

**Views: complexity
and reusability**

**Experiments comparing SwiftUI
and UIKit**

\$ whomami

- Felipe Espinoza
- iOS Dev @ FINN.no
- [@fespinoza on github](#)
- [@fespinozecast on twitter](#)

What do I mean by complexity?

Creating views in UIKit

- initialize
- position / size¹
- styling
- passing data
- update to changes

¹ size may depend on content

```
class SampleViewController: UIViewController {
    override func viewDidLoad() {
        super.viewDidLoad()
        view.backgroundColor = .red

        let label = UILabel()
        label.translatesAutoresizingMaskIntoConstraints = false
        label.text = "Hello"

        let button = UIButton()
        button.setTitle("Hello Who?", for: .normal)
        button.translatesAutoresizingMaskIntoConstraints = false
        button.addTarget(self, action: #selector(tap), for: .touchUpInside)

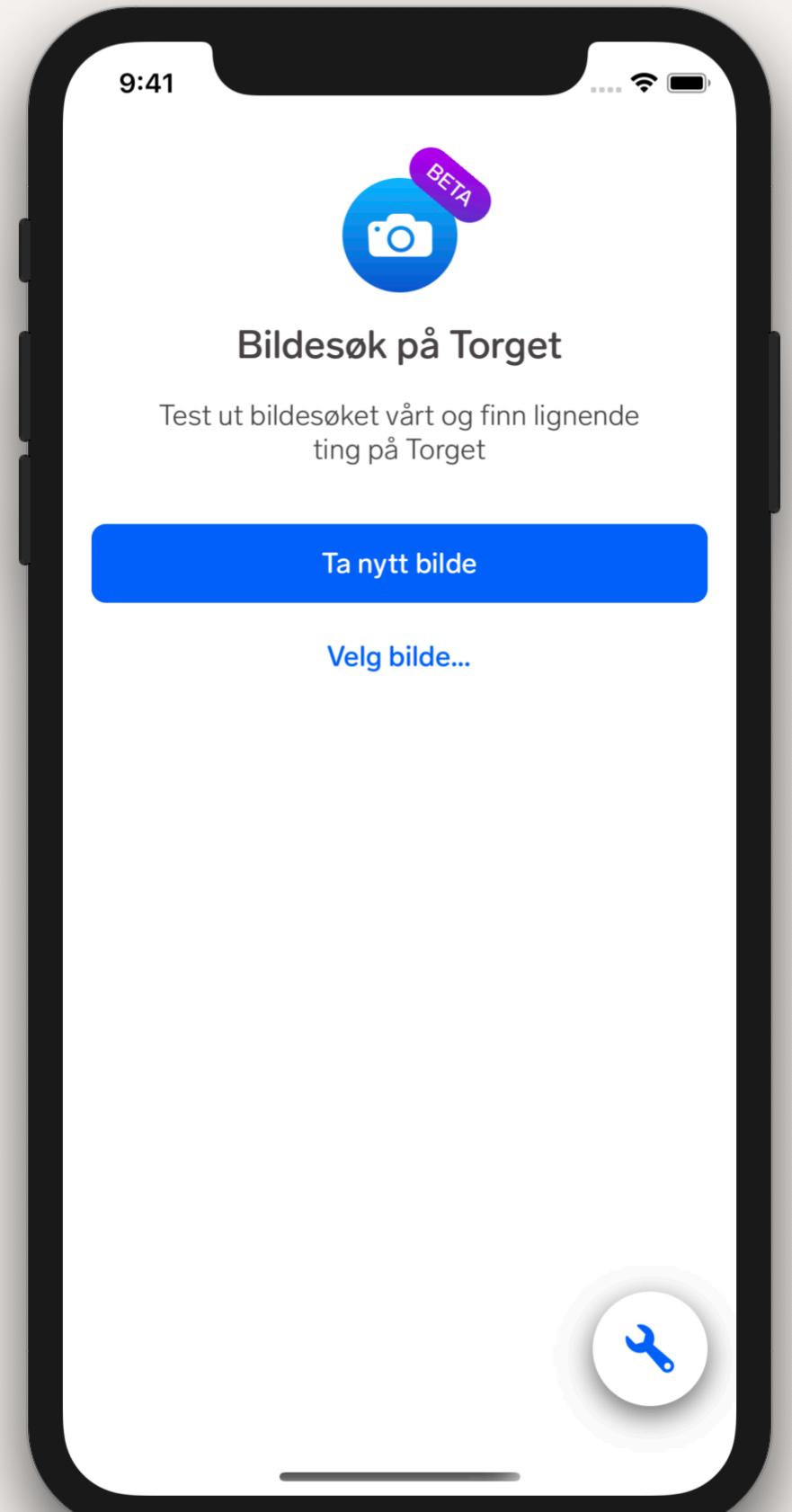
        self.view.addSubview(label)
        self.view.addSubview(button)

        NSLayoutConstraint.activate([
            label.topAnchor.constraint(equalTo: self.view.topAnchor, constant: 30),
            label.centerXAnchor.constraint(equalTo: self.view.centerXAnchor),

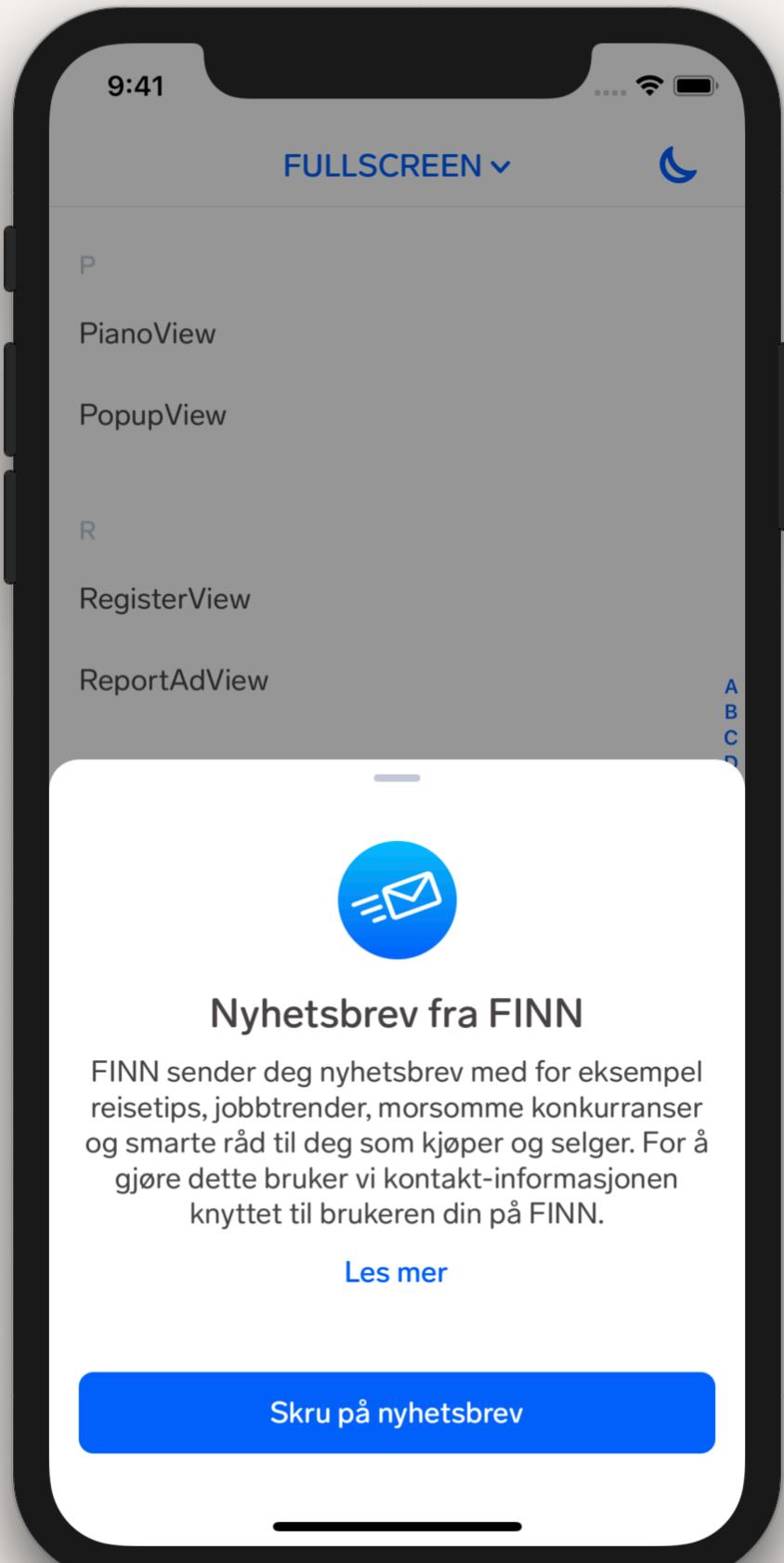
            button.topAnchor.constraint(equalTo: label.bottomAnchor, constant: 30),
            button.centerXAnchor.constraint(equalTo: self.view.centerXAnchor),
        ])
    }
}
```

FinniversKit

- FINN's UI component library
- it's open source and it's awesome :D
- [FinniversKit on Github](#)



iPhone 11 Pro — 13.3



iPhone 11 Pro — 13.3

BetaFeatureView

BetaFeatureViewModel

BetaFeatureViewDelegate

SettingDetailView

SettingDetailViewModel

SettingDetailViewDelegate

- similar layout
 - they differ in data requirements and order of elements
- delegates to bubble up touches on buttons
- tons of similar non-shareable boilerplate

Case 1: Object page & KeyValueGridView



The object page

bilia

| Drammen

Bilde 1 (1 / 29)



Volvo V40 Cross Country

Example: Object page & KeyValueGridView

Pris eks omreg 313 528 kr

Lån fra 3 927 kr
Eff.rente 4,93 %. 207 935 o/5 år. Kostnad: 27 685 kr. Totalt 235 620 kr.

Pris på forsikring

Reklamasjon Garanti Service Tilstand Bytterett Programbil

Bilen selges med 5 års reklamasjonsrett
Denne bilen selges av en bilforhandler og dette medfører at du som kjøper har 5 års reklamasjonsrett mot skjulte feil og mangler. Reklamasjonsretten kan brukes dersom bilen er i dårligere stand enn du som kjøper har grunn til å forvente, basert på bilens alder, kjørelengde og informasjon du har fått fra selger.
[Les mer om reklamasjonsrett på FINNs hjelpesenter.](#)

Kilometerstand 20 501 km

Farge Svart

Fargebeskrivelse Onyx black met. 717

Interiørfarge Sort

Modellår 2019

Girkasse Automat

Hjuldrift Forhjulsdrift

Forside Meldinger Mine annonser Meldinger Min FINN

Bilia Drammen Volvo

Send sms

Ring forhandler

Send melding

Chat med oss

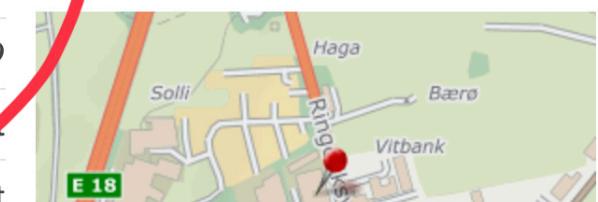
Flere annonser fra Bilia Drammen Volvo

Bruktilbiler hos Bilia

Garanti

[Les mer og kjøp bilen her](#)

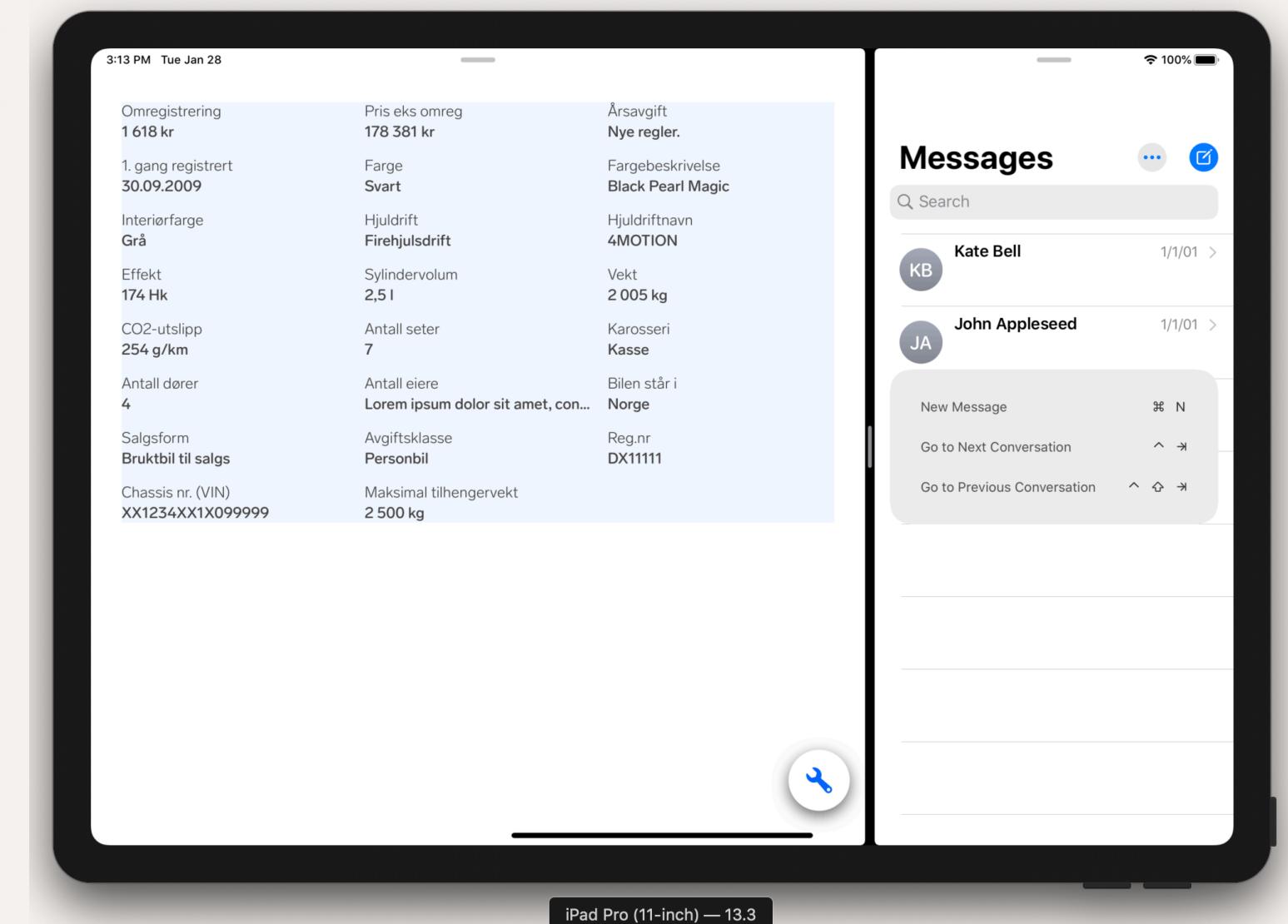
Industrigata 2, 3414 Lierstranda



old Table

Example: Object page & KeyValueGridView

Svart	Black Pearl Magic
Interiørfarge	Hjuldrift
Grå	Firehjulsdrift
Hjuldriftnavn	Effekt
4MOTION	174 Hk
Sylinderlengde	Vekt
2,5 l	2 005 kg
CO2-utslipp	Antall seter
254 g/km	7
Karosseri	Antall dører
Kasse	4
Antall eiere	Bilen står i
Lorem ipsum dolor si	Norge
Salgsform	Avgiftsklasse
Bruktil salgs	Personbil



UICollectionView

Understanding UICollectionViewLayout

To achieve the desired layout:

- Do i use flow layout?
- Do i subclass it? or the normal view layout?
- Do i use the delegate methods?

I need to make the cells's height to adjust to the content

Årsavgift	1. gang registrert
Nye regler.	30.09.2009
Farge	Fargebeskrivelse
Svart	Black Pearl Magic
Interiørfarge	Hjuldrift
Grå	Firehjulsdrift
Hjuldriftnavn	Effekt
4MOTION	174 Hk
Sylindervolum	Vekt
2,5 l	2 005 kg
CO2-utslipp	Antall seter
254 g/km	7
Karosseri	Antall dører
Kasse	4
Antall eiere	Bilen står i
Lorem ipsum dolor si	Norge
Salgsform	Avgiftsklasse
Bruktil bil til salgs	Personbil
Reg.nr	Chassis nr. (VIN)
DX11111	XX1234XX1X09999
Maksimal tilhengerv ekt	

Understanding UICollectionViewLayout

```
override func prepare() {
    super.prepare()

    itemAttributes = [UICollectionViewLayoutAttributes]()

    guard let collectionView = collectionView else {
        return
    }

    let columnsRange = 0 ...< configuration.numberOfColumns

    var columns = columnsRange.map { _ in 0 }
    var attributesCollection = [UICollectionViewLayoutAttributes]()
    var yOffset = configuration.topOffset

    if let height = delegate.collectionViewLayout(self, heightForHeaderInCollectionView: collectionView) {
        let attributes = UICollectionViewLayoutAttributes(forSupplementaryViewOfKind: UICollectionView.elementKindSectionHeader, with: IndexPath(item: 0, section: 0))
        attributes.frame = CGRect(x: 0, y: 0, width: collectionView.frame.size.width, height: height)
        attributesCollection.append(attributes)

        yOffset += height
    }

    for index in 0 ...< numberOfItems {
        let columnIndex = indexOfLowestValue(in: columns)

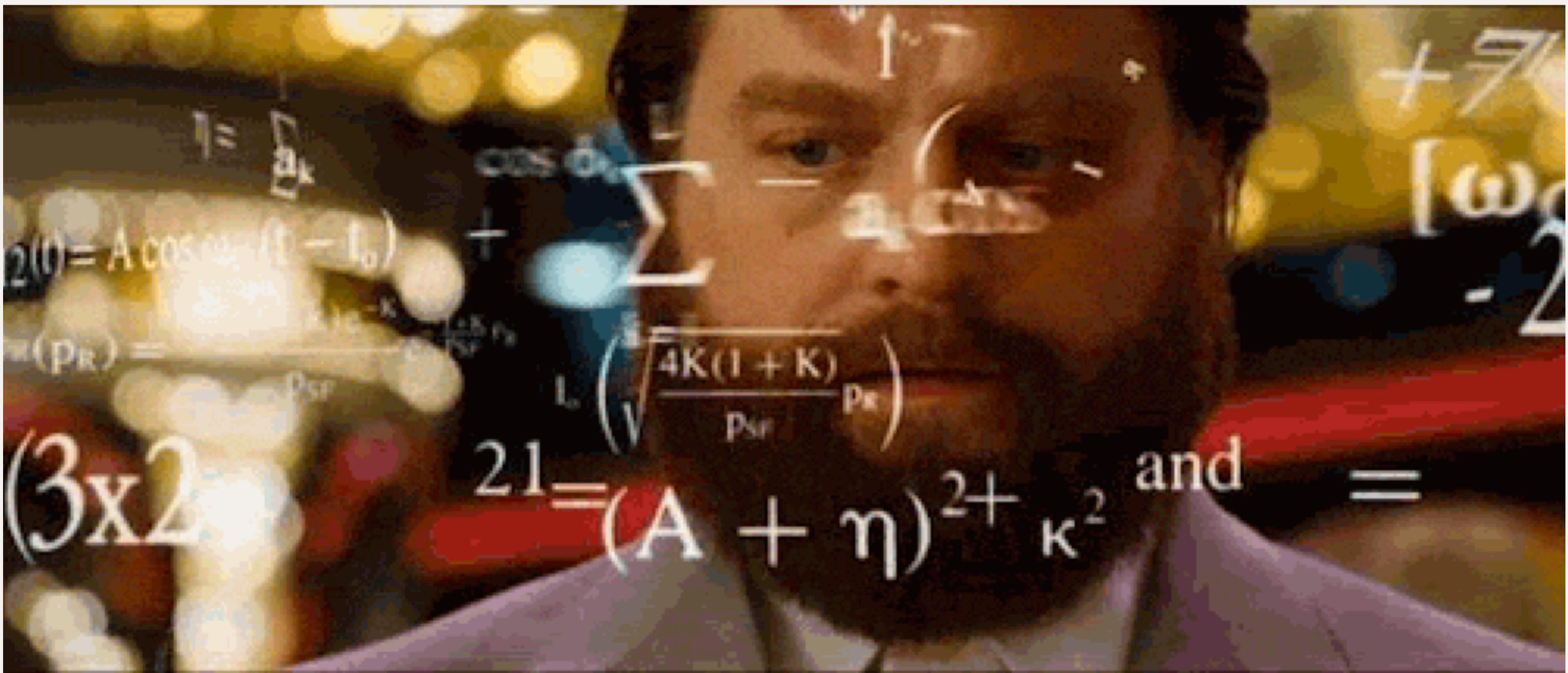
        let xOffset = xOffsetForItemInColumn(itemWidth: itemWidth, columnIndex: columnIndex)
        let topPadding = configuration.numberOfColumns > index ? yOffset : 0.0
        let verticalOffset = CGFloat(columns[columnIndex]) + topPadding

        let indexPath = IndexPath(item: index, section: 0)
        let itemHeight = delegate.collectionViewLayout(self, heightForItemWithWidth: itemWidth, at: indexPath)

        columns[columnIndex] = Int(verticalOffset + itemHeight + configuration.columnSpacing)

        let attributes = UICollectionViewLayoutAttributes(forCellWith: indexPath)
        attributes.frame = CGRect(x: xOffset, y: verticalOffset, width: itemWidth, height: itemHeight)
        attributesCollection.append(attributes)
    }

    itemAttributes.append(contentsOf: attributesCollection)
}
```



It shows clearly the "How", not the "What"

First attempt: UICollectionView

Motor MP: new table component #759

Merged fespinoza merged 1 commit into master from feature/table-data 12 days ago

Conversation 1 Commits 1 Checks 0 Files changed 11 +492 -1

fespinoza commented 13 days ago • edited

Why?

For the motor MP we have a new way to render the specification list

The screenshot shows a GitHub pull request titled "Motor MP: new table component #759". The status bar indicates it is merged into the master branch from the feature/table-data branch 12 days ago. Below the title, there are tabs for Conversation (1), Commits (1), Checks (0), and Files changed (11). A summary of code changes shows +492 additions and -1 deletion, with a green progress bar. The main content area displays a comment from user fespinoza asking "Why?", followed by a response explaining the new rendering method. On the right side, there are sections for Reviewers (bstien) and Assignees (bstien), each with a gear icon. Below the assignee section is a file tree for the "KeyValueCollectionView" directory:

- KeyValueCollectionView
 - ColumnBasedFlowLayout.swift
 - KeyValueCollectionView.swift (highlighted in blue)
 - KeyValuePair.swift
- Internal
 - KeyValueCollectionViewCell.swift
 - KeyValueCollectionDataSource.swift
 - KeyValueCollectionLayoutFactory.swift
- Klimabrolet

But then...

- The object page was a collection view
- The cells need to calculate their height for the given width
- My new component is supposed to be inside one of these cells
- The height of the new table component was calculated asynchronously

Challenges when using UICollectionView

- self-sizing content
- asynchronous calculation of height

collectionView(collectionView) = complexity²

Challenges when using UICollectionView

- it's hacking time!



Or...

Second Attempt: UIStackView

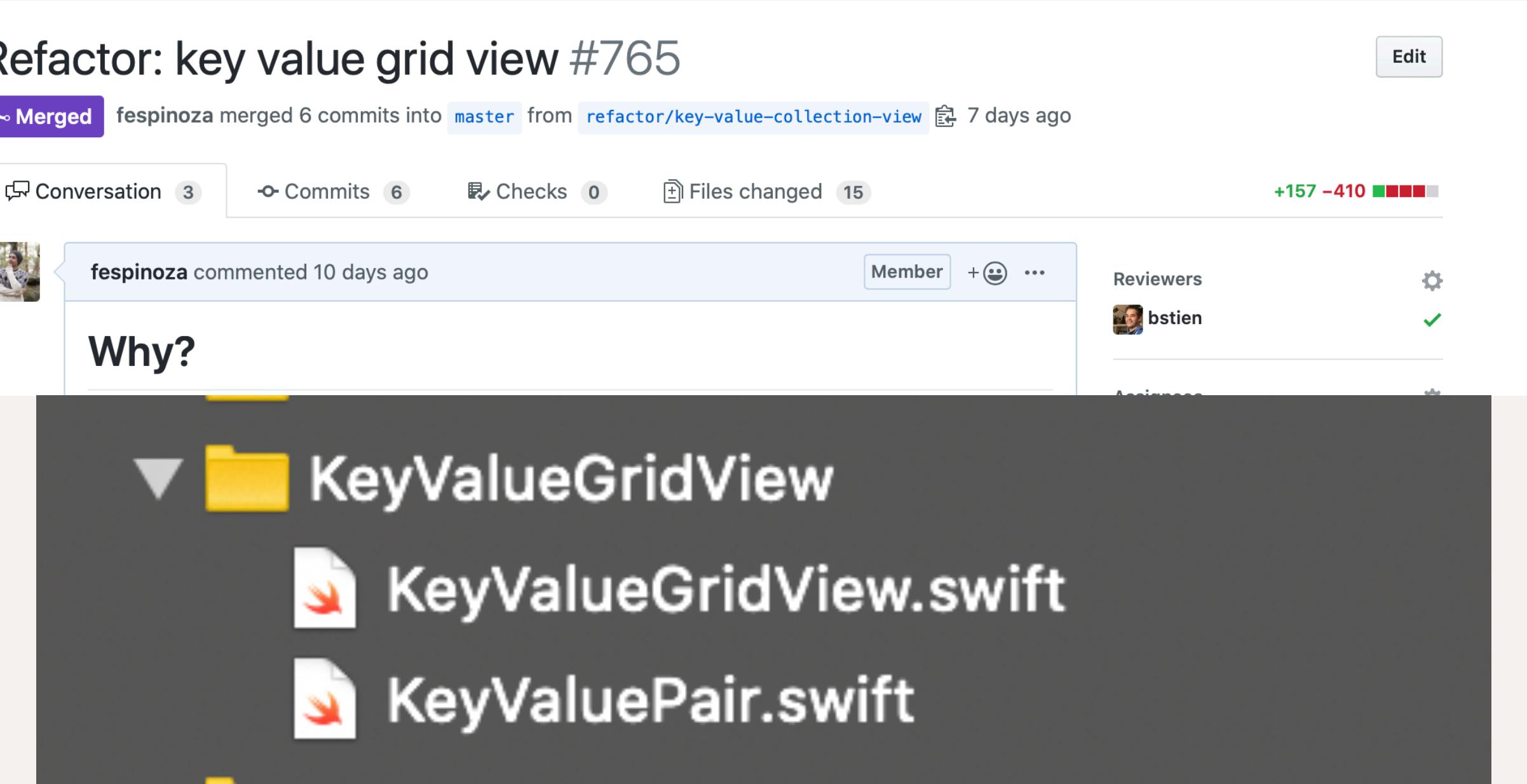
Refactor: key value grid view #765 Edit

Merged fespinoza merged 6 commits into `master` from `refactor/key-value-collection-view` 7 days ago

Conversation 3 Commits 6 Checks 0 Files changed 15 +157 -410

fespinoza commented 10 days ago Member + ...

Why?



Second Attempt: UIStackView

- The height of the component itself will be available synchronously when passing the data.
- The integration to the object page doesn't require any hacks and it's easier.

When not to use UICollectionView

- No cell reuse
- No dynamic data

Case 2: SwiftUI

20:53



◀ Search

🔍 Søk i hele FINN



Eiendom



Bil



Torget



Jobb



MC



Båt



Nyttekjøretøy



Økonomi



Reise



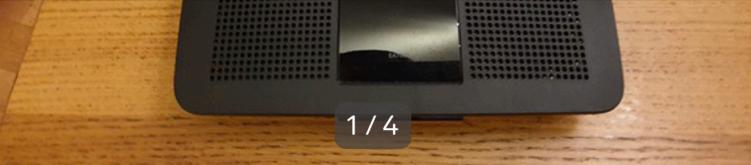
Leiebil



Oppdrag



Møteplassen

 1 / 4

Solgt

Linksys EA7500 trådløs router AC1900

500 kr

[Send melding](#)

Svarer vanligvis innen ti minutter

Mobil [Vis telefonnummer](#)

 [Jan A. Aasland Linn](#) 
Har vært på FINN siden 2006

 El [INNEIEN 96, 1253 Oslo](#)

 [Sjekk reiserute med ENTUR](#) 

Nyere Linksys EA7500 AC1900 wifi router selges.
Nypris er 1100 kr. Overføringshastigheter på opptil
1,9 Gbps med 802.11ac teknologi. Det er 2 USB
porter og 4 ethernet porter. Selges grunnet at iea

```
import SwiftUI

struct AdView: View {
    let adViewModel: AdViewModel
    let imageDownloader: CollectionImageDownloader

    var body: some View {
        ScrollView {
            VStack(alignment: .leading, spacing: .spacingM) {
                imageGalleryView

                Group {
                    titleView
                    priceView
                    sendMessageButton
                    phoneNumber
                    profileView
                    addressView
                    descriptionView
                    conditionView
                }.padding(.horizontal)

                similarAdsView
            }
        }
    }
}
```



Linksys EA7500 trådløs router AC1900

500 kr

Send melding

Svarer vanligvis innen ti minutter

Mobil

Vis telefonnummer



Homer Simpson

Har vært på FINN siden 2006

Evergreen Terrace 742, 1234 Oslo

Nyere Linksys EA7500 AC1900 wifi router selges. Nypris er 1100 kr.

Overføringshastigheter på opptil 1,9 Gbps med 802.11ac teknologi. Det er 2 USB porter og 4 ethernet porter. Selges grunnet at jeg

```
extension AdView {  
    var titleView: some View {  
        Text(adViewModel.title).title()  
    }  
  
    var priceView: some View {  
        Text(adViewModel.price).titleStrong()  
    }  
  
    var sendMessageButton: some View {  
        VStack(alignment: .center, spacing: .spacingS) {  
            FINCTAButton {  
                Text("Send melding")  
            }  
            Text(adViewModel.onwerAverageResponseTime).caption()  
        }  
    }  
  
    var imageGalleryView: some View {  
        GalleryView(imageDownloader: imageDownloader)  
    }  
}
```



Linksys EA7500 trådløs router AC1900

500 kr

Send melding

Svarer vanligvis innen ti minutter

Mobil

[Vis telefonnummer](#)



Homer Simpson

Har vært på FINN siden 2006

[Evergreen Terrace 742, 1234 Oslo](#)

Nyere Linksys EA7500 AC1900 wifi router
selges. Nypris er 1100 kr.

Overføringshastigheter på opptil 1,9 Gbps
med 802.11ac teknologi. Det er 2 USB porter
og 4 ethernet porter. Selges grunnet at jeg

8:56 PM Tue Feb 11

100%



1 / 4

**Linksys EA7500 trådløs router
AC1900**

500 kr

LUSETJERNVEIEN 96, 1253 Oslo

Nyere Linksys EA7500 AC1900 wifi router selges. Nypris er 1100 kr. Overføringshastigheter på opptil 1,9 Gbps med 802.11ac teknologi. Det er 2 USB porter og 4 ethernet porter. Selges grunnet at jeg fikk ny trådløs router gjennom Get. Selges med original emballasje. Anmeldelse av router kan leses her: <https://www.dinside.no/data/linksys-ea7500-med-mu-mimo-teknologi/60966544>

Tilstand Brukt

[Elektronikk og hvitevarer > Data > Datatilbehør >](#)

[Pris på lån](#)

[Rapporter svindel/regelbrudd](#)

FINN-kode 163137299

Sist endret 20. jan. 2020, 13:36

Lignende annonser



Diverse Philips Hue utstyr
selges (E27 pærer, brytere)

Oslo



Google Nest Hello

Oslo



Philips hue startsett med 2
ekstra GU10 spotter og 1 stk
E27 pære

Oslo

Send melding

Har vært på FINN siden 2006

Mobil

Vis telefonnummer

Svarer vanligvis innen ti minutter

Conditional Layout

```
struct AdView: View {
    let adViewModel: AdViewModel
    var imageDownloader: CollectionImageDownloader {
        CollectionImageDownloader(adViewModel: adViewModel)
    }
}

@Environment(\EnvironmentValues.horizontalSizeClass) var horizontalSizeClass

var body: some View {
    ScrollView {
        if horizontalSizeClass == UserInterfaceSizeClass.compact {
            compactLayout
        } else {
            regularLayout
        }
    }
}
```

Conditional Layout

```
struct AdView: View {
    let adViewModel: AdViewModel
    var imageDownloader: CollectionImageDownloader {
        CollectionImageDownloader(adViewModel: adViewModel)
    }
}

@Environment(\EnvironmentValues.horizontalSizeClass) var horizontalSizeClass

var body: some View {
    ScrollView {
        if horizontalSizeClass == UserInterfaceSizeClass.compact {
            compactLayout
        } else {
            regularLayout
        }
    }
}
```

Conditional Layout

```
struct AdView: View {
    let adViewModel: AdViewModel
    var imageDownloader: CollectionImageDownloader {
        CollectionImageDownloader(adViewModel: adViewModel)
    }
}

@Environment(\EnvironmentValues.horizontalSizeClass) var horizontalSizeClass

var body: some View {
    ScrollView {
        if horizontalSizeClass == UserInterfaceSizeClass.compact {
            compactLayout
        } else {
            regularLayout
        }
    }
}
```

Conditional Layout

```
extension AdView {  
    var compactLayout: some View {  
        VStack(alignment: .leading, spacing: .spacingM) {  
            imageGalleryView  
  
            Group {  
                titleView  
                priceView  
                sendMessageButton  
                phoneNumber  
                // same as before..  
            }.padding(.horizontal)  
  
            similarAdsView  
        }  
    }  
}
```

Conditional Layout

```
extension AdView {  
    var regularLayout: some View {  
        VStack(alignment: .leading, spacing: .spacingM) {  
            imageGalleryView  
  
            HStack(alignment: .top, spacing: .spacingM) {  
                VStack(alignment: .leading, spacing: .spacingM) {  
                    Group {  
                        titleView  
                        priceView  
                        addressView  
                        descriptionView  
                        // ...  
                    }.padding(.horizontal)  
                }  
  
                VStack(alignment: .leading, spacing: .spacingM) {  
                    profileView  
                    sendMessageButton  
                    phoneNumber  
                }  
            }  
            similarAdsView  
        }  
    }  
}
```



Recap

Recap (or just stating the obvious)

- use the right tool for the job (*I'm looking at you UICollectionView*)
- SwiftUI declarative syntax makes creating small/reusable components way better than UIKit
- multiple simple components, over single "universal" component
- 🤝 together with top level decisions, make implementing different layouts/scenarios easy.

**Views: complexity
and reusability**

**Experiments comparing SwiftUI
and UIKit**

Bonus

Bonus

- Attempt to use iOS 13 UICollectionView features while supporting iOS 11+: <https://github.com/finn-no/FinniversKit/pull/759>