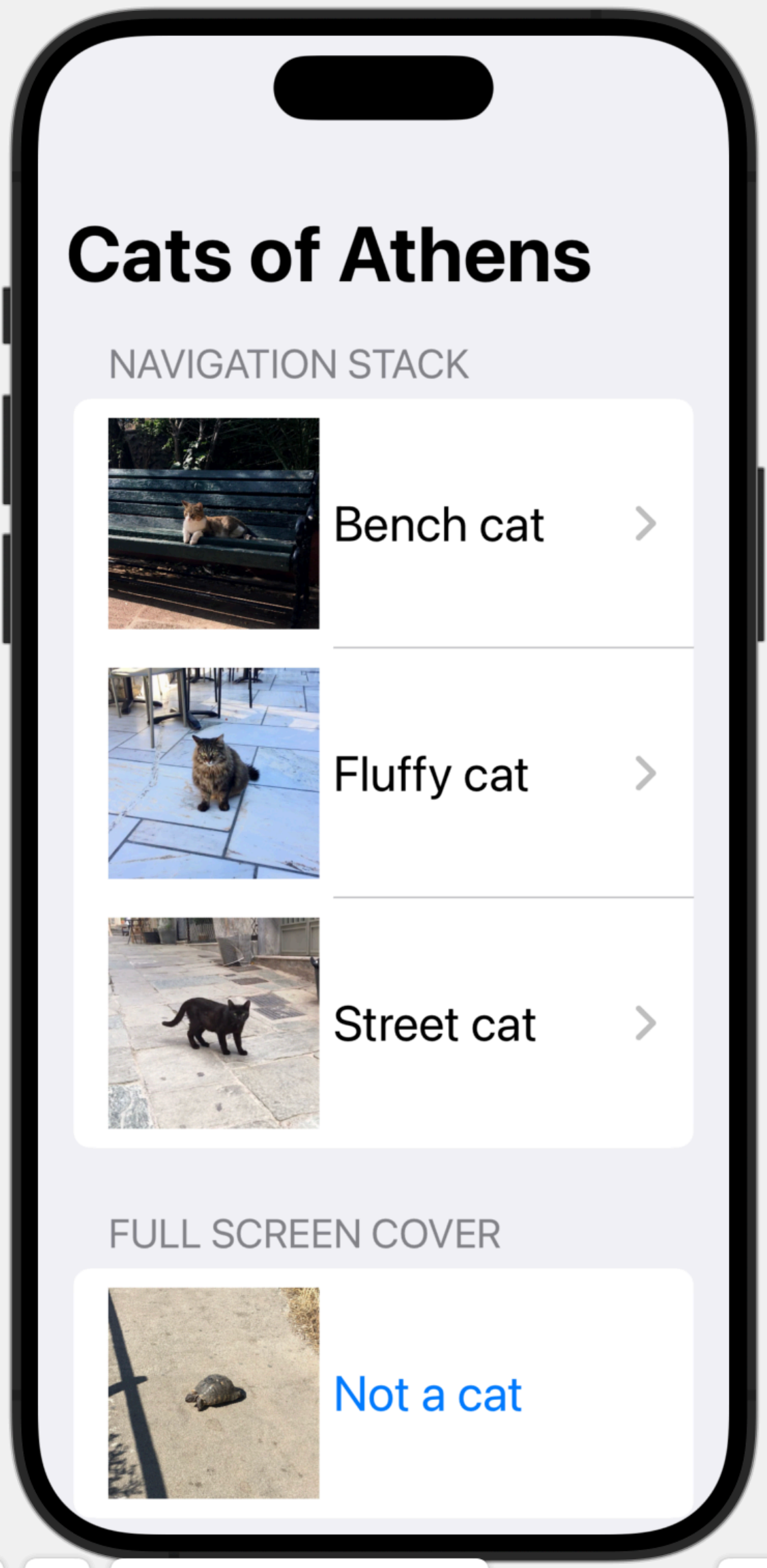
Line: 62 Col: 19

calculating transforms
edge swipe gesture recogniser
breaking layout by moving views
away from what they expected
coordinate space conversions
UIKitViewControllerTransitioningDelegate
four-component velocity
(x, y, scale, rotate)
pinch gesture recogniser
waiting for layout to settle
continuous velocity
converting between
velocity conventions
animating opacity
animating transforms
rotate gesture recogniser
trackpad/mouse scrolling
interruptible animations
UIKitViewControllerAnimatedTransitioning
reduce motion accessibility setting
UIKitViewControllerInteractiveTransitioning
swipe down gesture recogniser
gesture cancellation
spring animations
missing source view


```
struct ContentView: View {
    let catPhotos: [Photo]
    let nonCatPhotos: [Photo]
    @State private var presentedPhoto: Photo?
    @Namespace private var namespace

    var body: some View {
        NavigationStack {
            List {
                Section(header: Text("Navigation Stack")) {
                    ForEach(catPhotos) { photo in
                        NavigationLink {
                            PhotoDetail(photo: photo)
                                .navigationTransition(.zoom(sourceID: photo, in: namespace))
                        } label: {
                            HStack {
                                PhotoThumbnail(photo: photo)
                                    .matchedTransitionSource(id: photo, in: namespace)
                                Text(photo.caption)
                            }
                        }
                    }
                }
            }
            Section(header: Text("Full Screen Cover")) {
                ForEach(nonCatPhotos) { photo in
                    Button {
                        presentedPhoto = photo
                    } label: {
                        HStack {
                            PhotoThumbnail(photo: photo)
                                .matchedTransitionSource(id: photo, in: namespace)
                            Text(photo.caption)
                        }
                    }
                }
            }
        }
    }
}
```

ContentView




```
class CatsListViewController: UIViewController, UICollectionViewDelegate {
```

```
private func showDetailView(for photo: Photo) {
    let detailViewController = PhotoDetailViewController(photo: photo)

    detailViewController.preferredTransition = .zoom(sourceViewProvider: { context in
        let detailViewController = context.zoomedViewController as! PhotoDetailViewController
        let photo = detailViewController.photo

        guard let indexPath = self.dataSource.indexPath(for: photo) else {
            return nil
        }

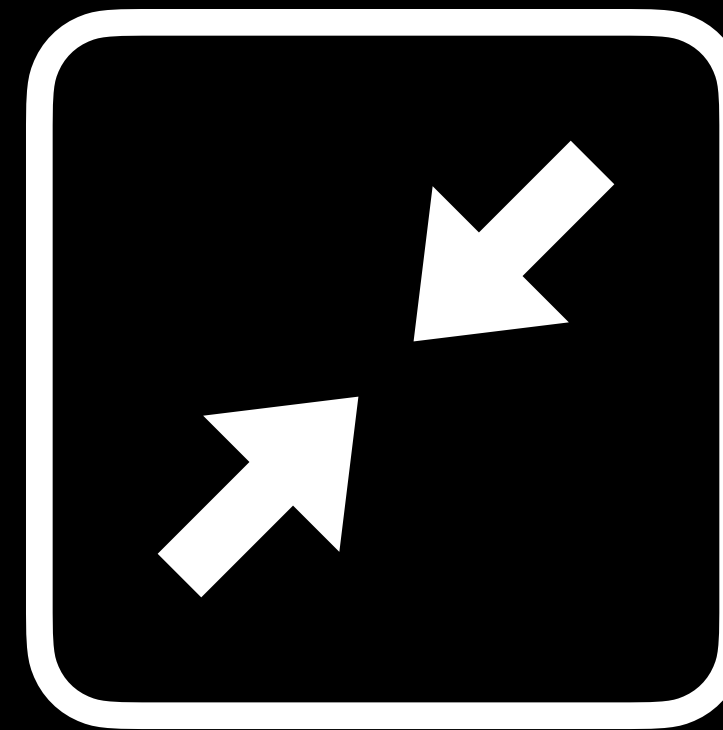
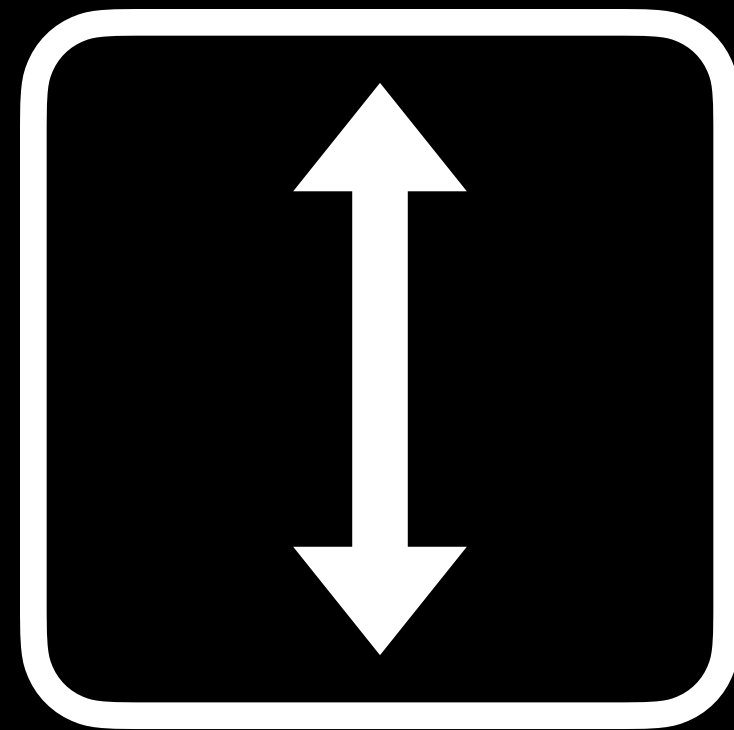
        guard let cell = self.collectionView.cellForItem(at: indexPath) else {
            return nil
        }

        return cell.imageView ?? cell.contentView
    })

    navigationController!.pushViewController(detailViewController, animated: true)
}
```

Gesture conflicts

- Your gestures take precedence
- Must be scrolled to the top and fully zoomed out
- No API to access the gesture recognisers



Platform availability

	API available	API implemented
iOS	✓	✓
visionOS	✓	
tvOS	✓	
watchOS	✓	
Mac Catalyst Scaled to Match iPad	✓	✓
Mac Catalyst Optimized for Mac	✓	
macOS		

Excellent animations

Zoomy

Zoomy > iPhone 16 (18.0)

Finished running Zoomy on iPhone 16

Zoomy > Zoomy > ContentView > No Selection

ContentView

Cats of Athens

NAVIGATION STACK

Bench cat

Fluffy cat

Street cat

FULL SCREEN COVER

Not a cat

AX 1

iPhone 16

Line: 60 Col: 2

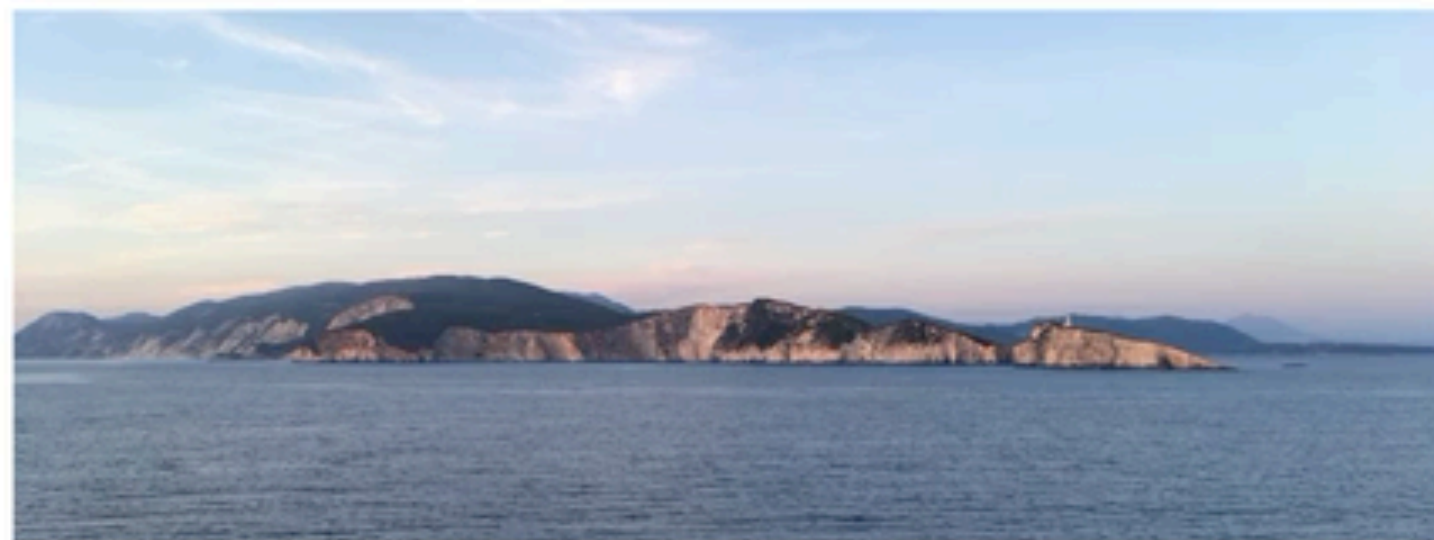
```
struct CatsListViewSwiftUI: View {
    let catPhotos: [Photo]
    let nonCatPhotos: [Photo]
    @State private var presentedPhoto: Photo?
    @Namespace private var namespace

    var body: some View {
        NavigationStack {
            List {
                Section(header: Text("Navigation Stack")) {
                    ForEach(catPhotos) { photo in
                        NavigationLink {
                            PhotoDetail(photo: photo)
                                .navigationTransition(.zoom(sourceID: photo, in: namespace))
                        } label: {
                            HStack {
                                Text(photo.caption)
                                Spacer(minLength: 0)
                            }
                            .matchedTransitionSource(id: photo, in: namespace)
                        }
                    }
                }
            }
            Section(header: Text("Full Screen Cover")) {
                ForEach(nonCatPhotos) { photo in
                    Button {
                        presentedPhoto = photo
                    } label: {
                        HStack {
                            Text(photo.caption)
                            Spacer(minLength: 0)
                        }
                        .matchedTransitionSource(id: photo, in: namespace)
                    }
                }
            }
        }
    }
}
```




Douglas Hill ...

London to Athens by train and ferry



Cape Doukato on Lefkada seen from the ferry at sunset

I recently arrived back in the UK after attending a wedding in Athens. To continue my four-year no-flying streak driven by concern about the environmental impact of air travel, I decided to go on an adventure to Greece and back over the land and sea while working remotely (as the iOS team lead at PSPDFKit).

In this post, I'll share a few photos and notes about the trip.



The return ferry waiting to leave from Patra in Greece.



I covered a lot of ground on the excellent Italian Frecciarossa trains.



London to Athens by train and



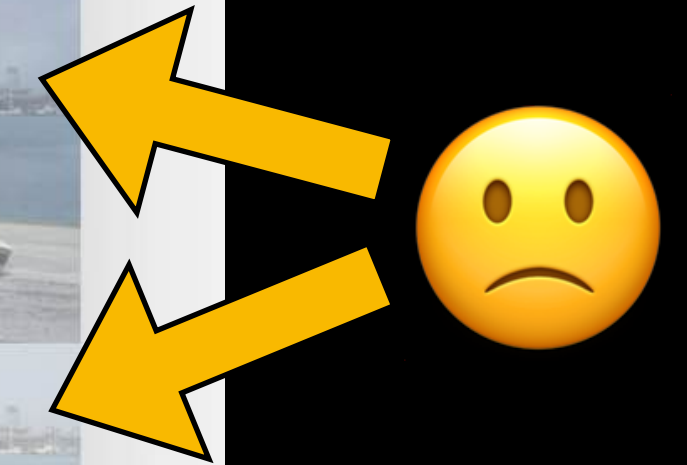
Cape Doukato on Lefkada seen from the

I recently arrived back in the UK after attending a conference in Athens. To continue my four-year no-fly streak, motivated by concern about the environmental impact of flying, I decided to go on an adventure to Greece and back over the land and sea while working remotely (as the iOS team lead at PSPDFKit).

In this post, I'll share a few photos and notes about the trip.



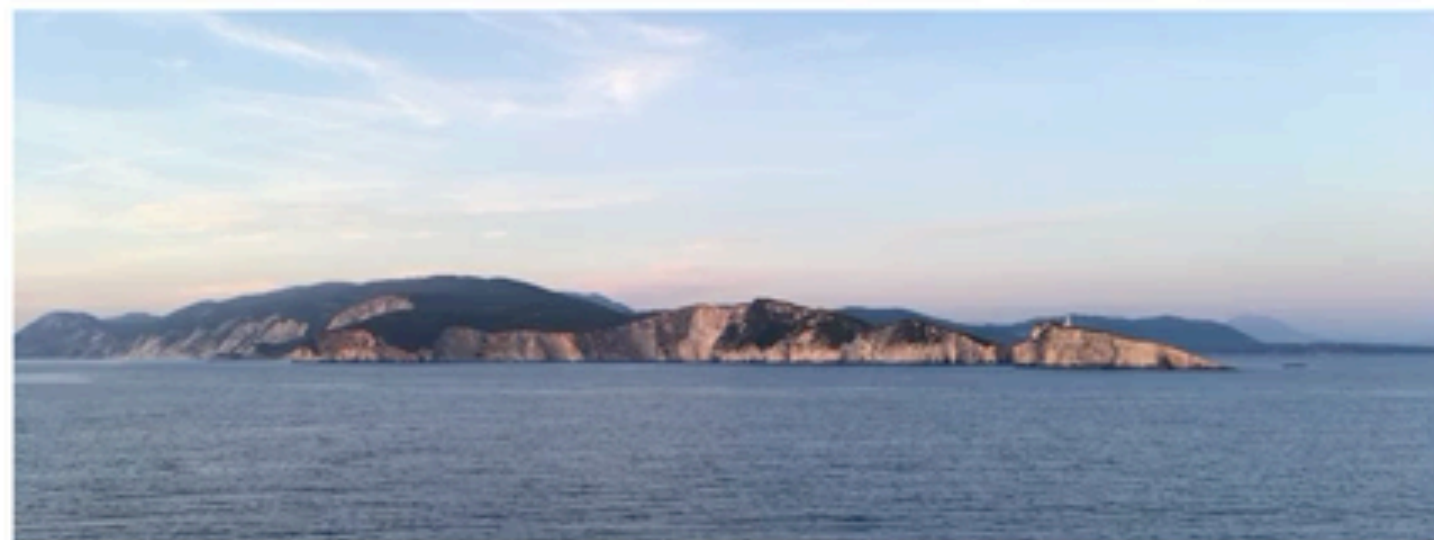
I covered a lot of ground on the excellent Italian Frecciarossa trains.





Douglas Hill ...

London to Athens by train and ferry



Cape Doukato on Lefkada seen from the ferry at sunset

I recently arrived back in the UK after attending a wedding in Athens. To continue my four-year no-flying streak driven by concern about the environmental impact of air travel, I decided to go on an adventure to Greece and back over the land and sea while working remotely (as the iOS team lead at PSPDFKit).

In this post, I'll share a few photos and notes about the trip.



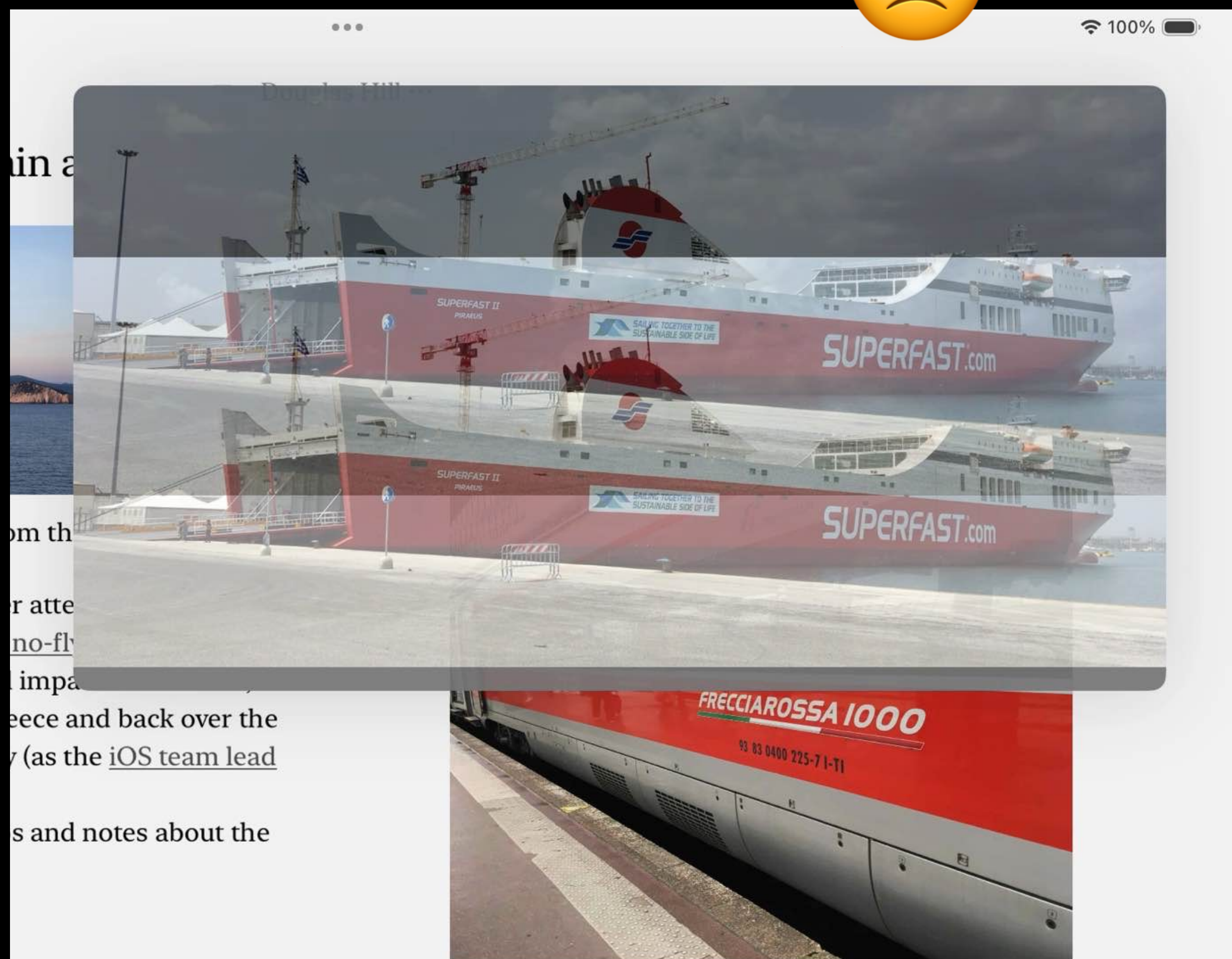
The return ferry waiting to leave from Patra in Greece.



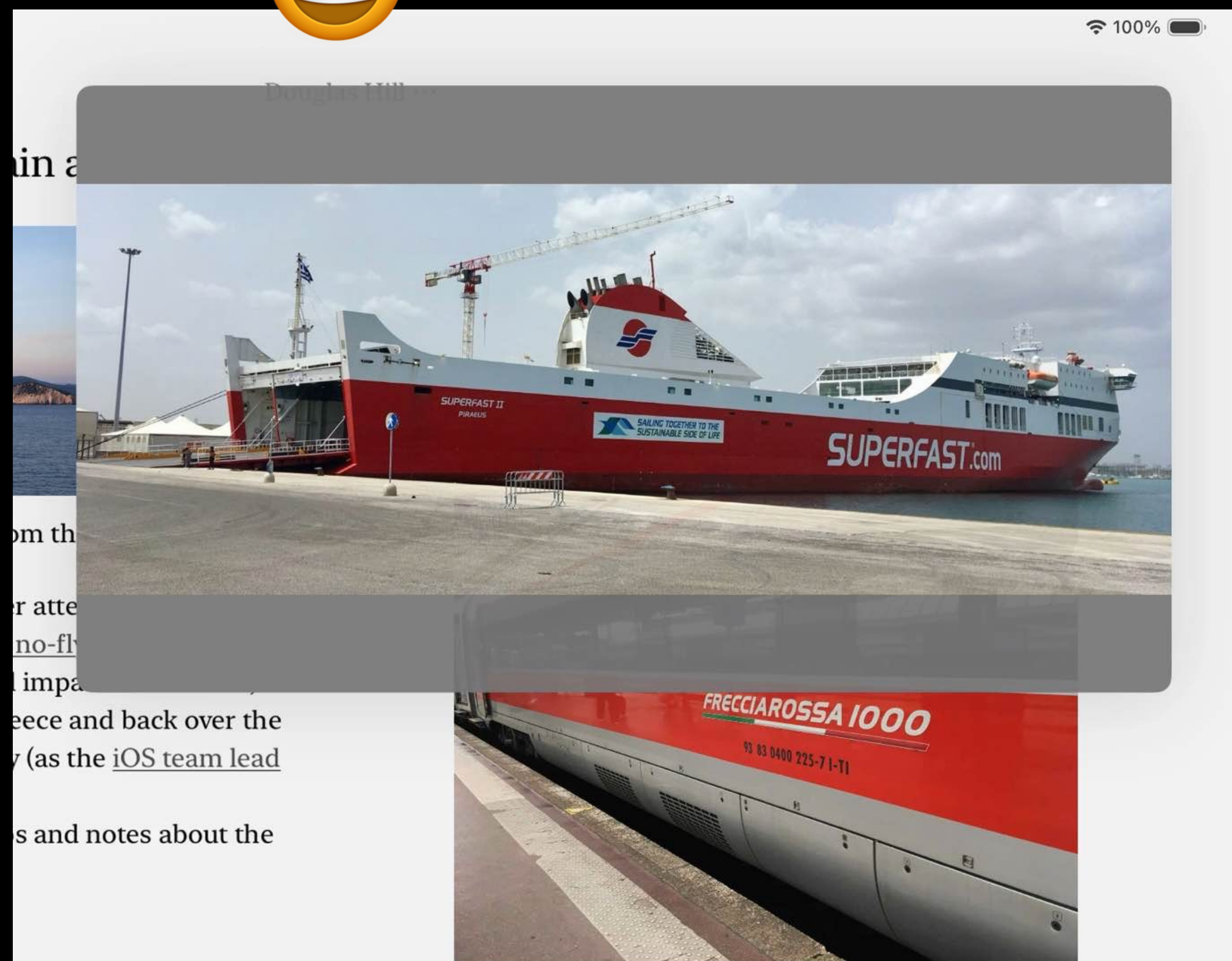
I covered a lot of ground on the excellent Italian Frecciarossa trains.

alignmentRectProvider

without 😞 with 😊



I covered a lot of ground on the excellent Italian Frecciarossa trains.



I covered a lot of ground on the excellent Italian Frecciarossa trains.

iOS SDK's zoom transition



Photos app



Zoom transitions

Douglas Hill

douglashill.co

iOS team lead at PSPDFKit

pspdfkit.com

NSLondon organiser

nslondon.com