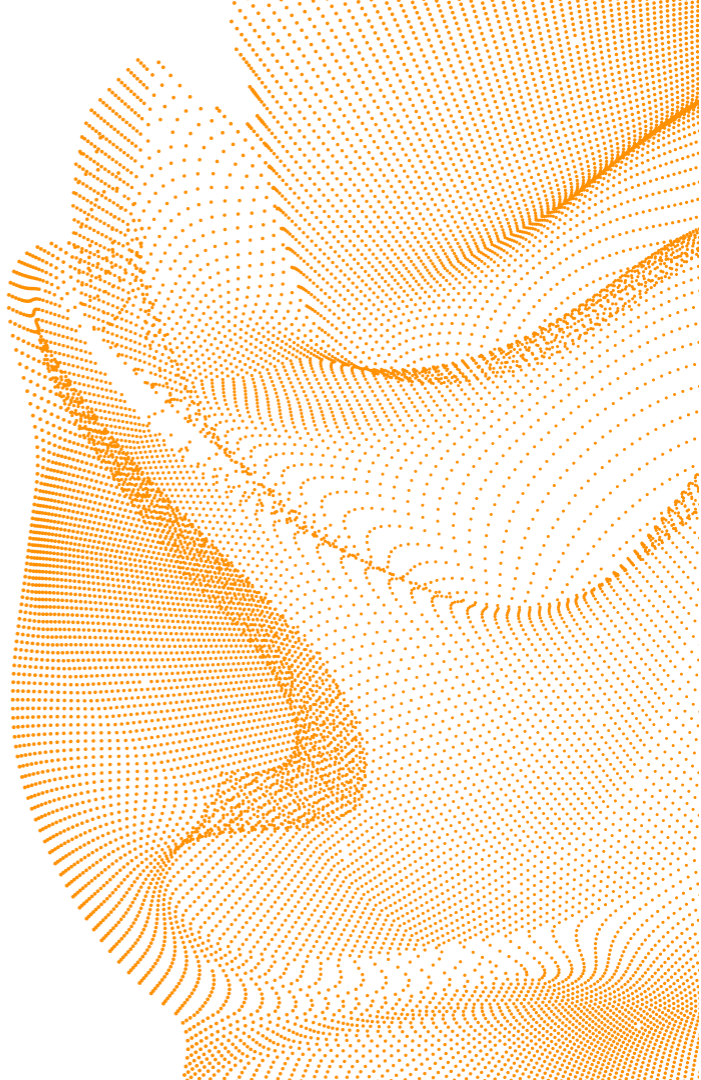
ОРЕНБУРГ | 12 — 13 СЕНТЯБРЯ | 2025

ПРОФЕССИОНАЛЬНАЯ МЕЖРЕГИОНАЛЬНАЯ

IT-КОНФЕРЕНЦИЯ MERGE



merge

ОРЕНБУРГ | 12 — 13 СЕНТЯБРЯ | 2025

1С-Коннект
Программист iOS

Долгих Сергей

Custom UICollectionViewLayout

МЕЛКИЕ УЛУЧШЕНИЯ ЧАТА... ПОЛГОДА БОЛИ

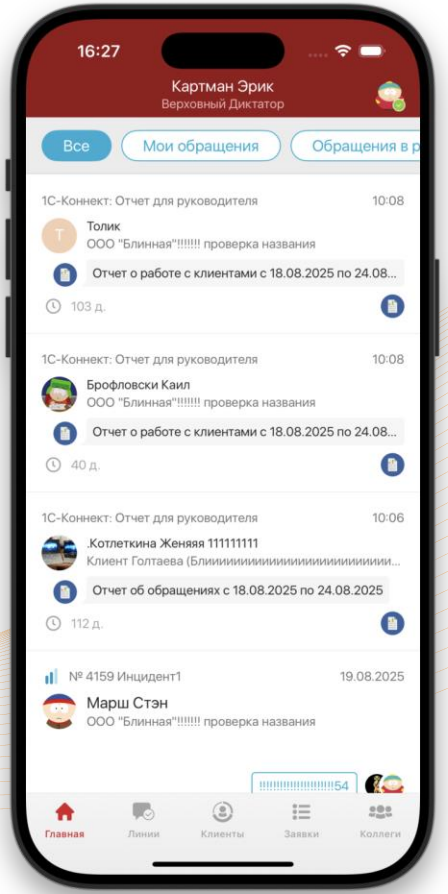
«На первые 90% кода уходят первые 90% времени. На оставшиеся 10% — остальные 90% времени разработки»
Том Каргилл, Bell Labs, 1985 г.

merge

1С-Коннект
Программист iOS

Почему чат – сложная штука?

1С-Коннект
Программист iOS



1С-Коннект
Программист iOS

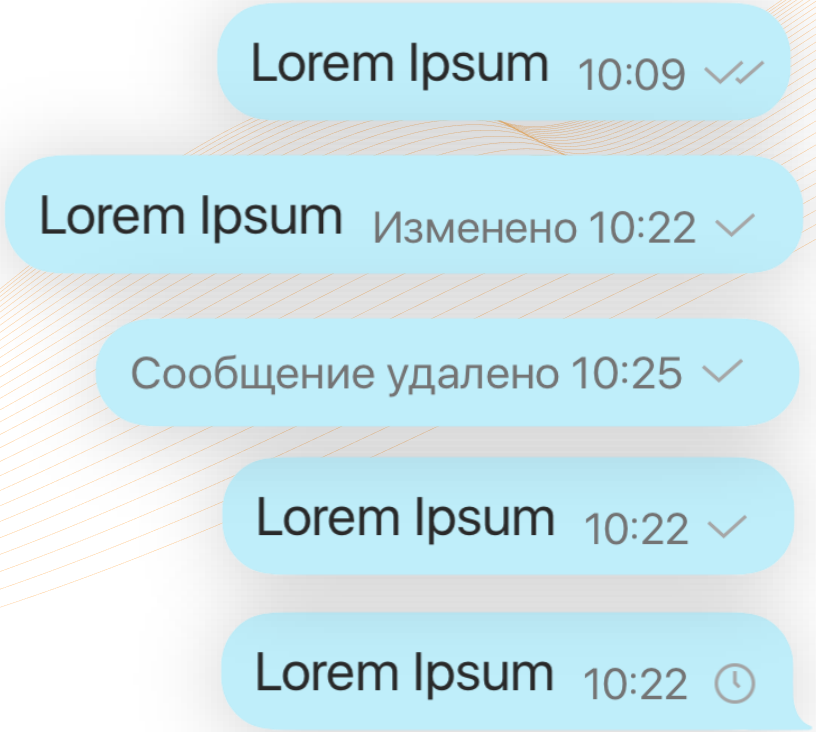
1С-Коннект
Программист iOS

1. Огромное количество состояний и сценариев

1С-Коннект
Программист iOS

1. Огромное количество состояний и сценариев

- **Статусы сообщений**



1С-Коннект
Программист iOS

1. Огромное количество состояний и сценариев

- **Статусы сообщений**
- **Типы сообщений**

Сломано в этом релизе? 17:25

2025-06-06 10-06-48 - Copy.log
Размер 21,1 МБ 15:51

Голтаева Катерина
Сломано в этом релизе?
Думаю да... 14:43

Звонок
Начало разговора 17:44

Звонок завершен
2 мин 17:47

Обращение завершено автоматически из-за отсутствия активности
16:00

Голтаева Катерина
№ 2339 Консультация
Москвин Ян
1С-Коннект
Консультация по релизу с тикетами
25.06.2025 16:21 Новая
17:14

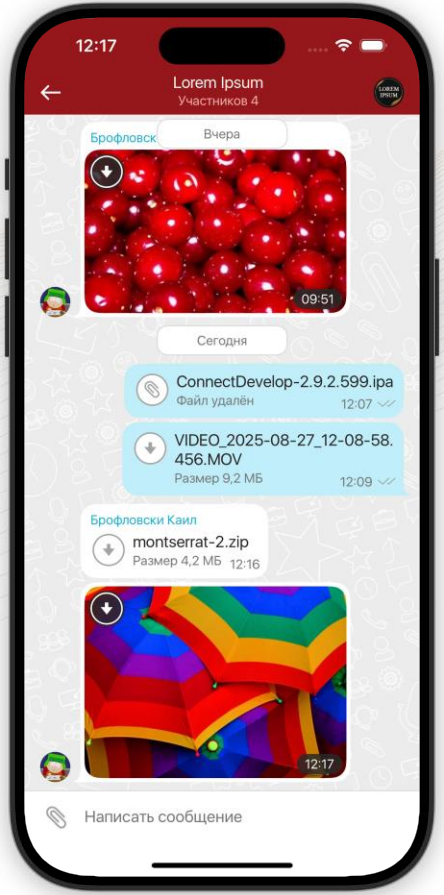
XII корпоративная конференция 1С-...
Футболки для корпоративной Конференции 2025
Привет, Сергей!
Пройди пожалуйста опрос до конца сегодняшнего дня!)
Пройти опрос
11:15

НОВЫЕ СООБЩЕНИЯ

1С-Коннект
Программист iOS

1. Огромное количество состояний и сценариев

- **Статусы сообщений**
- **Типы сообщений**
- **Состояния загрузки**



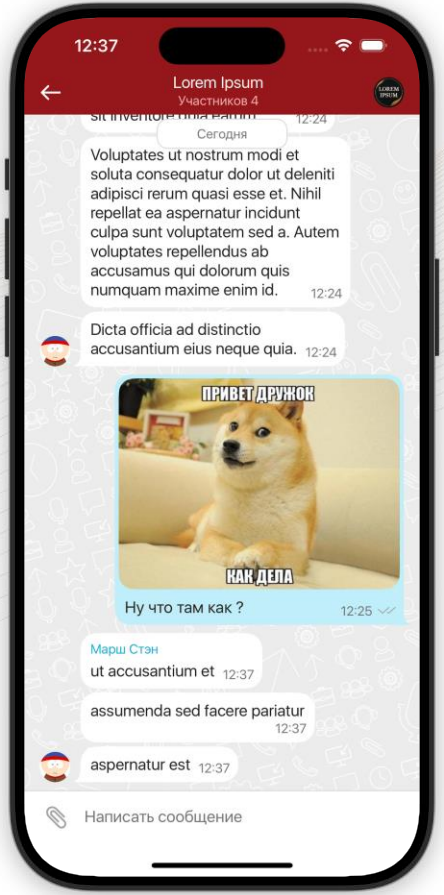
1. Огромное количество состояний и сценариев

- **Статусы сообщений**
- **Типы сообщений**
- **Состояния загрузки**
- **Ввод сообщения**

1С-Коннект
Программист iOS

2. Работа с разными размерами контента

- **Динамическая высота сообщений**
- **"Пузыри" сообщений**



3. Проблемы производительности

- **Рендеринг большого количества сообщений**
- **Медиафайлы**
- **Бесконечная лента**

1С-Коннект
Программист iOS

4. Навигация и жесты

- **Клавиатура**

4. Навигация и жесты

- **Клавиатура**
- **Контекстное меню**

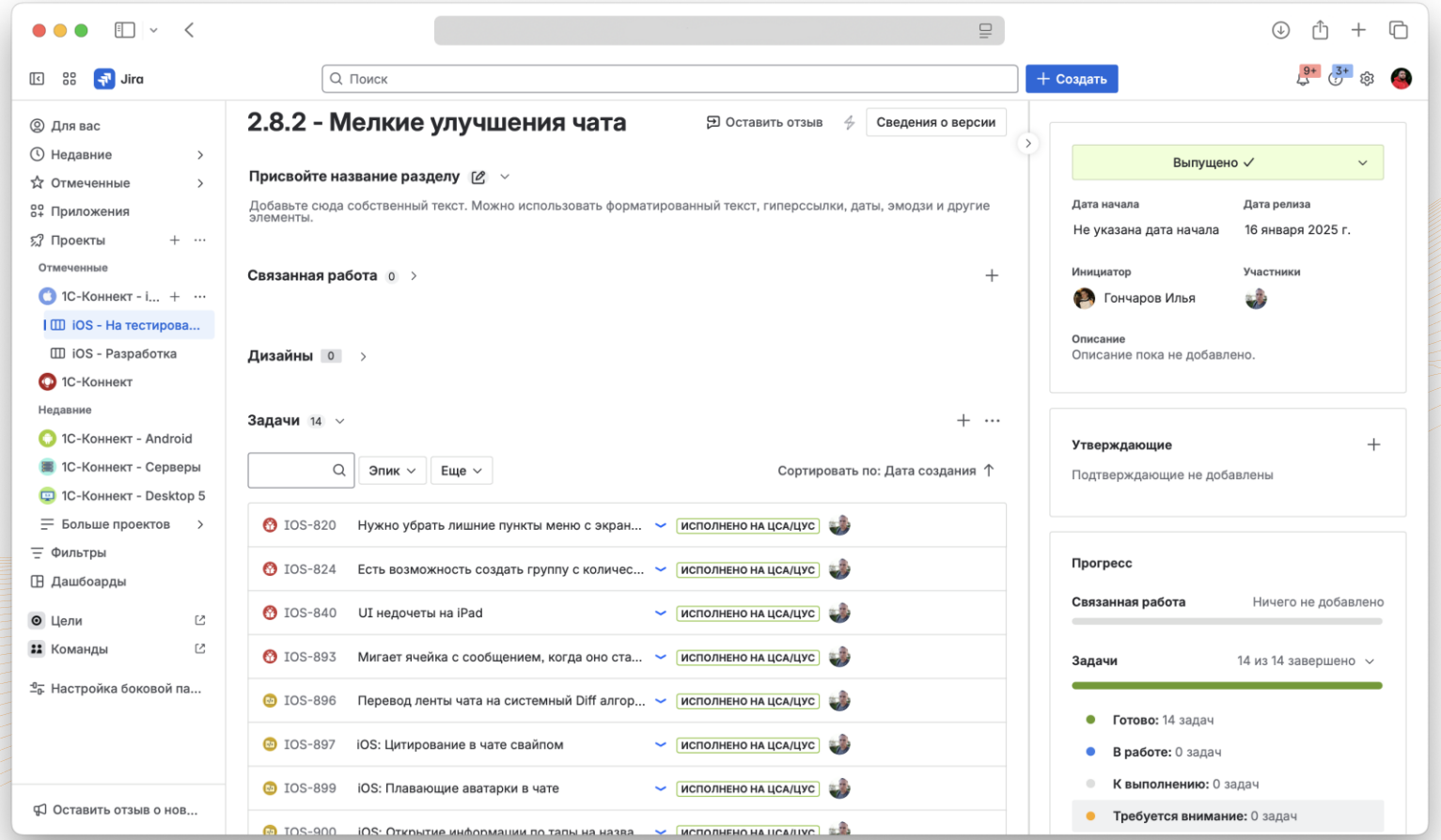
1С-Коннект
Программист iOS

4. Навигация и жесты

- **Клавиатура**
- **Контекстное меню**
- **Свайпы**

1С-Коннект
Программист iOS

1С-Коннект
Программист iOS

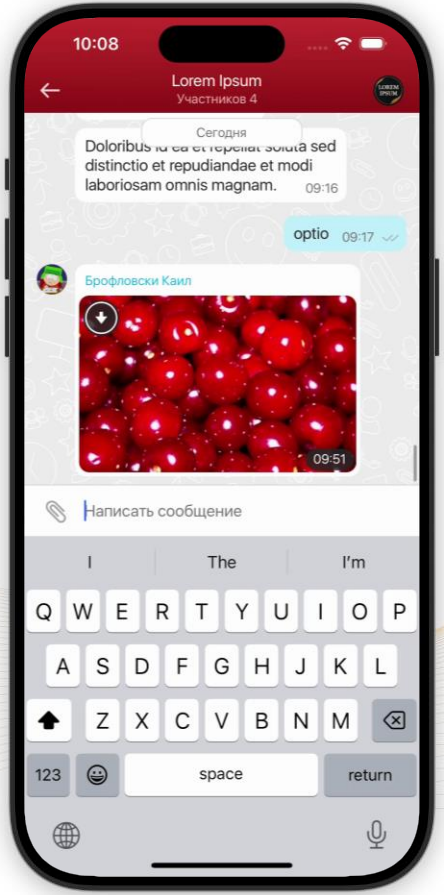


1С-Коннект
Программист iOS

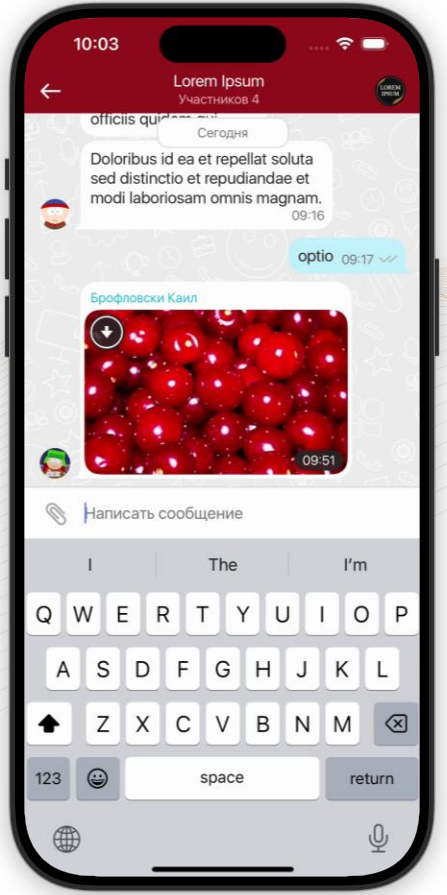
Почему мелкие улучшения?

1С-Коннект
Программист iOS

Было

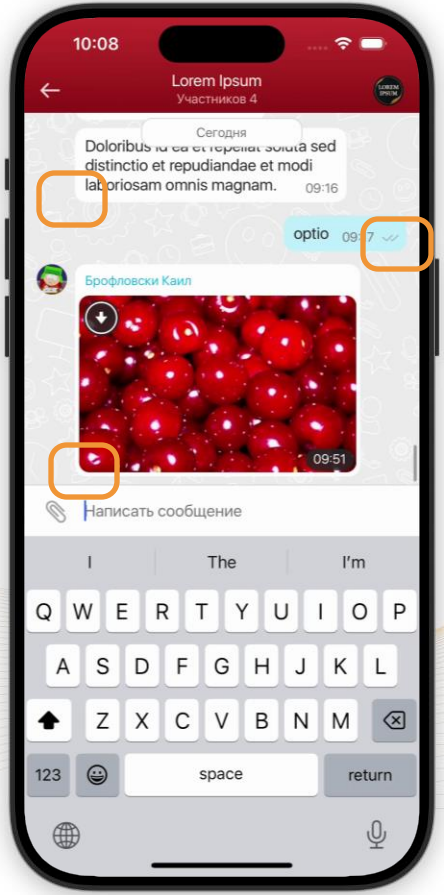


Стало

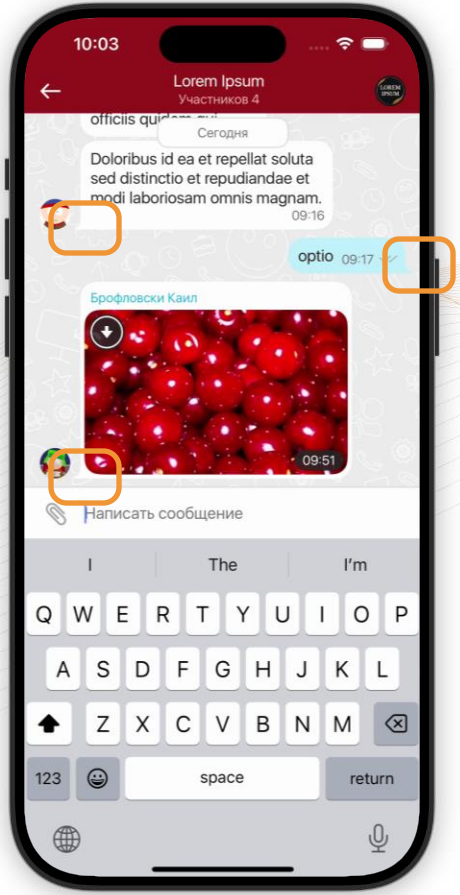


1С-Коннект
Программист iOS

Было

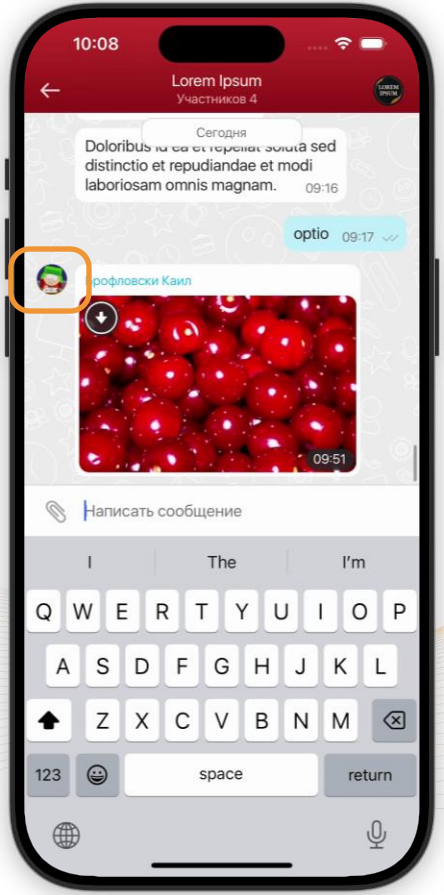


Стало

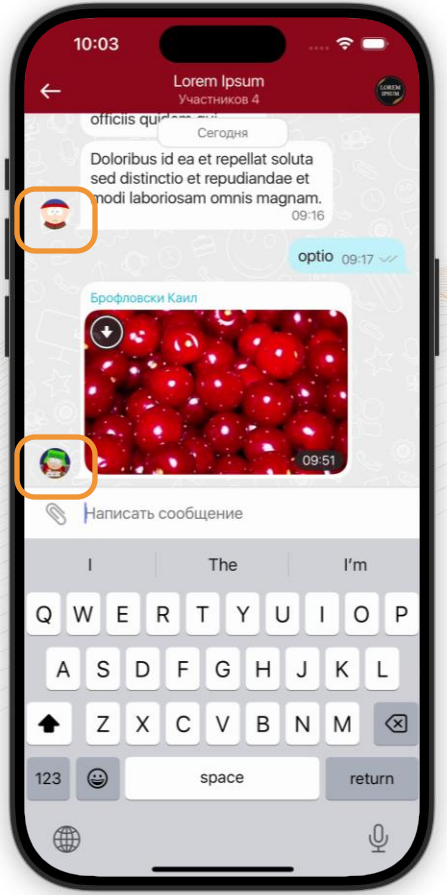


1С-Коннект
Программист iOS

Было

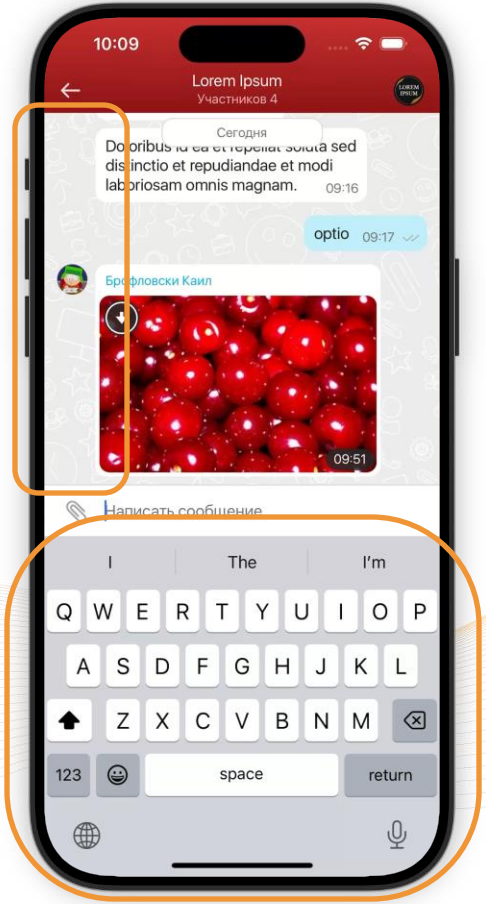


Стало

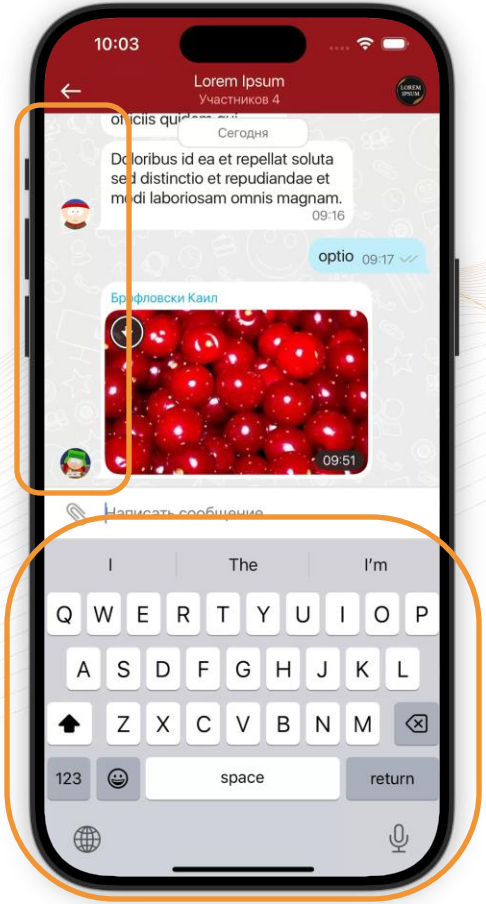


1С-Коннект
Программист iOS

Было



Стало



1С-Коннект
Программист iOS

Что изменилось в коде чата?

1С-Коннект
Программист iOS

Всё

1С-Коннект
Программист iOS

iOS Devs / IOS App / Merge requests / !516

Мелкие улучшения чата

Merged Dolgih Sergey requested to merge feature/IOS-896-new-chat-d… into release/2.8.2 9 months ago

Edit Code

Overview 0 Commits 130 Pipelines 8 Changes 398 Add a to-do item

Compare release/2.8.2 and latest version 398 files +24515 −19990

Project
IOS App
Pinned
Issues 145
Merge requests 3
Manage
Plan
Code
Merge requests 3
Repository
Branches
Commits
Tags
Repository graph
Compare revisions
Build
Deploy
Analyze
Help

Files 398

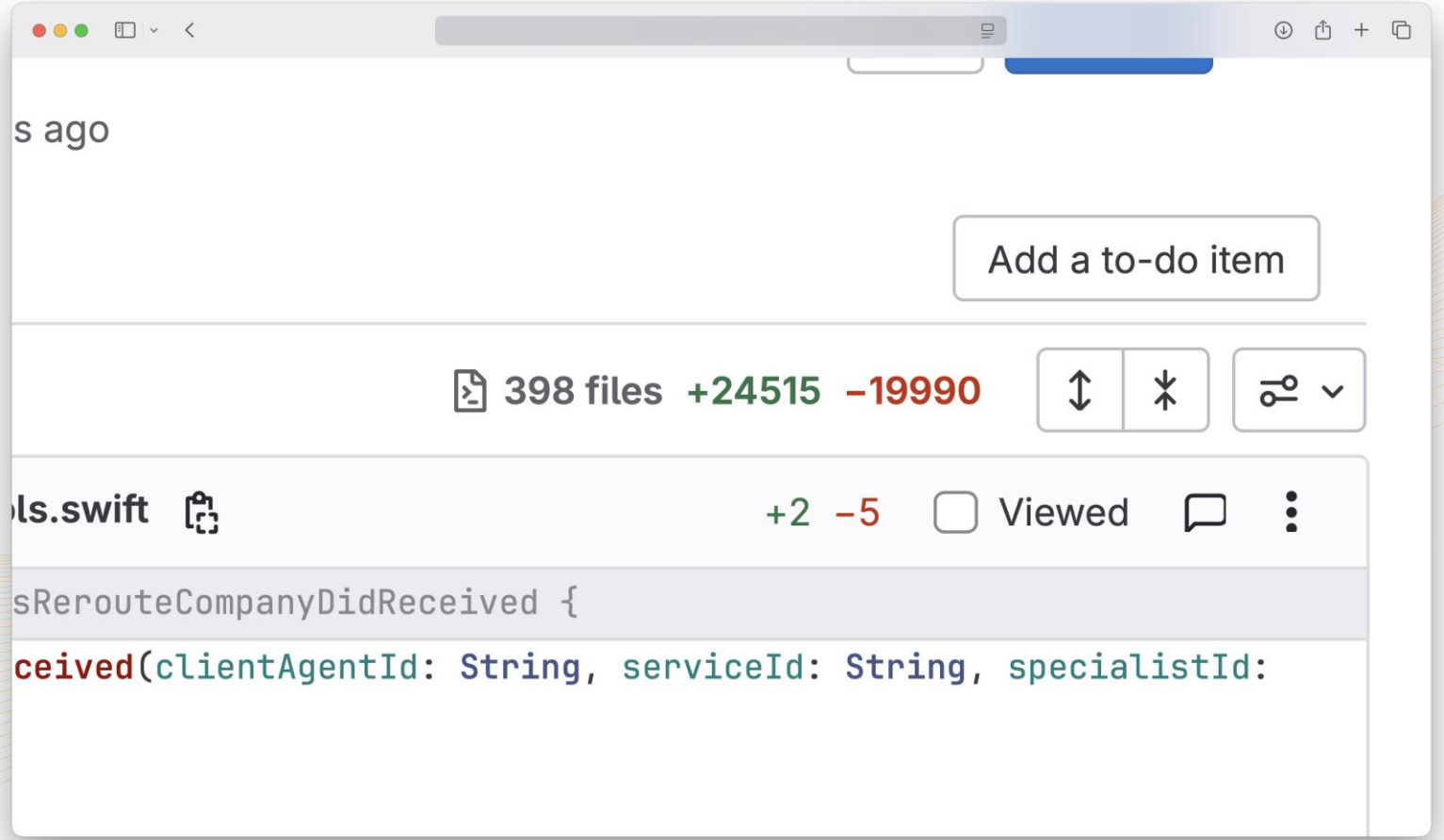
Search (e.g. *.vue) (⌘P)

Connect
Common/Observable
ObservableProtocols.swift +2 -5
Entities
ApiResponses/APIv1/Common
Quality + init.swift +23 -0
Common
Agent
AgentModel + init.swift +25 -20
AgentModel.swift +8 -0
ClientUser
ClientUserModel.swift +44 -0
ConferenceModel
Conference...rModel.swift +22 -0
ConferenceModel.swift +56 -0
Counterparty
Counterpart... + init.swift +34 -0

Connect/Common/Observable/ObservableProtocols.swift +2 −5 Viewed

```
@@ -214,10 +214,6 @@ protocol XmppNssRerouteCompanyDidReceived {
    func xmppNssRerouteCompanyDidReceived(clientAgentId: String, serviceId: String, specialistId: String?, newCompanyId: String)
}

- protocol XmppNssQualityDidReceived {
-     func xmppNssQualityDidReceived(clientAgentId: String, serviceId: String, messageId: String, qualityValue: Int)
- }
-
// MARK: Message
protocol XmppP2PMessageDidReceive {
    func xmppP2PMessageDidReceive(message: PersonToPersonMessageModel)
@@ -397,10 +393,11 @@ protocol FileTransferDidUpdated {
    func fileTransferDidUpdated(chatId: String, fileTransferModel: ChatFileTransferModel)
}

+ @available(*, deprecated, message: "Нужно удалить так как не используется")
protocol FileTransferDidRemoved {
    func fileTransferDidRemoved(chatId: String)
}
-
+ @available(*, deprecated, message: "Нужно удалить так как не используется")
protocol FileUploadingDidStarted {
    func fileUploadingDidStarted(chatId: String, fileId: String)
}
```

1С-Коннект
Программист iOS



1С-Коннект
Программист iOS

ageres7 — -zsh — 84×24

```
ageres7@MacBook-Air ~ % cloc /Users/ageres7/connect-app
    2103 text files.
    1775 unique files.
     843 files ignored.

github.com/AlDanial/cloc v 2.04  T=1.54 s (1149.6 files/s, 131585.6 lines/s)
-------------------------------------------------------------------------------
Language                     files          blank        comment           code
-------------------------------------------------------------------------------
Swift                         1264          26874          12522         134435
XML                            121            321             20          10556
JSON                           288              0              0           9005
Markdown                         5            771              0           3494
C/C++ Header                    89            970           1798           1909
Objective-C                      3             39             23            154
Bourne Shell                     1             24             24            141
YAML                             1              3             12             33
Python                           1              8              0             24
SVG                              1              0              0              3
Text                             1              0              0              1
-------------------------------------------------------------------------------
SUM:                          1775          29010          14399         159755
-------------------------------------------------------------------------------
ageres7@MacBook-Air ~ %
```

1С-Коннект
Программист iOS

UIKit

1С-Коннект
Программист iOS

UICollectionView

1С-Коннект
Программист iOS

UICollectionViewCell

Officiis minima ipsa quae optio cum repellat autem voluptatem. 13:27

```
func configure(from model: ChatItem, maxSize: CGSize) { ••• }
```

UICollectionViewCell

Марш Стэн

Accusantium ipsam sunt aut quia qui rem et magni enim provident reprehenderit quia ea doloremque. 13:28

1С-Коннект
Программист iOS

UICollectionViewCell

Officiis minima ipsa quae optio cum repellat autem voluptatem. 13:27

```
func configure(from model: ChatItem, maxSize: CGSize) { ••• }
```

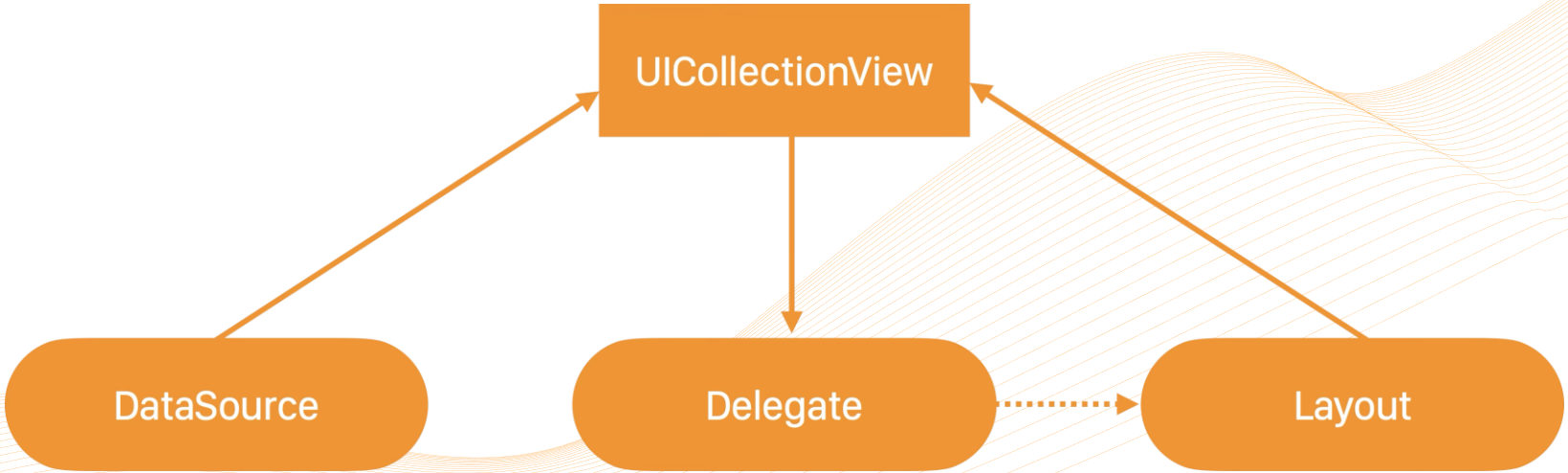
UICollectionViewCell

Марш Стэн

Accusantium ipsam sunt aut quia qui rem et magni enim provident reprehenderit quia ea doloremque. 13:28

1С-Коннект
Программист iOS

Отображает ячейки.
Отвечает за переиспользование.
Главный элемент.



Количество секций, ячеек.
Вью для ячеек и
дополнительных элементов.

Обрабатывает действия пользователя
Предоставляет информацию для Layout

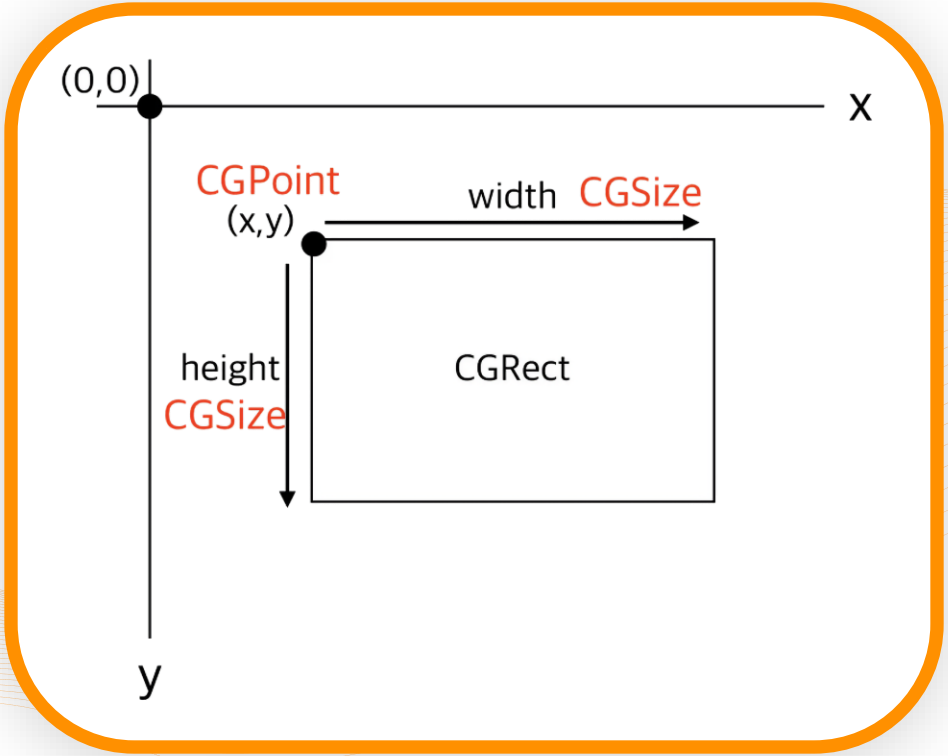
Информация о
фреймах ячеек

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

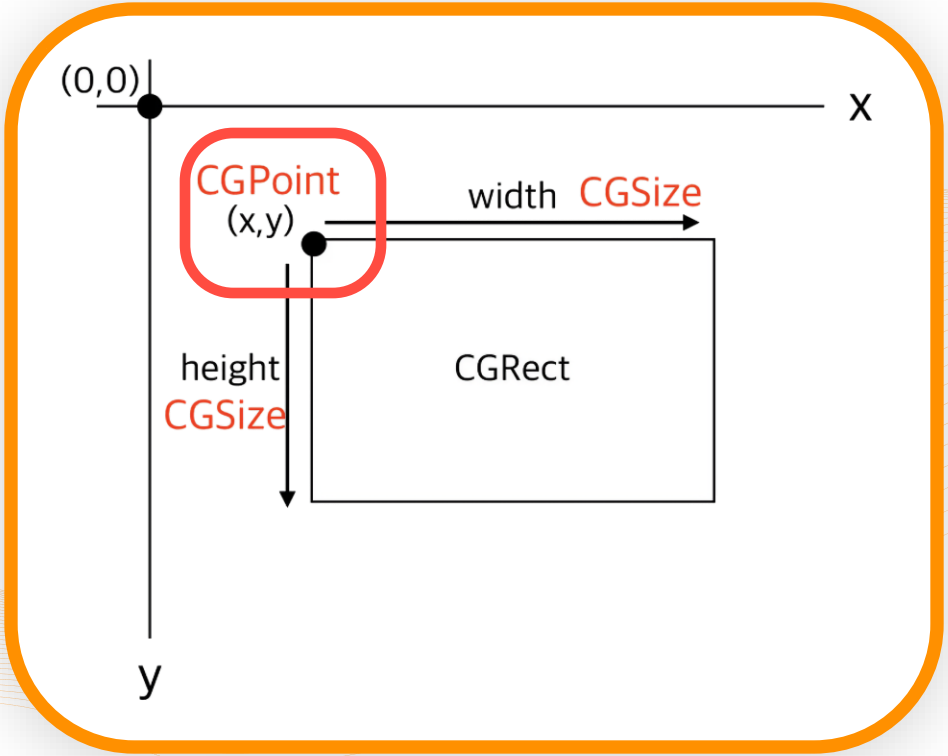
UICollectionViewCell

1С-Коннект
Программист iOS



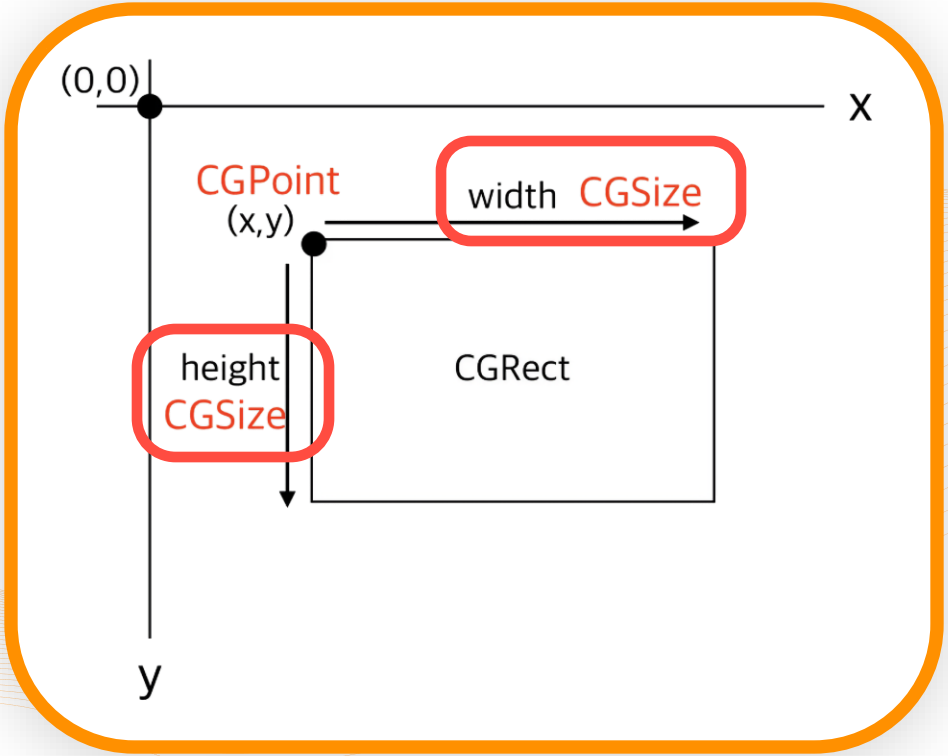
CGRect

1С-Коннект
Программист iOS



CGRect

1С-Коннект
Программист iOS



CGRect

1С-Коннект
Программист iOS

```
let myView = UIView(frame: CGRect(x: 20, y: 50, width: 200, height: 100))
```

1С-Коннект
Программист iOS

1С-Коннект
Программист iOS

```swift
class AutoLayoutView: UIView {
    private let blueView: UIView = {
        let view = UIView()
        view.backgroundColor = .systemBlue
        view.translatesAutoresizingMaskIntoConstraints = false
        return view
    }()

    override init(frame: CGRect) {
        super.init(frame: frame)
        backgroundColor = .white
        setupUI()
    }

    required init?(coder: NSCoder) {
        fatalError("init(coder:) has not been implemented")
    }

    private func setupUI() {
        addSubview(blueView)

        // Задаем констрейнты для blueView: отступ 10pt со всех сторон
        NSLayoutConstraint.activate([
            blueView.topAnchor.constraint(equalTo: topAnchor, constant: 10),
            blueView.leadingAnchor.constraint(equalTo: leadingAnchor, constant: 10),
            blueView.trailingAnchor.constraint(equalTo: trailingAnchor, constant: -10),
            blueView.bottomAnchor.constraint(equalTo: bottomAnchor, constant: -10)
        ])
    }
}
```

1С-Коннект
Программист iOS

```swift
class FrameView: UIView {
    private let blueView: UIView = {
        let view = UIView()
        view.backgroundColor = .systemBlue
        return view
    }()

    private let padding: CGFloat = 10

    override init(frame: CGRect) {
        super.init(frame: frame)
        backgroundColor = .white
        addSubview(blueView)
    }
    required init?(coder: NSCoder) {
        fatalError("init(coder:) has not been implemented")
    }

    override func layoutSubviews() {
        super.layoutSubviews()
        // Ручной расчет фрейма синей вью с отступами 10pt со всех сторон
        let x = padding
        let y = padding
        let width  = bounds.width  - padding * 2
        let height = bounds.height - padding * 2
        blueView.frame = CGRect(
            x: x,
            y: y,
            width: max(0, width),
            height: max(0, height)
        )
    }
}
```

1С-Коннект
Программист iOS

Голтаева Катерина

№ 2339 Консультация

Москвин Ян
1С-Коннект

Консультация по релизу с тикетами

25.06.2025 16:21 Новая

17:14

1С-Коннект
Программист iOS

UICollectionViewFlowLayout

1С-Коннект
Программист iOS

Search documentation

UIKit › Views and controls › Collection views › Layouts › UICollectionViewFlowLayout

Language: Swift | Objective-C

Class

UICollectionViewFlowLayout

A layout object that organizes items into a grid with optional header and footer views for each section.

iOS 6.0+ | iPadOS 6.0+ | Mac Catalyst 13.1+ | tvOS | visionOS 1.0+

```
@MainActor
class UICollectionViewFlowLayout
```

Overview

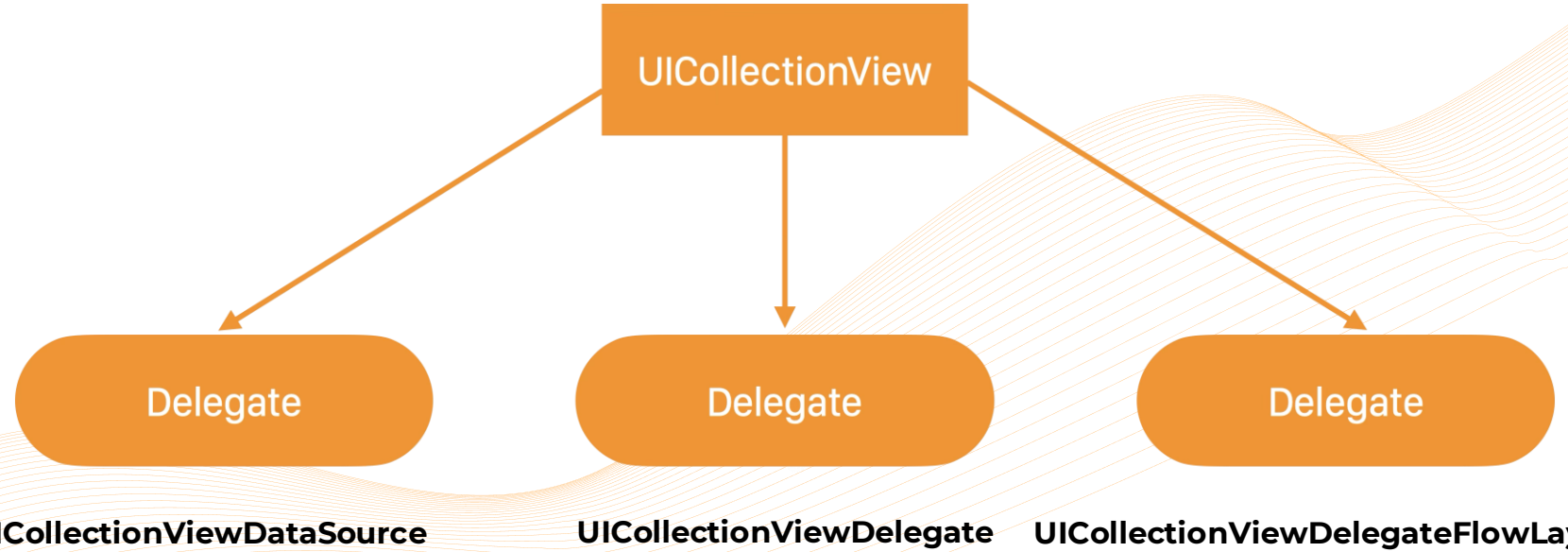
A flow layout is a type of collection view layout. Items in the collection view flow from one row or column (depending on the scrolling direction) to the next, with each row containing as many cells as will fit. Cells can be the same sizes or different sizes.

A flow layout works with the collection view's delegate object to determine the size of items, headers, and footers in each section and grid. That delegate object must conform to the `UICollectionViewDelegateFlowLayout` protocol. Use of the delegate allows you to adjust layout information dynamically. For example, you use a delegate object to specify different sizes for items in the grid. If you don't provide a delegate, the flow layout uses the default values you set in the properties of this class.

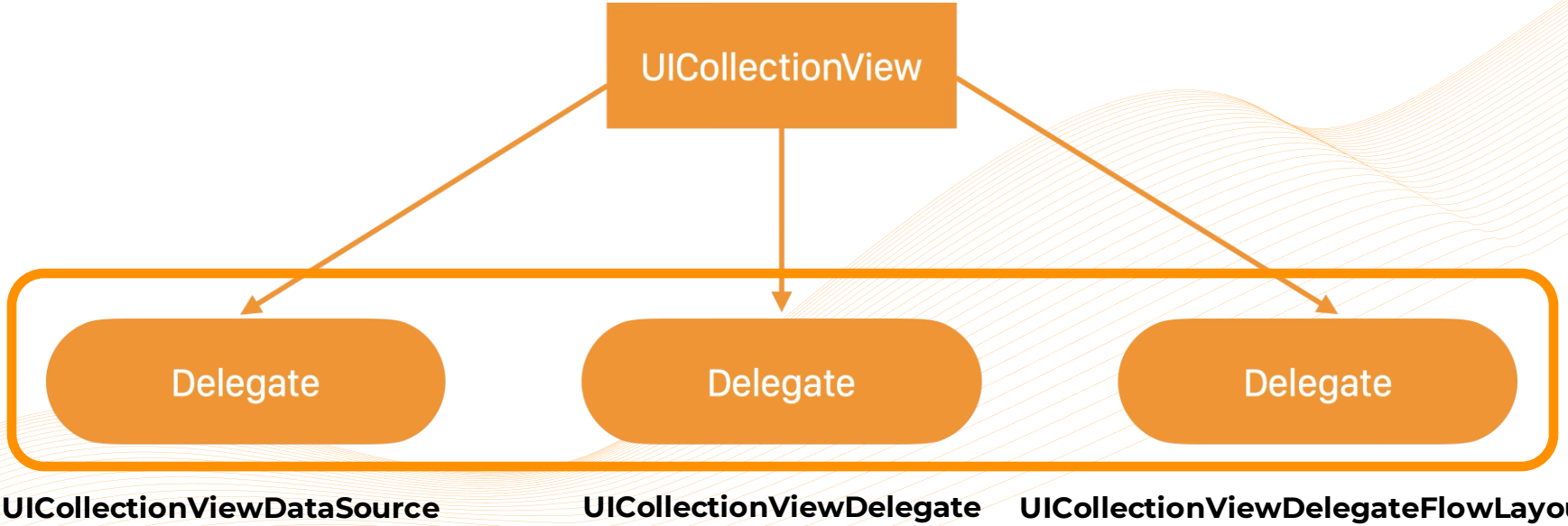
Flow layouts lay out their content using a fixed distance in one direction and a scrollable distance in the other. For example, in a vertically scrolling grid, the width of the grid content is constrained to the width of the corresponding collection view while the height of the content adjusts dynamically to match the number of sections and items in the grid. The layout scrolls vertically by default, but you can configure the scrolling direction using the `scrollDirection` property.

Each section in a flow layout can have its own custom header and footer. To configure the header or footer for a view, configure the size

1С-Коннект
Программист iOS



1С-Коннект
Программист iOS



1С-Коннект
Программист iOS

```swift
extension ChatViewController: UICollectionViewDataSource {

    func numberOfSections(in collectionView: UICollectionView) -> Int { ••• }

    func collectionView(
        _ collectionView: UICollectionView,
        numberOfItemsInSection section: Int
    ) -> Int { ••• }

    func collectionView(
        _ collectionView: UICollectionView,
        cellForItemAt indexPath: IndexPath
    ) -> UICollectionViewCell { ••• }

    func collectionView(
        _ collectionView: UICollectionView,
        viewForSupplementaryElementOfKind kind: String,
        at indexPath: IndexPath
    ) -> UICollectionReusableView { ••• }
}
```

1С-Коннект
Программист iOS

```
extension ChatViewController: UICollectionViewDelegate {
    func collectionView(
        _ collectionView: UICollectionView,
        willDisplay cell: UICollectionViewCell,
        forItemAt indexPath: IndexPath
    ) { ••• }
}
```

1С-Коннект
Программист iOS

```swift
extension ChatBotKeyboardView: UICollectionViewDelegateFlowLayout {
    func collectionView(
        _ collectionView: UICollectionView,
        layout collectionViewLayout: UICollectionViewLayout,
        sizeForItemAt indexPath: IndexPath
    ) -> CGSize { ••• }

    func collectionView(
        _ collectionView: UICollectionView,
        layout collectionViewLayout: UICollectionViewLayout,
        referenceSizeForFooterInSection section: Int
    ) -> CGSize { ••• }
}
```

1С-Коннект
Программист iOS

UICollectionViewFlowLayout позволяет

- Задать размер ячейки

1С-Коннект
Программист iOS

UICollectionViewFlowLayout позволяет

- Задать размер ячейки
- Установить минимальные отступы между ячейками

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewFlowLayout позволяет

- Задать размер ячейки
- Установить минимальные отступы между ячейками
- Установить отступы между секциями

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewFlowLayout позволяет

- Задать размер ячейки
- Установить минимальные отступы между ячейками
- Установить отступы между секциями
- Добавить header и footer для секции

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewFlowLayout позволяет

- Задать размер ячейки
- Установить минимальные отступы между ячейками
- Установить отступы между секциями
- Добавить header и footer для секции
- Сделать header и footer плавающими

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewFlowLayout не позволяет

- Добавить декоративные элементы

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

- Задать размер ячейки

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

- Задать размер ячейки
- Задать количество строк и колонок

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

- Задать размер ячейки
- Задать количество строк и колонок
- Задать отступы между секциями, ячейками и группами

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

- Задать размер ячейки
- Задать количество строк и колонок
- Задать отступы между секциями, ячейками и группами
- Добавить декоративные элементы

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

- Задать размер ячейки
- Задать количество строк и колонок
- Задать отступы между секциями, ячейками и группами
- Добавить декоративные элементы
- Добавить header и footer для секции, сделать их плавающими

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout позволяет

- Задать размер ячейки
- Задать количество строк и колонок
- Задать отступы между секциями, ячейками и группами
- Добавить декоративные элементы
- Добавить header и footer для секции, сделать их плавающими
- Добавить дополнительные элементы

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

Пару недель

1С-Коннект
Программист iOS

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout не позволяет

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout не позволяет

- Гибко управлять декоративными элементами

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout не позволяет

- Гибко управлять декоративными элементами
- Управлять плавающими элементами

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout не позволяет

- Гибко управлять декоративными элементами
- Управлять плавающими элементами
- Управлять скролом

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout не позволяет

- Гибко управлять декоративными элементами
- Управлять плавающими элементами
- Управлять скролом
- Управлять анимациями

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewCompositionalLayout не подходит

1С-Коннект
Программист iOS

Auto Layout ячейки – дорога в тупик

1С-Коннект
Программист iOS

Что делать дальше?

1С-Коннект
Программист iOS

Начать все с начала...

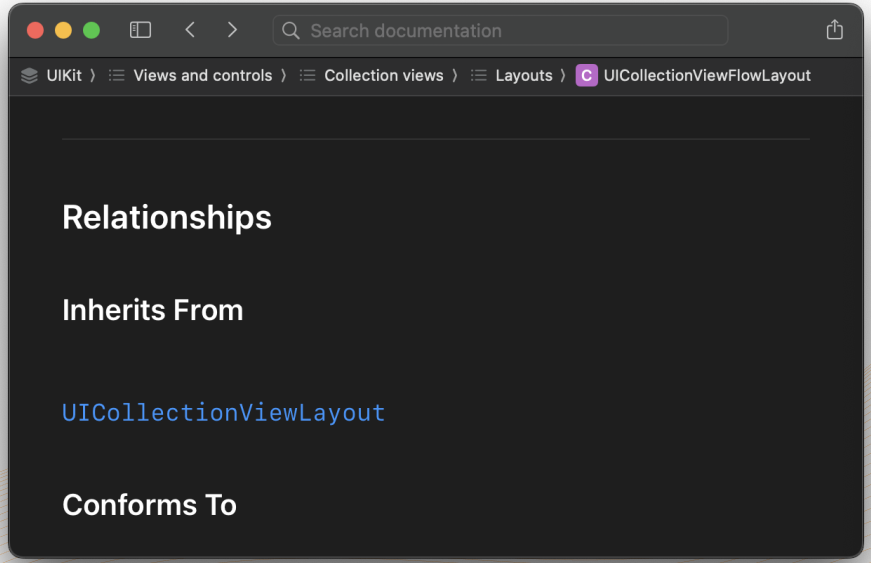
1С-Коннект
Программист iOS

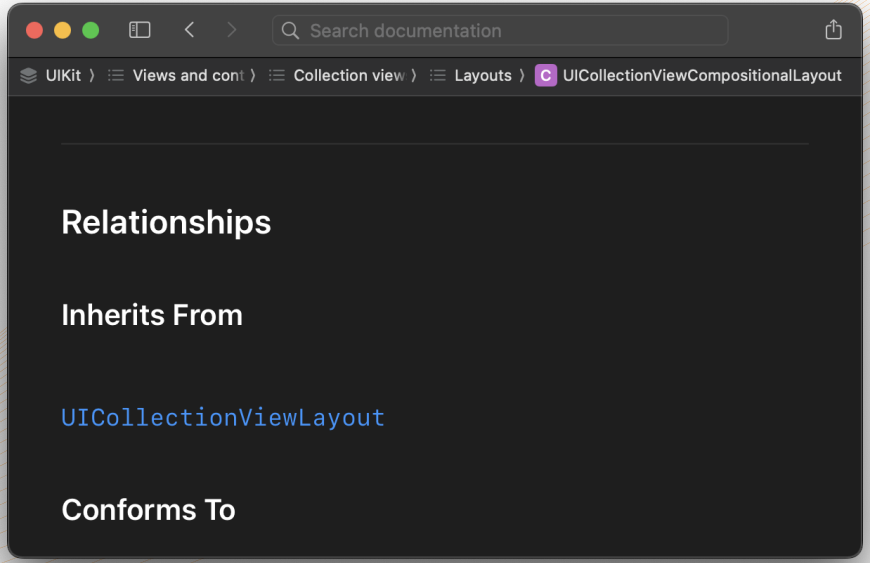
Верстка с помощью frame: писать сложно – работает быстро

1С-Коннект
Программист iOS

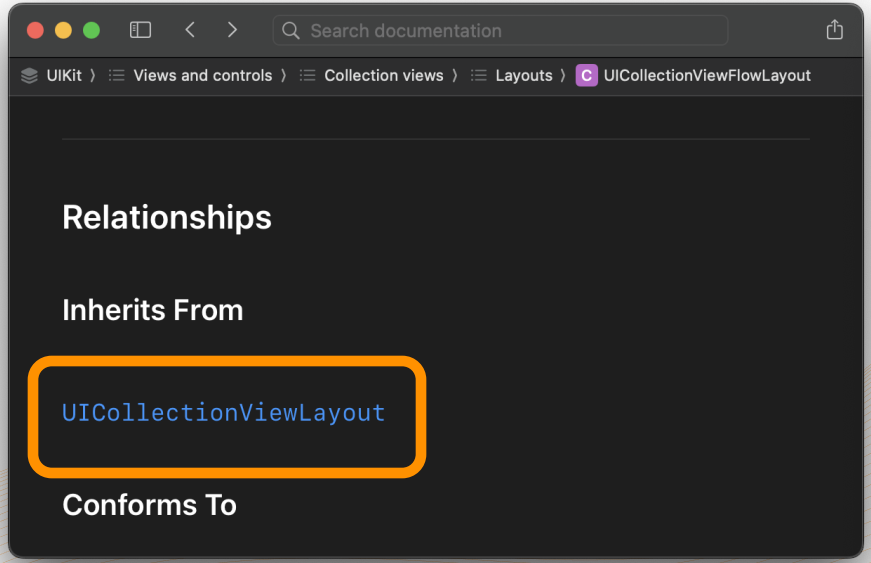
Что делать с Layout?

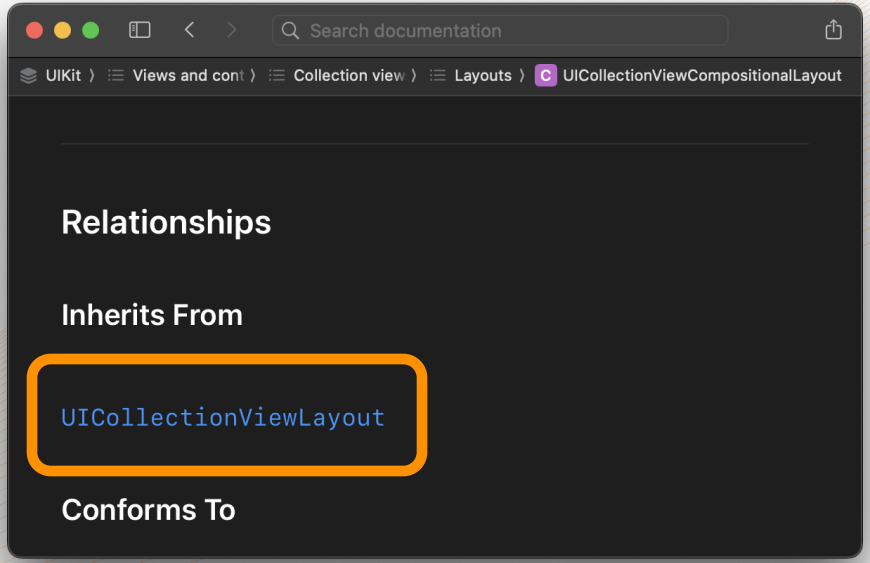
1С-Коннект
Программист iOS





1С-Коннект
Программист iOS





1С-Коннект
Программист iOS

UICollectionViewLayout

1С-Коннект
Программист iOS

Search documentation

UIKit › Views and controls › Collection views › Layouts › C UICollectionViewLayout

Overview

A layout object determines the placement of cells, supplementary views, and decoration views inside the collection view's bounds and reports that information to the collection view. The collection view then applies the provided layout information to the corresponding views so that they can be presented onscreen.

You must subclass `UICollectionViewLayout` in order to use it. Before you consider subclassing, however, consider whether you can adapt `UICollectionViewCompositionalLayout` to your layout needs.

Subclassing notes

The layout object defines the position, size, and visual state of items in the collection view, based on the design of the layout. The views for the layout are created by the collection view's data source.

You lay out three types of visual elements in a collection view:

- *Cells* are the main elements positioned by the layout. Each cell represents a single data item in the collection. You can make cells interactive so that a user can perform actions like selecting, dragging, and reordering the cells. A collection view can have a single group of cells, or you can divide those cells into multiple sections. The layout object arranges the cells in the collection view's content area.
- *Supplementary views* present data but can't be selected by the user. You use supplementary views to implement things like header and footer views for a given section or for the entire collection view. Supplementary views are optional and their use and placement is defined by the layout object.
- *Decoration views* are visual adornments, like badges, that can't be selected and aren't inherently tied to the data of the collection view. Decoration views are another type of supplementary view. Like supplementary views, they're optional and their use and placement is defined by the layout object.

The collection view asks its layout object to provide layout information for these elements at many different times. Every cell and view

1С-Коннект
Программист iOS

Search documentation

UIKit › Views and controls › Collection views › Layouts › C UICollectionViewLayout

Methods to override

Every layout object should implement the following methods:

- `collectionViewContentSize`
- `layoutAttributesForElements(in:)`
- `layoutAttributesForItem(at:)`
- `layoutAttributesForSupplementaryView(ofKind:at:)` (if your layout supports supplementary views)
- `layoutAttributesForDecorationView(ofKind:at:)` (if your layout supports decoration views)
- `shouldInvalidateLayout(forBoundsChange:)`

These methods provide the fundamental layout information that the collection view needs to place contents on the screen. If your layout doesn't support supplementary or decoration views, don't implement the corresponding methods.

When the data in the collection view changes and items are to be inserted or deleted, the collection view asks its layout object to update the layout information. Specifically, any item that's moved, added, or deleted must have its layout information updated to reflect its new location. For moved items, the collection view uses the standard methods to retrieve the item's updated layout attributes. For items being inserted or deleted, the collection view calls some different methods, which you should override to provide the appropriate layout information:

- `initialLayoutAttributesForAppearingItem(at:)`
- `initialLayoutAttributesForAppearingSupplementaryElement(ofKind:at:)`
- `initialLayoutAttributesForAppearingDecorationElement(ofKind:at:)`
- `finalLayoutAttributesForDisappearingItem(at:)`
- `finalLayoutAttributesForDisappearingSupplementaryElement(ofKind:at:)`
- `finalLayoutAttributesForDisappearingDecorationElement(ofKind:at:)`

In addition to these methods, you can also override the `prepare(forCollectionViewUpdates:)` to handle any layout-related preparation. You can also override the `finalizeCollectionViewUpdates()` method and use it to add animations to the overall animation block or to implement any final layout-related tasks.

1С-Коннект
Программист iOS

Search documentation

UIKit › Views and controls › Collection views › Layouts › C UICollectionViewLayout

Optimizing layout performance using invalidation contexts

When designing your custom layouts, you can improve performance by invalidating only those parts of your layout that actually changed. When you change items, calling the `invalidateLayout()` method forces the collection view to recompute all of its layout information and reapply it. A better solution is to recompute only the layout information that changed, which is exactly what invalidation contexts allow you to do. An invalidation context lets you specify which parts of the layout changed. The layout object can then use that information to minimize the amount of data it recomputes.

To define a custom invalidation context for your layout, subclass the `UICollectionViewLayoutInvalidationContext` class. In your subclass, define custom properties that represent the parts of your layout data that can be recomputed independently. When you need to invalidate your layout at runtime, create an instance of your invalidation context subclass, configure the custom properties based on what layout information changed, and pass that object to your layout's `invalidateLayout(with:)` method. Your custom implementation of that method can use the information in the invalidation context to recompute only the portions of your layout that changed.

If you define a custom invalidation context class for your layout object, you should also override the `invalidationContextClass` method and return your custom class. The collection view always creates an instance of the class you specify when it needs an invalidation context. Returning your custom subclass from this method ensures that your layout object always has the invalidation context it expects.

1С-Коннект
Программист iOS

1С-Коннект
Программист iOS

Примеры кода:

- Евгений Ёлчев – UICollectionViewLayout from scratch
- Евгений Казаев – Мой Covid-19 lockdown проект, или, как я полез в кастомный UICollectionViewLayout и получил ChatLayout
- Репозиторий MagazineLayout от Airbnb
- Рубанов Михаил – UICollectionViewLayout для пиццы из разных половинок

1С-Коннект
Программист iOS

UICollectionViewLayout позволяет

- Все что угодно

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

1С-Коннект
Программист iOS

UICollectionViewLayout сложный

- Все нужно делать самостоятельно

https://mobiusconf.com/archive/2024%20Spring/talks/7e416ba8a7ae45af81988fca2bfeb003/?referer=%2Farchive%2F2024%2520Spring%2Fschedule%2Fdays%2F

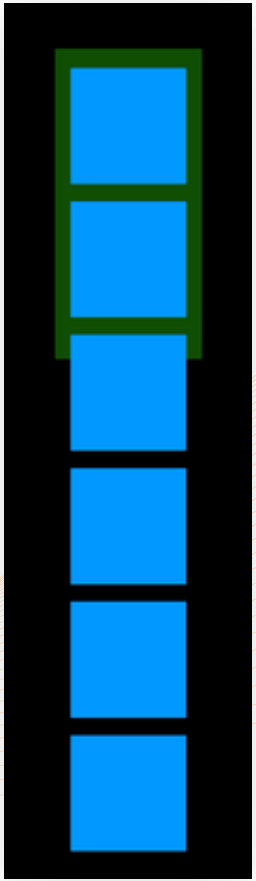
1С-Коннект
Программист iOS

Начинаем делать

1С-Коннект
Программист iOS

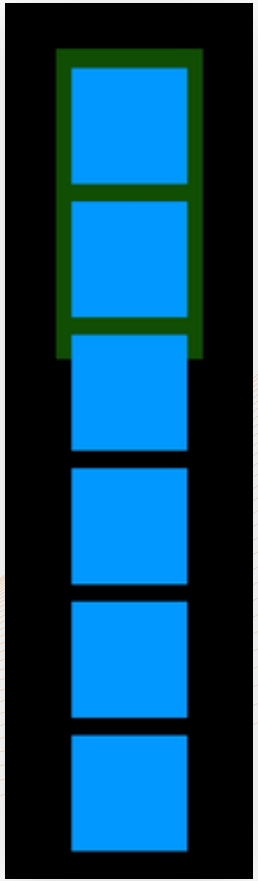
ConnectChatLayout

1С-Коннект
Программист iOS



https://habr.com/ru/companies/dododev/articles/452876/

1С-Коннект
Программист iOS



https://habr.com/ru/companies/dododev/articles/452876/

1С-Коннект
Программист iOS

```
class UICollectionViewLayoutAttributes {

    var frame: CGRect

    var alpha  CGFloat

    var zIndex  Int

    ...

}
```

1С-Коннект
Программист iOS

```
class UICollectionViewLayoutAttributes {

    var frame: CGRect

    var alpha  CGFloat

    var zIndex  Int

    ...

}
```

1С-Коннект
Программист iOS

```
class UICollectionViewLayoutAttributes {

    var frame: CGRect

    var alpha  CGFloat

    var zIndex  Int

    ...

}
```

1С-Коннект
Программист iOS

```
class UICollectionViewLayoutAttributes {

    var frame: CGRect

    var alpha  CGFloat

    var zIndex  Int

    ...

}
```

1С-Коннект
Программист iOS

```
class UICollectionViewLayoutAttributes {

    var frame: CGRect

    var alpha  CGFloat

    var zIndex  Int

    ...

}
```

1С-Коннект
Программист iOS

Как отобразить хоть что-то?

1С-Коннект
Программист iOS

```swift
/// Ширина и высота содержимого collection view.
override var collectionViewContentSize: CGSize { ••• }

/// Сообщает объекту layout, что нужно обновить текущий макет.
override func prepare() { ••• }

/// Получает атрибуты макета для всех ячеек и вью внутри указанного прямоугольника.
override func layoutAttributesForElements(
    in rect: CGRect
) -> [UICollectionViewLayoutAttributes]? { ••• }

/// Получает информацию о макете для элемента по указанному indexPath и соответствующей ячейке.
override func layoutAttributesForItem(
    at indexPath: IndexPath
) -> UICollectionViewLayoutAttributes? { ••• }
```

1С-Коннект
Программист iOS

```swift
/// Ширина и высота содержимого collection view.
override var collectionViewContentSize: CGSize { ••• }

/// Сообщает объекту layout, что нужно обновить текущий макет.
override func prepare() { ••• }

/// Получает атрибуты макета для всех ячеек и вью внутри указанного прямоугольника.
override func layoutAttributesForElements(
    in rect: CGRect
) -> [UICollectionViewLayoutAttributes]? { ••• }

/// Получает информацию о макете для элемента по указанному indexPath и соответствующей ячейке.
override func layoutAttributesForItem(
    at indexPath: IndexPath
) -> UICollectionViewLayoutAttributes? { ••• }
```

1С-Коннект
Программист iOS

```swift
/// Ширина и высота содержимого collection view.
override var collectionViewContentSize: CGSize { ••• }

/// Сообщает объекту layout, что нужно обновить текущий макет.
override func prepare() { ••• }

/// Получает атрибуты макета для всех ячеек и вью внутри указанного прямоугольника.
override func layoutAttributesForElements(
    in rect: CGRect
) -> [UICollectionViewLayoutAttributes]? { ••• }

/// Получает информацию о макете для элемента по указанному indexPath и соответствующей ячейке.
override func layoutAttributesForItem(
    at indexPath: IndexPath
) -> UICollectionViewLayoutAttributes? { ••• }
```

1С-Коннект
Программист iOS

```swift
/// Ширина и высота содержимого collection view.
override var collectionViewContentSize: CGSize { ••• }

/// Сообщает объекту layout, что нужно обновить текущий макет.
override func prepare() { ••• }

/// Получает атрибуты макета для всех ячеек и вью внутри указанного прямоугольника.
override func layoutAttributesForElements(
    in rect: CGRect
) -> [UICollectionViewLayoutAttributes]? { ••• }

/// Получает информацию о макете для элемента по указанному indexPath и соответствующей ячейке.
override func layoutAttributesForItem(
    at indexPath: IndexPath
) -> UICollectionViewLayoutAttributes? { ••• }
```

1С-Коннект
Программист iOS

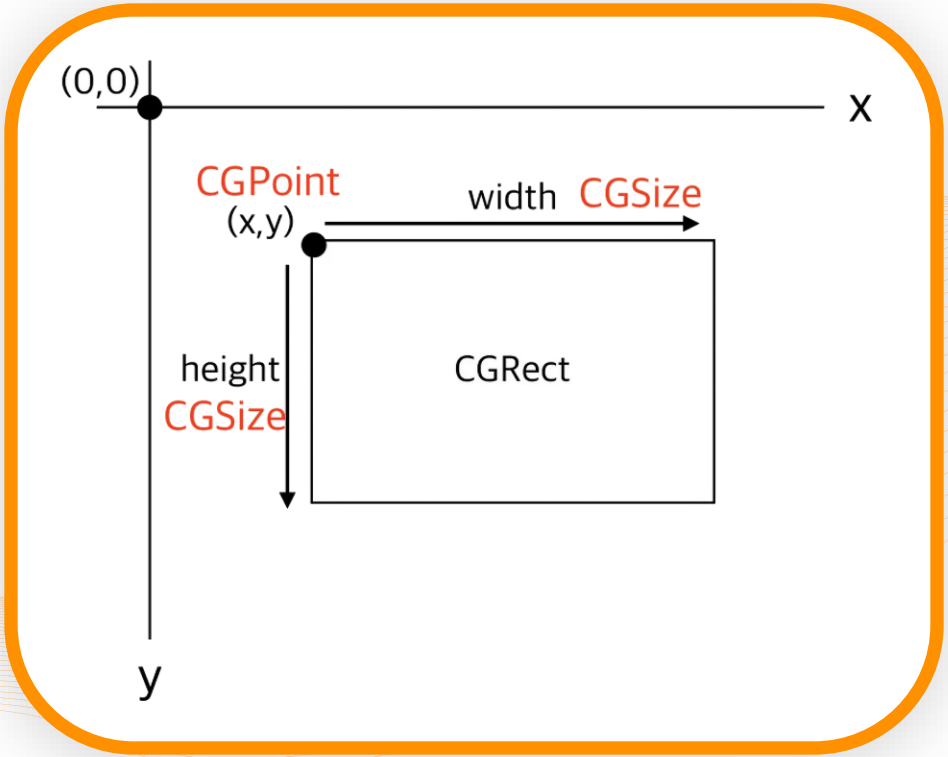
```swift
/// Ширина и высота содержимого collection view.
override var collectionViewContentSize: CGSize { ••• }

/// Сообщает объекту layout, что нужно обновить текущий макет.
override func prepare() { ••• }

/// Получает атрибуты макета для всех ячеек и вью внутри указанного прямоугольника.
override func layoutAttributesForElements(
    in rect: CGRect
) -> [UICollectionViewLayoutAttributes]? { ••• }

/// Получает информацию о макете для элемента по указанному indexPath и соответствующей ячейке.
override func layoutAttributesForItem(
    at indexPath: IndexPath
) -> UICollectionViewLayoutAttributes? { ••• }
```

1С-Коннект
Программист iOS

prepare()

1С-Коннект
Программист iOS

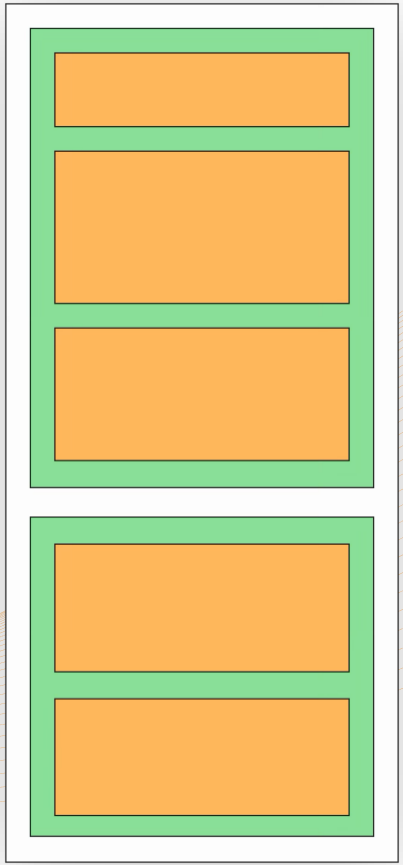
func prepare()



CGRect

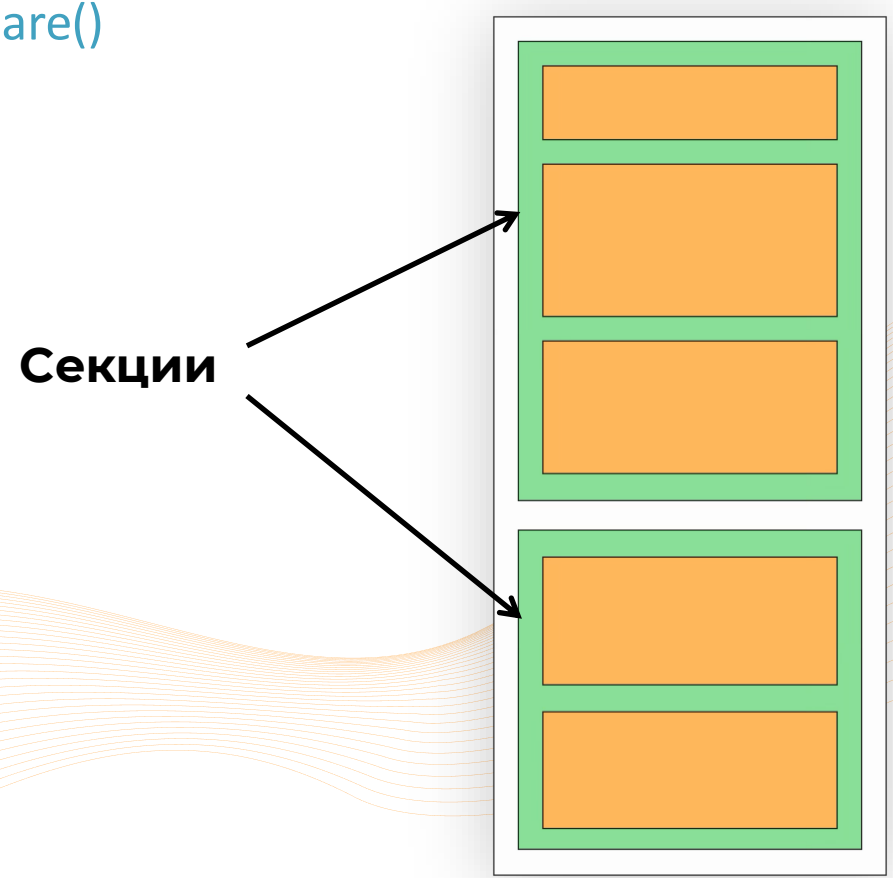
1С-Коннект
Программист iOS

func prepare()



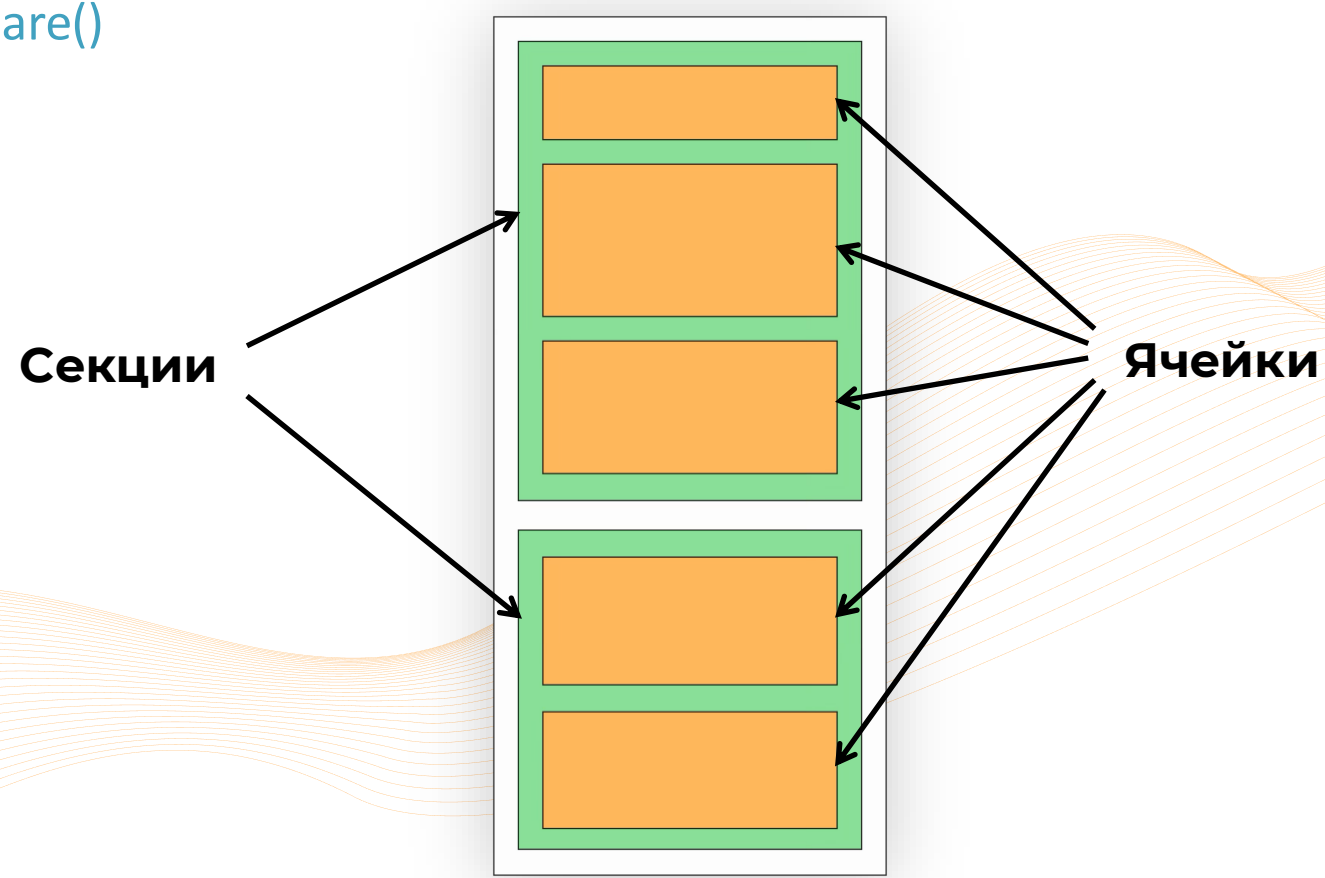
1С-Коннект
Программист iOS

func prepare()



1С-Коннект
Программист iOS

func prepare()

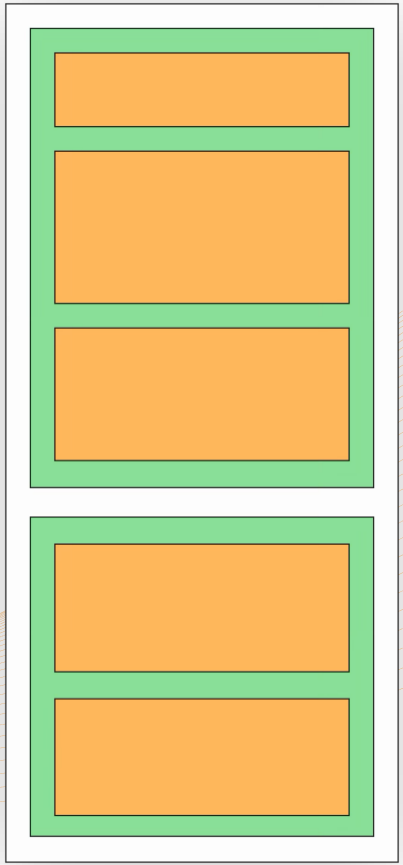


1С-Коннект
Программист iOS

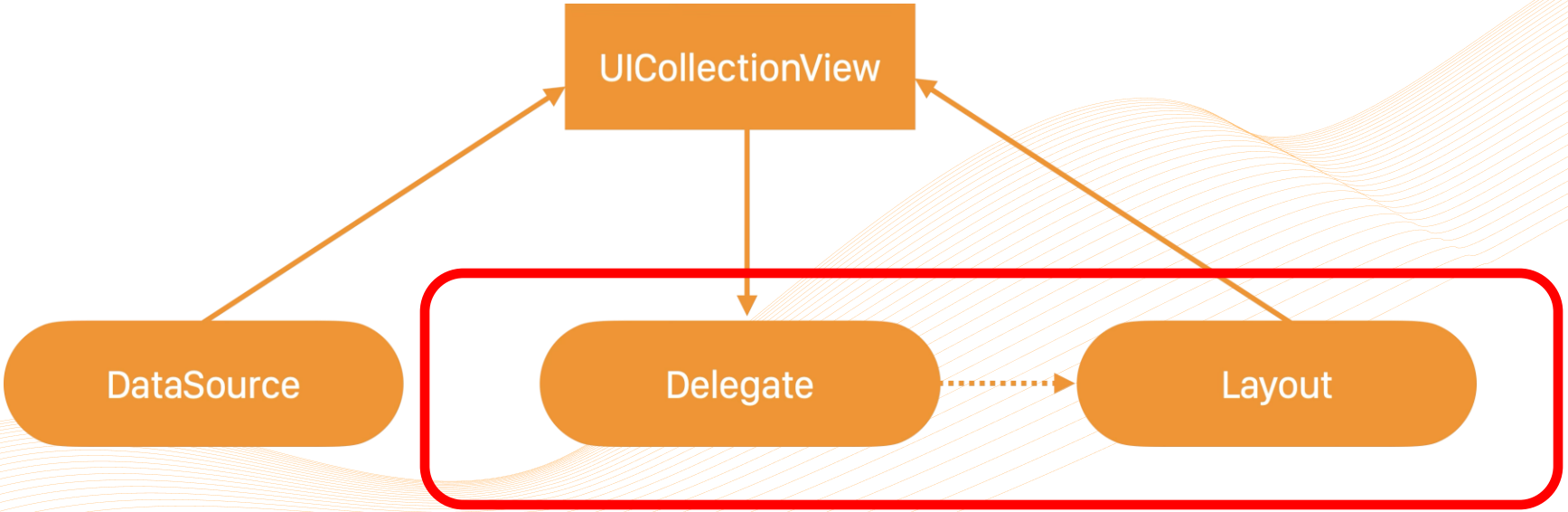
func prepare()

Сложности:

- **Динамическая высота ячеек**



1С-Коннект
Программист iOS




```
func getCellConfig(indexPath: IndexPath, width: Double) -> CellConfiguration
```

1С-Коннект
Программист iOS

```
template.isTemplateMode = true
template.isNeedAvatar = isNeedAvatar
let incomingBubbleInsets = template.incomingBubbleInsets
let outgoingBubbleInsets = template.outgoingBubbleInsets

template.frame.size = .max(width: width)
template.configure(from: model, maxSize: maxBubbleSize)
let height = template.getHeight()
template.model = nil

let config = CellConfiguration(
    itemSize: CGSize(width: width, height: height),
    authorId: model.authorId,
    isIncoming: model.isIncoming,
    incomingBubbleInsets: incomingBubbleInsets,
    outgoingBubbleInsets: outgoingBubbleInsets
)
cellConfigCache[hash] = config
return config
```

1С-Коннект
Программист iOS

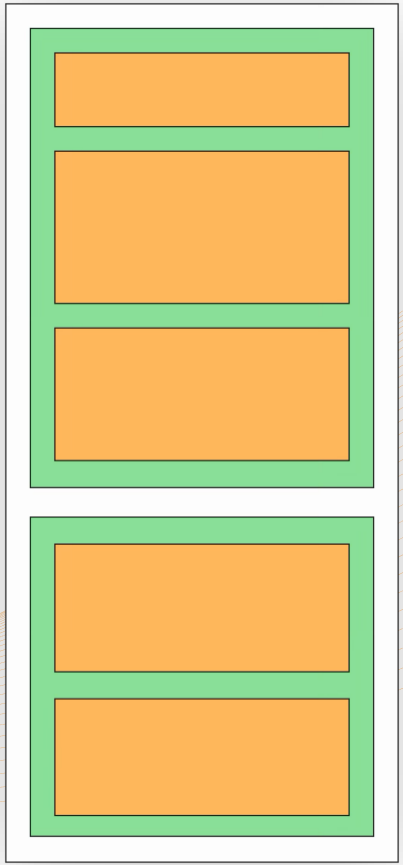
```
struct CellConfiguration {
    let itemSize: CGSize
    let authorId: String?
    let isIncoming: Bool?
    let incomingBubbleInsets: UIEdgeInsets
    let outgoingBubbleInsets: UIEdgeInsets
}
```

1С-Коннект
Программист iOS

func prepare()

Сложности:

- **Динамическая высота ячеек**
- **Нельзя рассчитать фрейм ячейки, не зная фрейм предыдущей ячейки**



1С-Коннект
Программист iOS

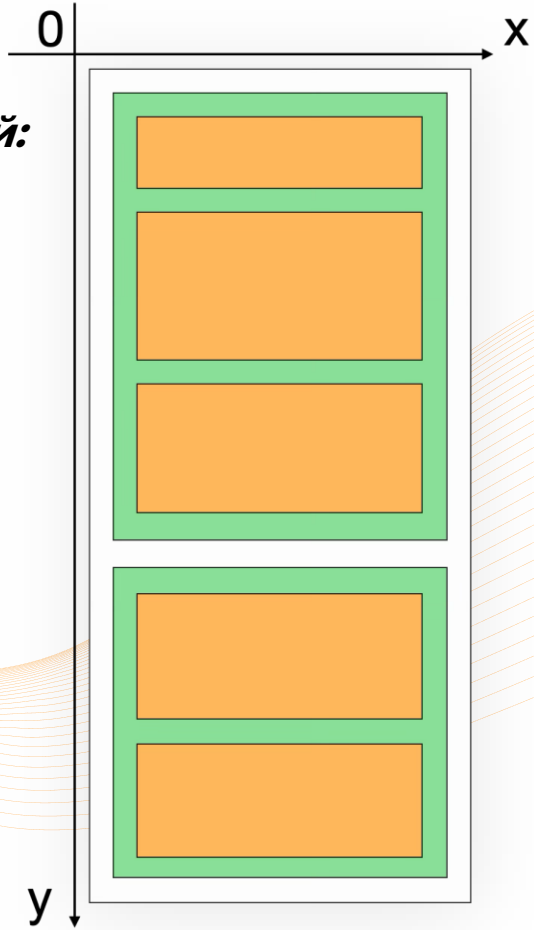
Пара хитростей

1С-Коннект
Программист iOS

func prepare()

Оптимизация вычислений:

- **Кеширование фрейма секции**



1С-Коннект
Программист iOS

```
class ChatSection {
    var items: [CustomLayoutAttributes]
    var header: CustomLayoutAttributes
    var frame: CGRect = .zero
    var minIndexChangedItem: Int? = nil

    init( ••• ) { ••• }

    func addChangedItem(index: Int) { ••• }

    func typedCopy() -> ChatSection { ••• }
}
```

1С-Коннект
Программист iOS

```
class ChatSection {
    var items: [CustomLayoutAttributes]
    var header: CustomLayoutAttributes
    var frame: CGRect = .zero
    var minIndexChangedItem: Int? = nil

    init( ••• ) { ••• }

    func addChangedItem(index: Int) { ••• }

    func typedCopy() -> ChatSection { ••• }
}
```

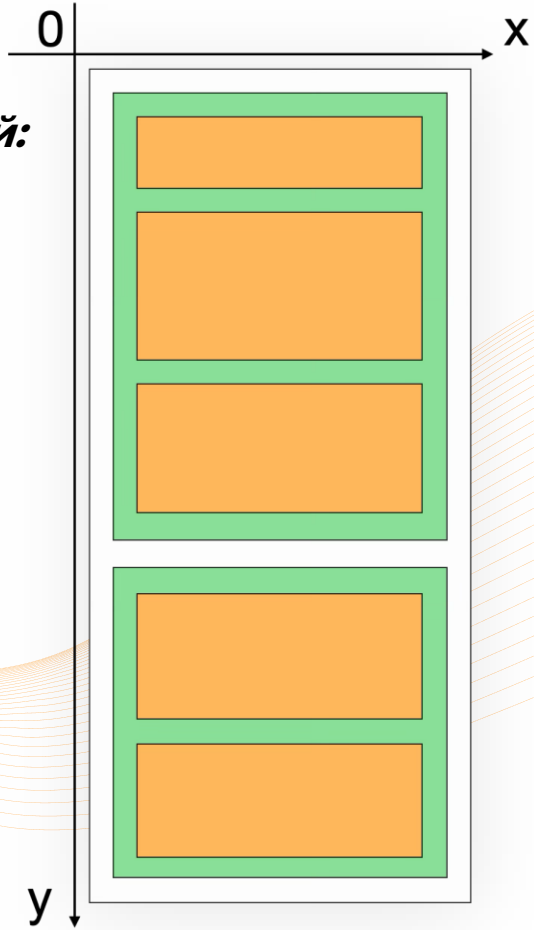
1С-Коннект
Программист iOS

func prepare()

Оптимизация вычислений:

- **Кеширование фрейма секции**

- **Расчет фрейма ячейки относительно секции**



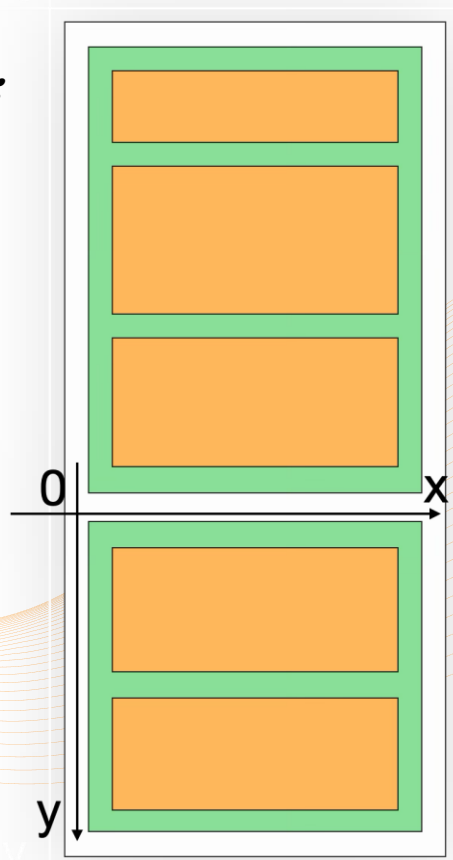
1С-Коннект
Программист iOS

func prepare()

Оптимизация вычислений:

- **Кеширование фрейма секции**

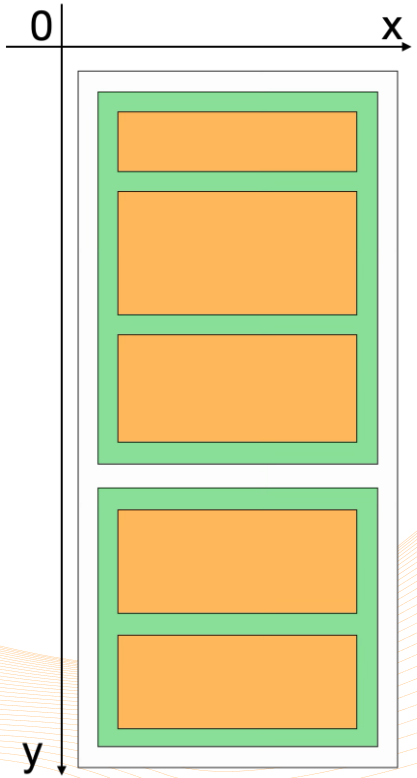
- **Расчет фрейма ячейки относительно секции**



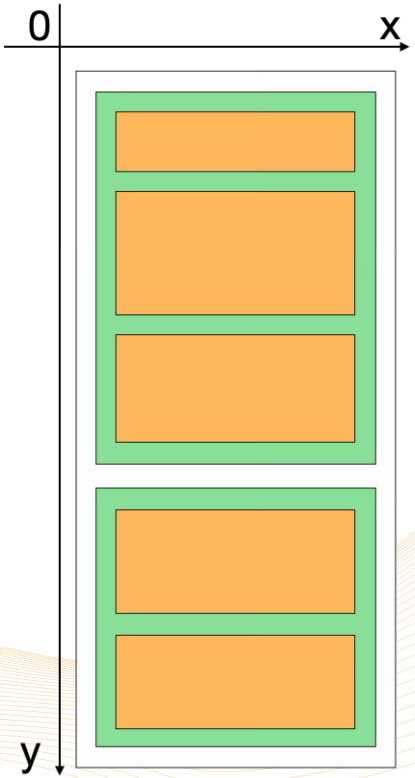
1С-Коннект
Программист iOS

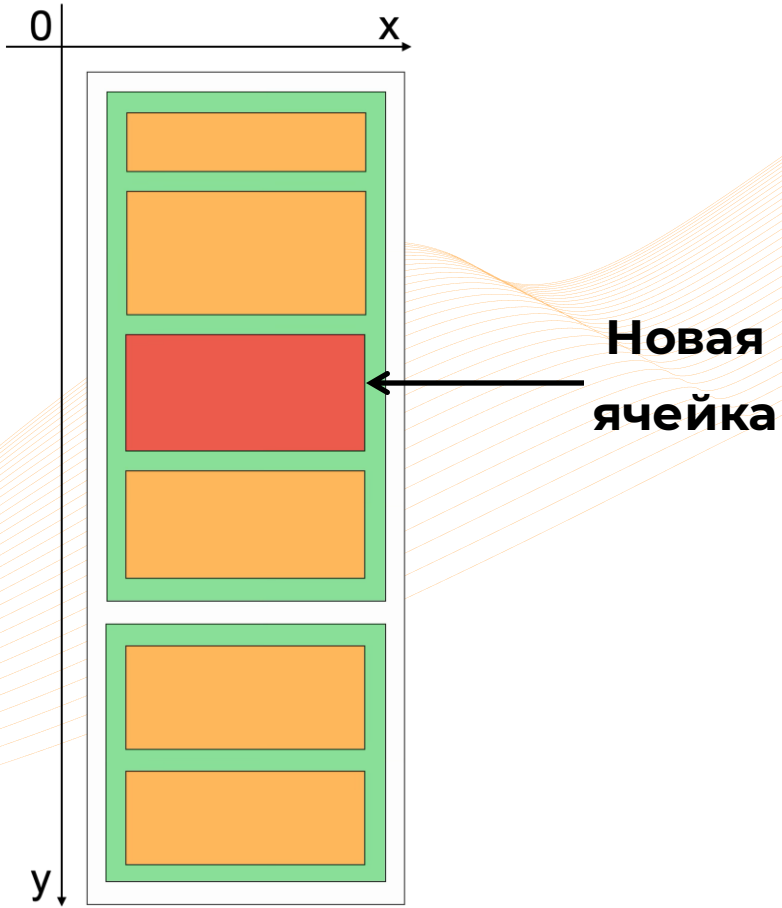
Для чего это нужно?

1С-Коннект
Программист iOS

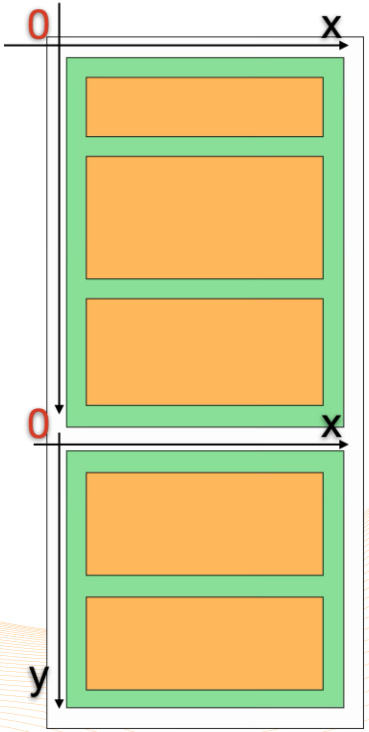


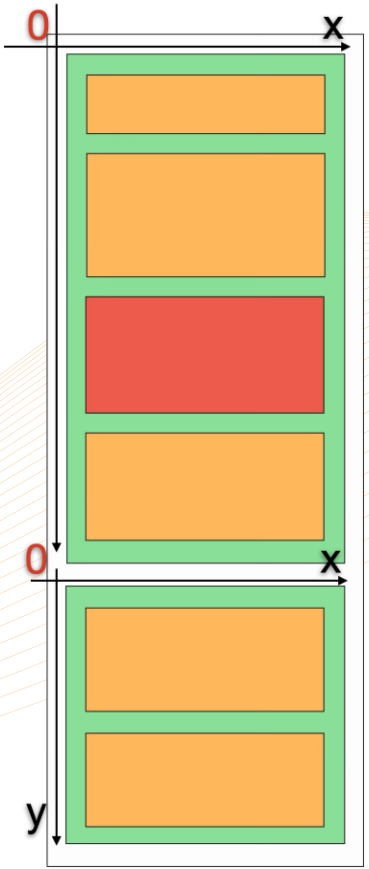
1С-Коннект
Программист iOS



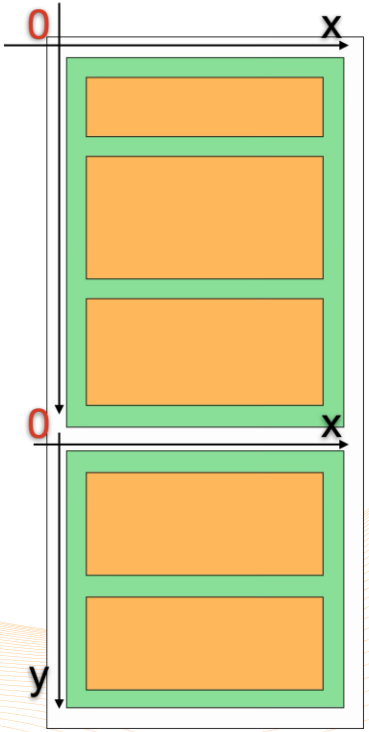


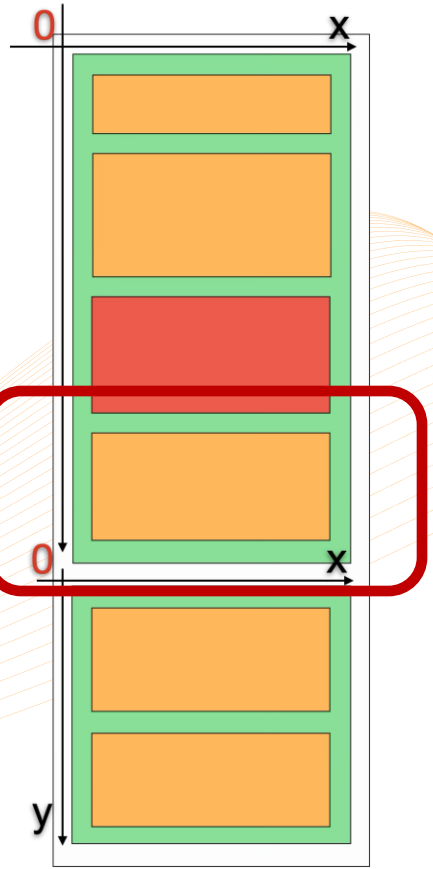
1С-Коннект
Программист iOS



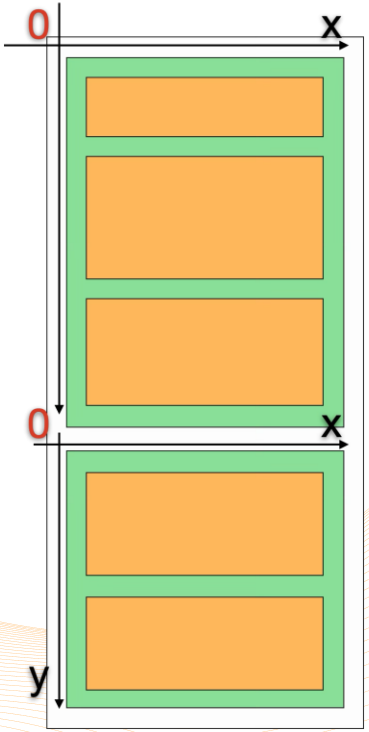


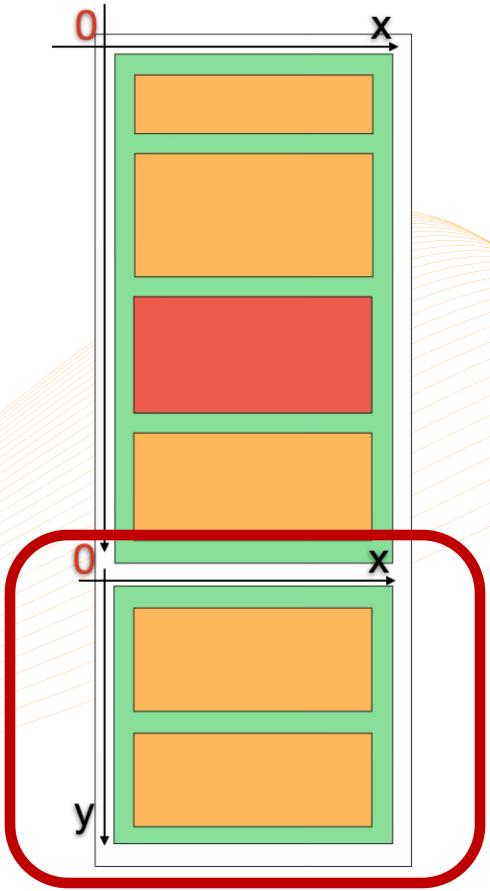
1С-Коннект
Программист iOS





1С-Коннект
Программист iOS



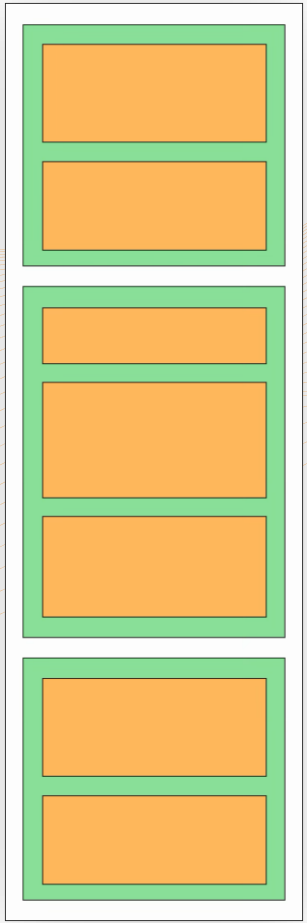


1С-Коннект
Программист iOS

collectionViewContentSize

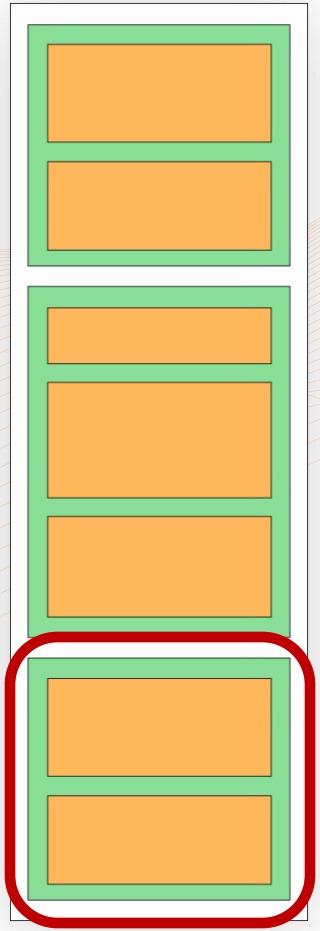
1С-Коннект
Программист iOS

- **Рассчитанный макет**



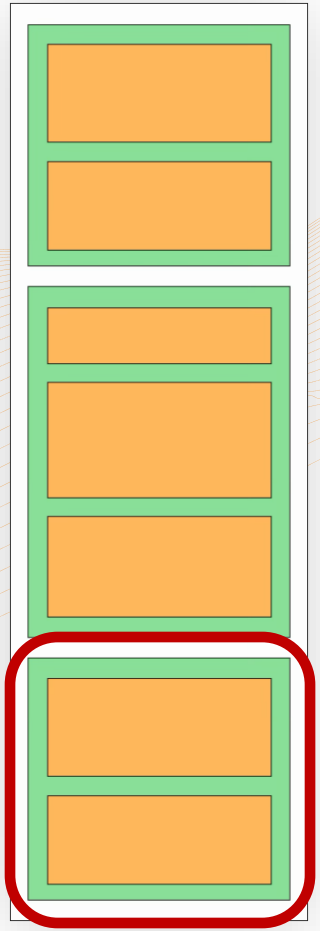
1С-Коннект
Программист iOS

- **Рассчитанный макет**
- **Берем фрейм последней секции**

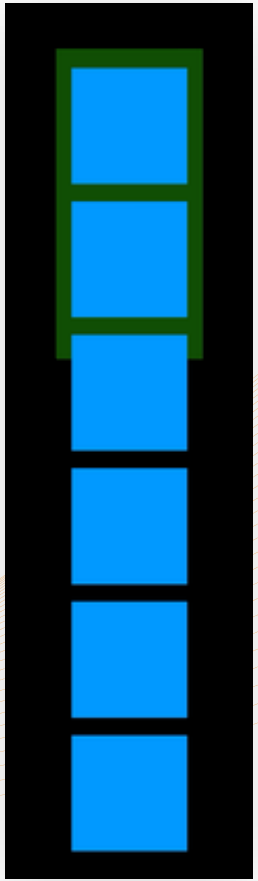


1С-Коннект
Программист iOS

- Рассчитанный макет
- Берем фрейм последней секции
- Возвращаем maxY этого фрейма



1С-Коннект
Программист iOS

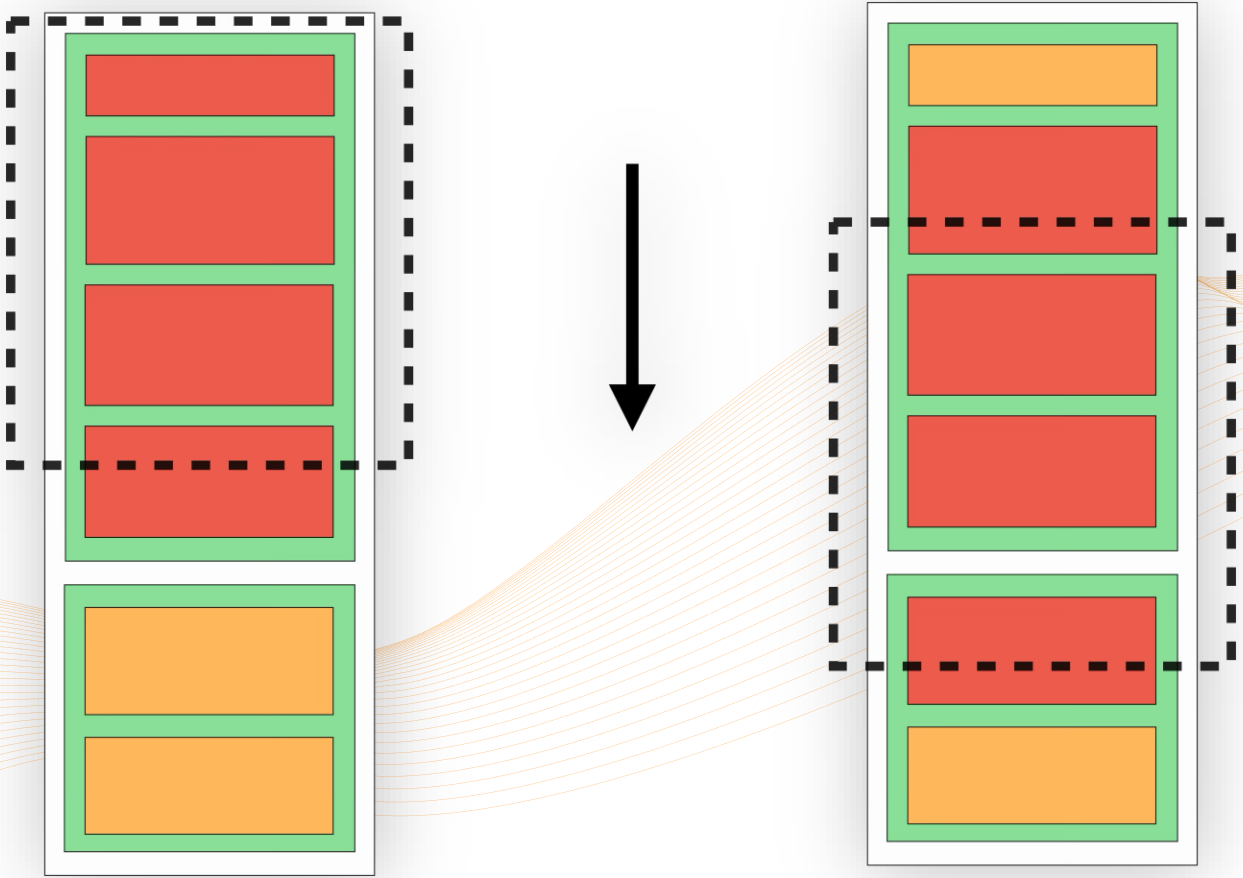


https://habr.com/ru/companies/dododev/articles/452876/

1С-Коннект
Программист iOS

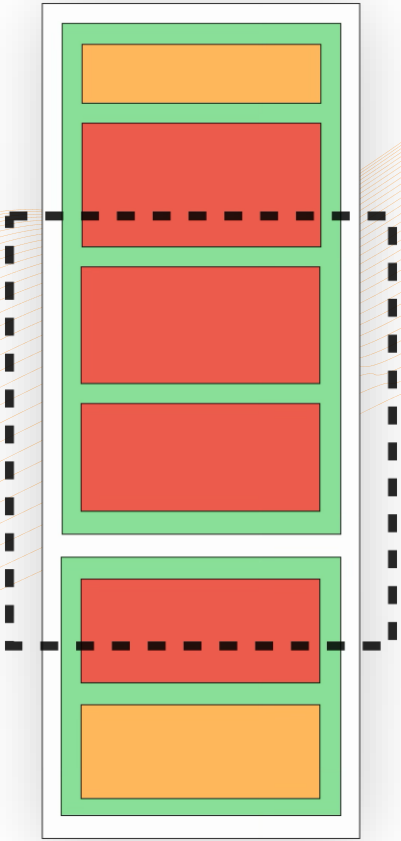
layoutAttributesForElements(in: rect)

1С-Коннект
Программист iOS



1С-Коннект
Программист iOS

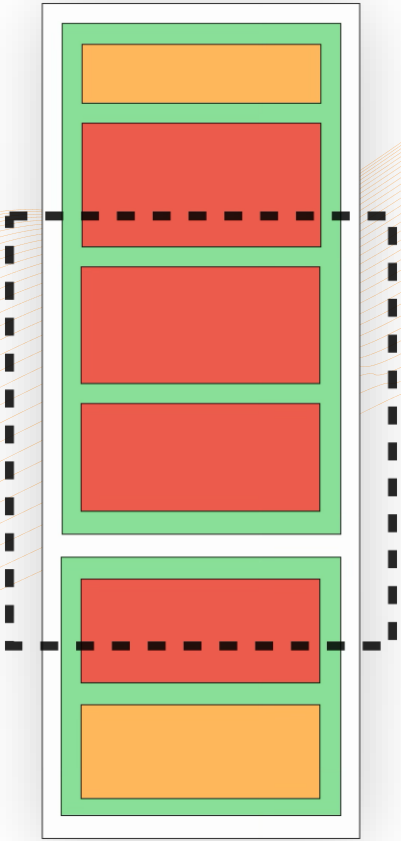
Что мы делаем:



1С-Коннект
Программист iOS

Что мы делаем:

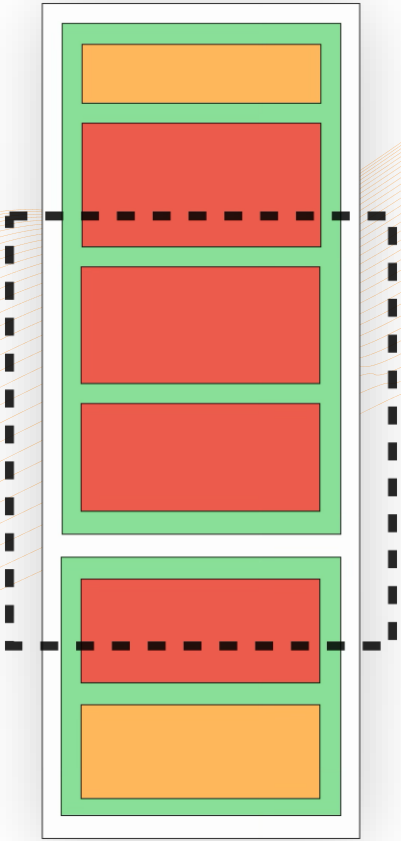
- **Берем все секции, фрейм которых пересекается с видимой областью**



1С-Коннект
Программист iOS

Что мы делаем:

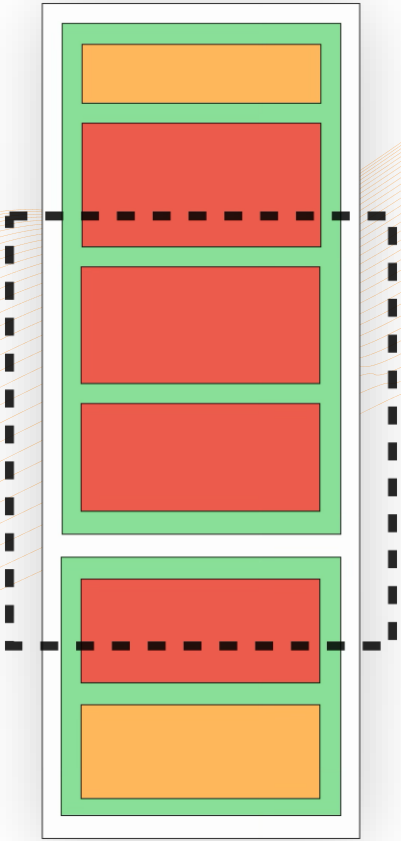
- **Берем все секции, фрейм которых пересекается с видимой областью**
- **Определяем с какого конца нужно перебирать секцию**



1С-Коннект
Программист iOS

Что мы делаем:

- Берем все секции, фрейм которых пересекается с видимой областью
- Определяем с какого конца нужно перебирать секцию
- Прибавляем к положению фрейма ячейки, положение секции и если фрейм пересекается, то отдаем его системе



1С-Коннект
Программист iOS

Как нарисовать сову

1.

2.

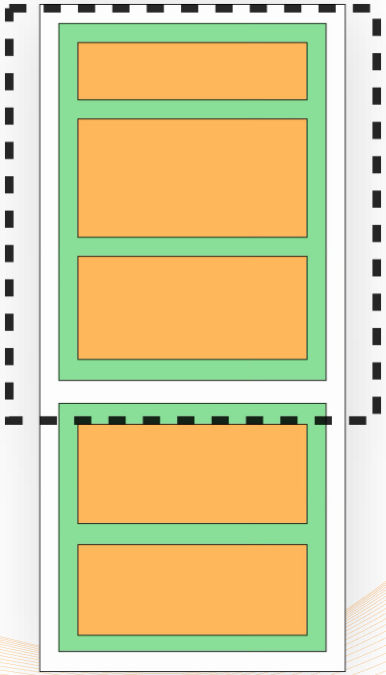
1. Рисуем кружочки

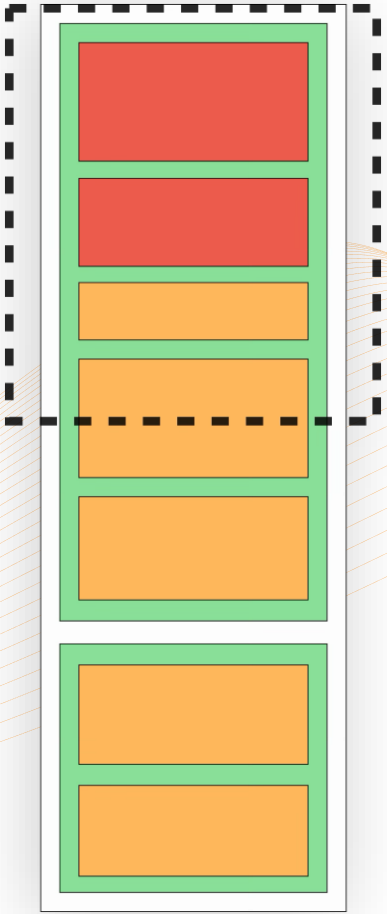
2. Рисуем остаток совы

1С-Коннект
Программист iOS

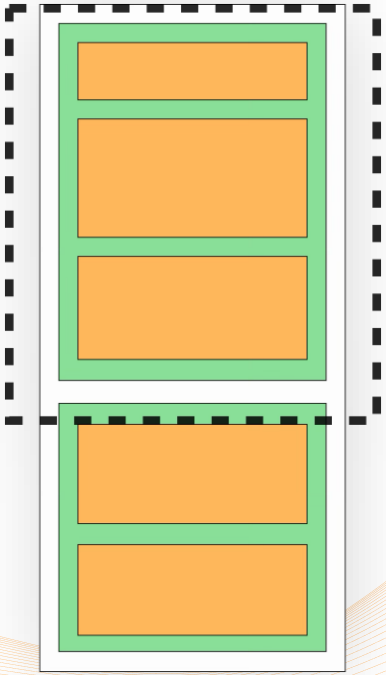
Подскрол при обновлении коллекции

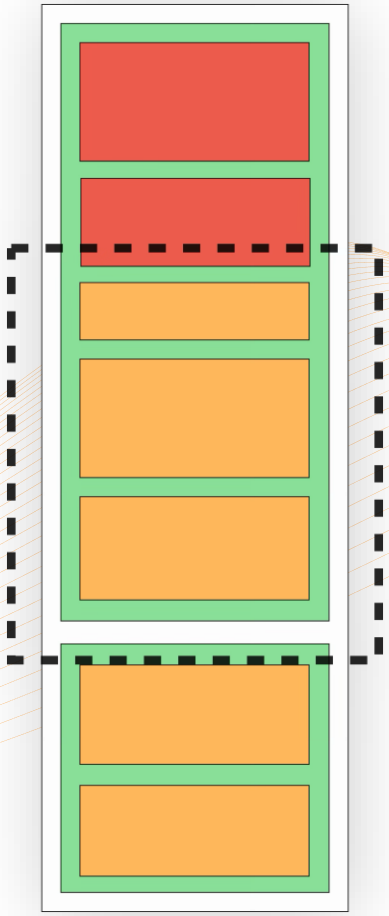
1С-Коннект
Программист iOS





1С-Коннект
Программист iOS





1С-Коннект
Программист iOS

Как это сделать?

1С-Коннект
Программист iOS

prepare(forCollectionViewUpdates:)

1С-Коннект
Программист iOS

prepare(forCollectionViewUpdates:)

- Обновление секции
- Обновление ячейка

- Удаление секции
- Удаление ячейки

- Вставка секции
- Вставка ячейки

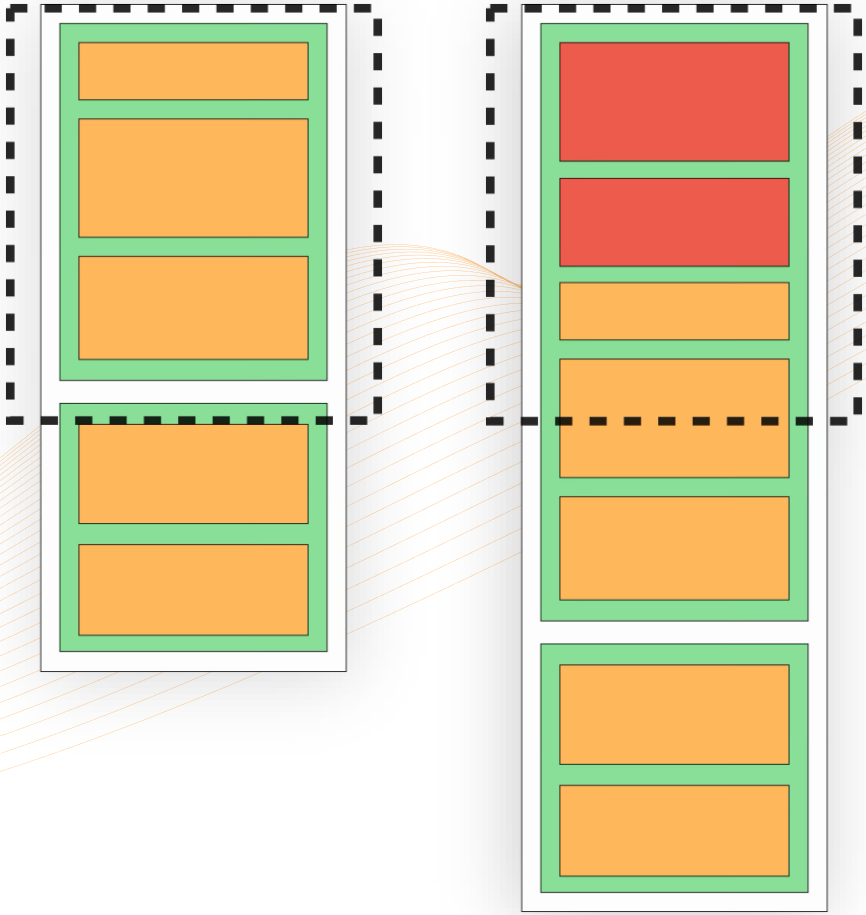
- Перемещение секции
- Перемещение ячейки

1С-Коннект
Программист iOS

targetContentOffset(forProposedContentOffset:)

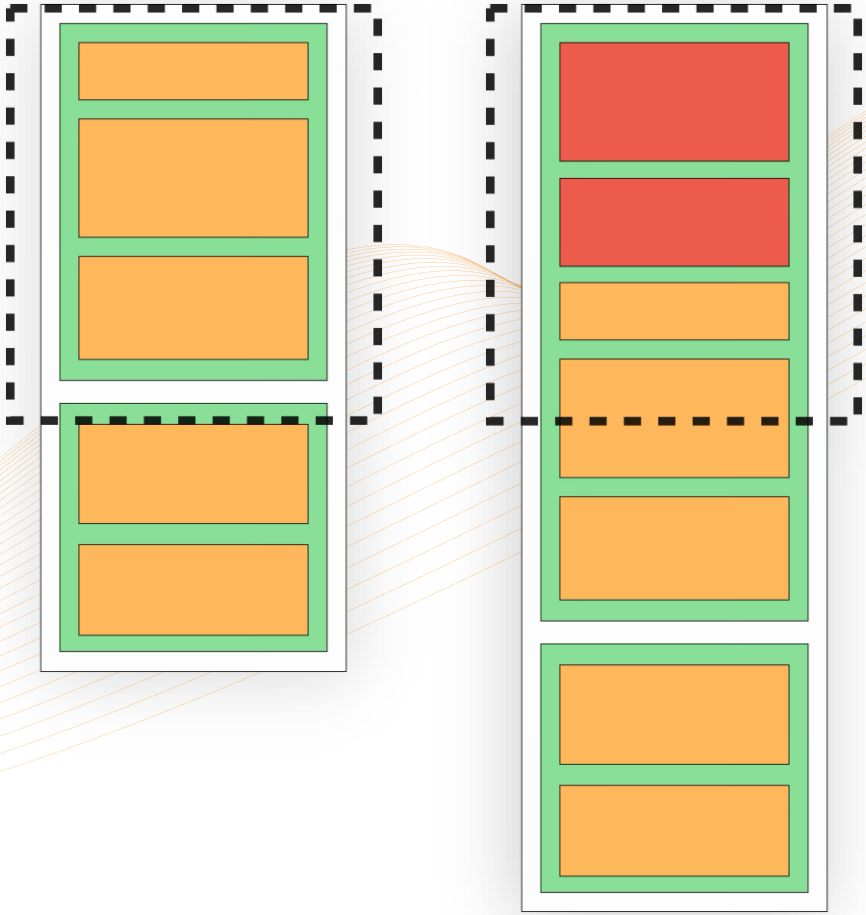
1С-Коннект
Программист iOS

- **Делаем копию макета**



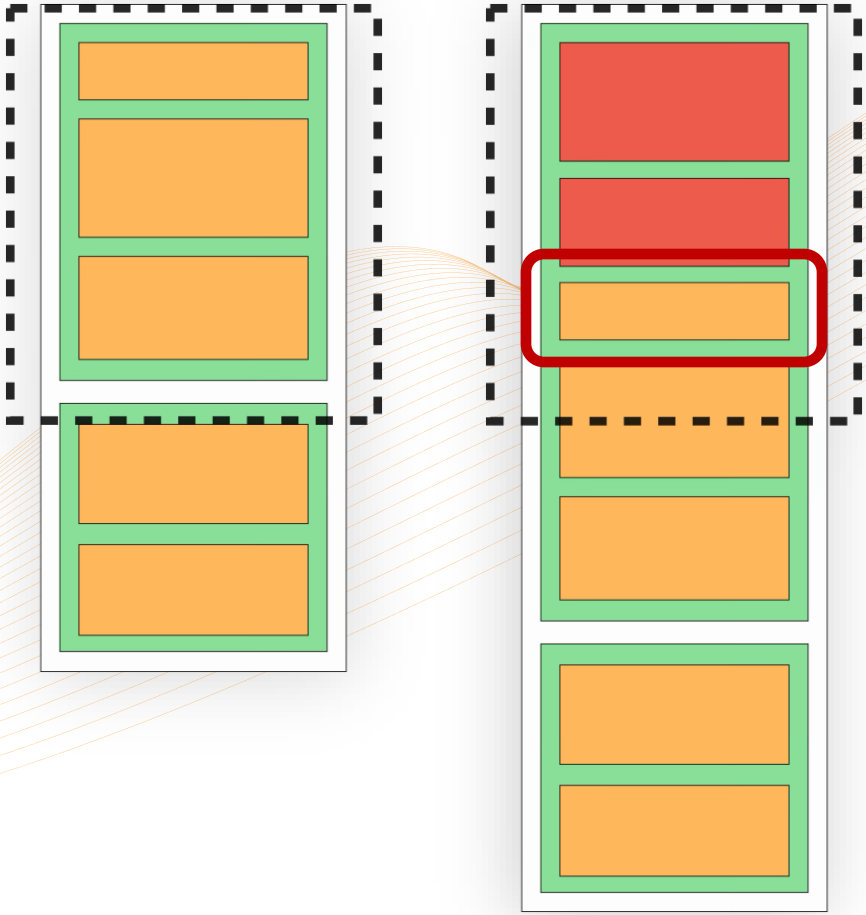
1С-Коннект
Программист iOS

- Делаем копию макета
- Пересчитываем макет



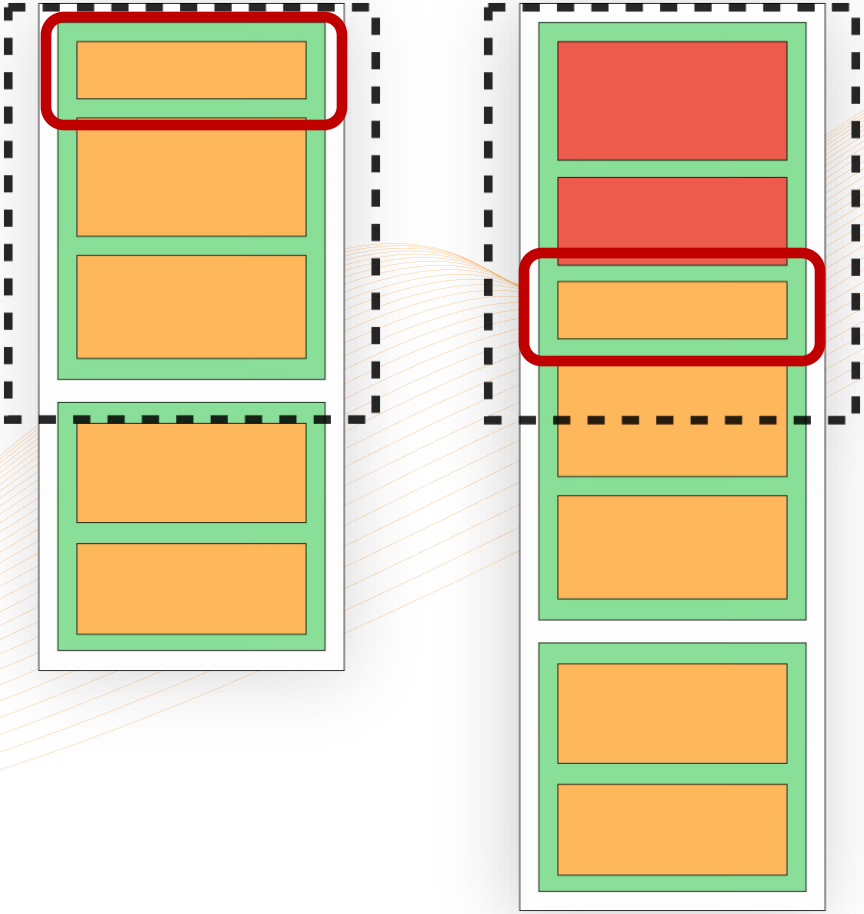
1С-Коннект
Программист iOS

- Делаем копию макета
- Пересчитываем макет
- Ищем ячейку которая не была изменена



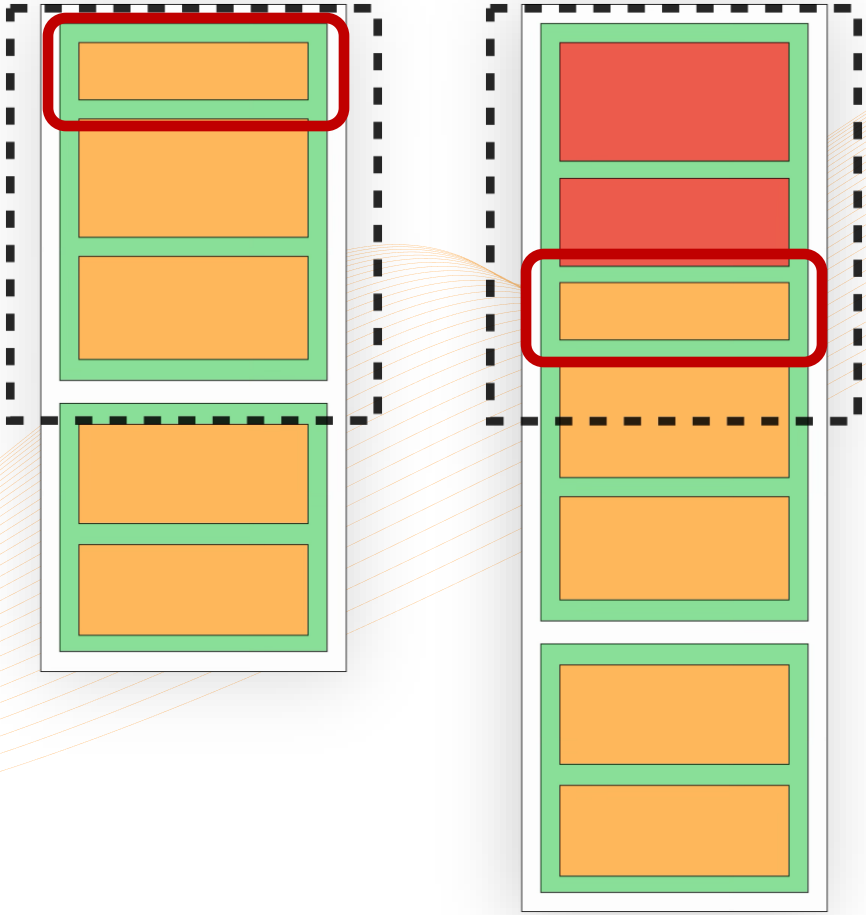
1С-Коннект
Программист iOS

- Делаем копию макета
- Пересчитываем макет
- Ищем ячейку которая не была изменена
- В копии макета находим эту ячейку



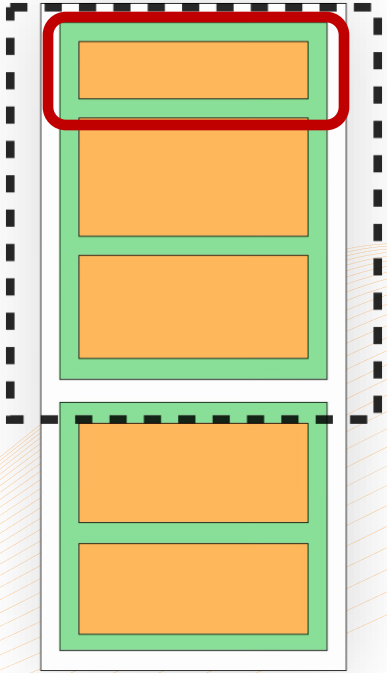
1С-Коннект
Программист iOS

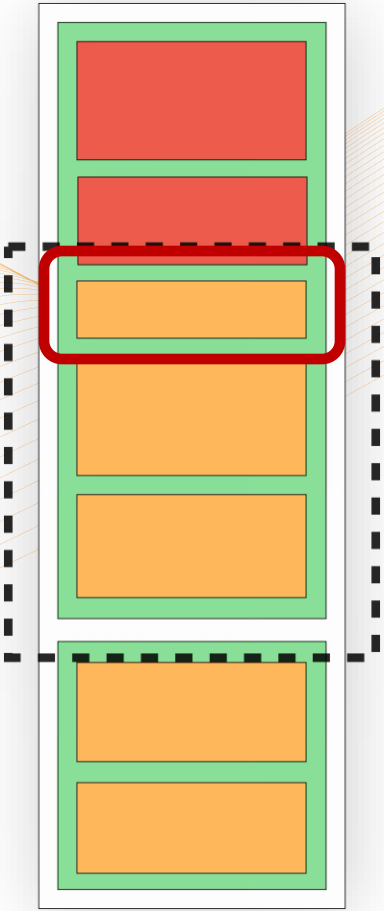
- Делаем копию макета
- Пересчитываем макет
- Ищем ячейку которая не была изменена
- В копии макета находим эту ячейку
- Определяем смещение видимой области относительно этой ячейки



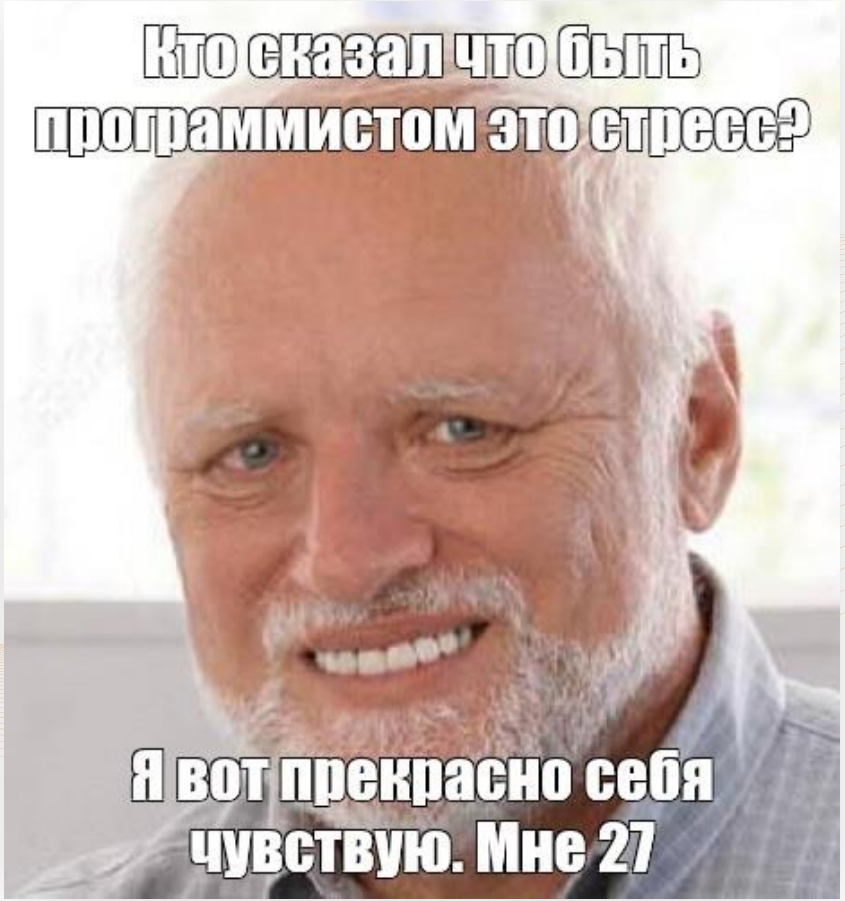
1С-Коннект
Программист iOS

- Делаем копию макета
- Пересчитываем макет
- Ищем ячейку которая не была изменена
- В копии макета находим эту ячейку
- Определяем смещение видимой области относительно этой ячейки
- В обновленном макете выставляем такое же смещение

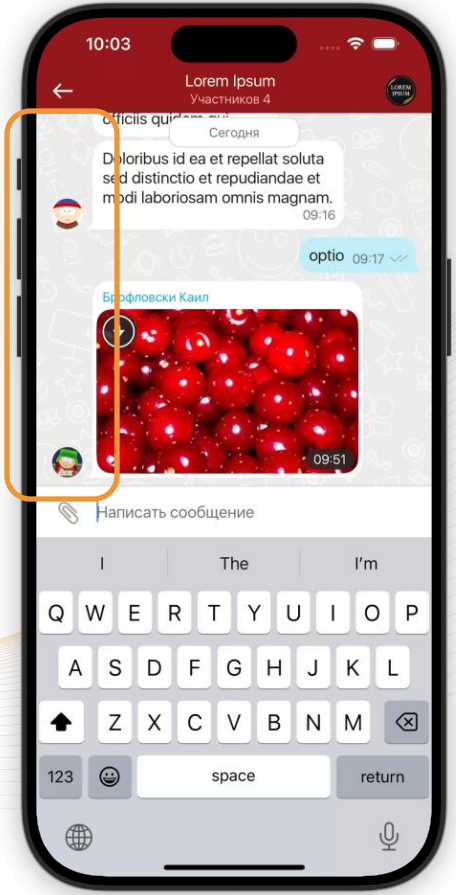




1С-Коннект
Программист iOS



1С-Коннект
Программист iOS



1С-Коннект
Программист iOS

```swift
class CustomLayoutAttributes: UICollectionViewLayoutAttributes {
    var frameInSection: CGRect = .zero
    var itemConfig: CellConfiguration = .empty
    var avatarAttributes: UICollectionViewLayoutAttributes? = nil

    override func copy(with zone: NSZone? = nil) -> Any { ••• }

    func typedCopy() -> CustomLayoutAttributes { ••• }
}
```

1С-Коннект
Программист iOS

```
class CustomLayoutAttributes: UICollectionViewLayoutAttributes {
    var frameInSection: CGRect = .zero
    var itemConfig: CellConfiguration = .empty
    var avatarAttributes: UICollectionViewLayoutAttributes? = nil

    override func copy(with zone: NSZone? = nil) -> Any { ••• }

    func typedCopy() -> CustomLayoutAttributes { ••• }
}
```

1С-Коннект
Программист iOS

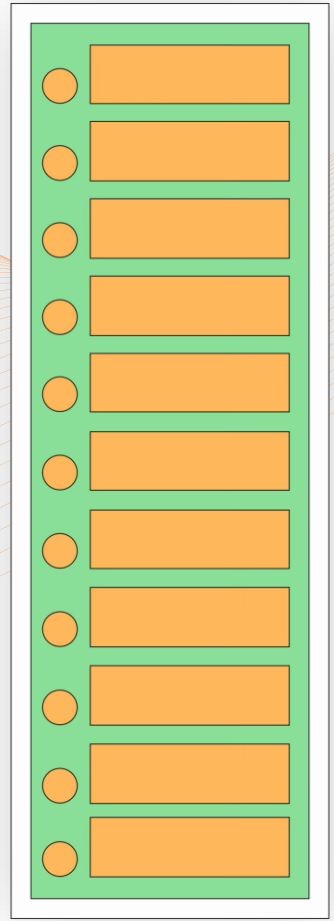
```
struct CellConfiguration {
    let itemSize: CGSize
    let authorId: String?
    let isIncoming: Bool?
    let incomingBubbleInsets: UIEdgeInsets
    let outgoingBubbleInsets: UIEdgeInsets
}
```

1С-Коннект
Программист iOS

layoutAttributesForElements(in: rect)

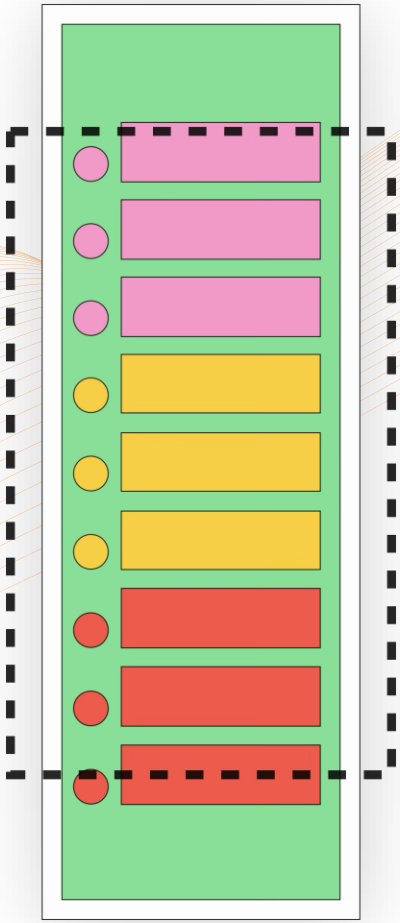
1С-Коннект
Программист iOS

- **В макете рассчитаны аватарки для каждой ячейки**



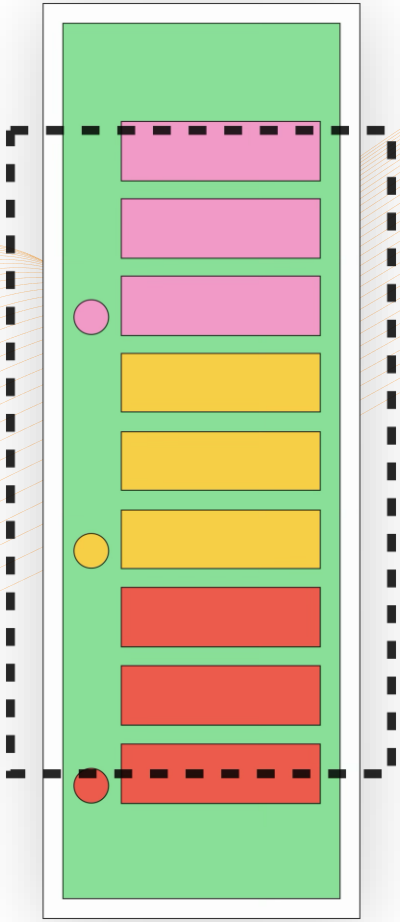
1С-Коннект
Программист iOS

- **В макете рассчитаны аватарки для каждой ячейки**
- **В видимой области находим последовательности по автору**



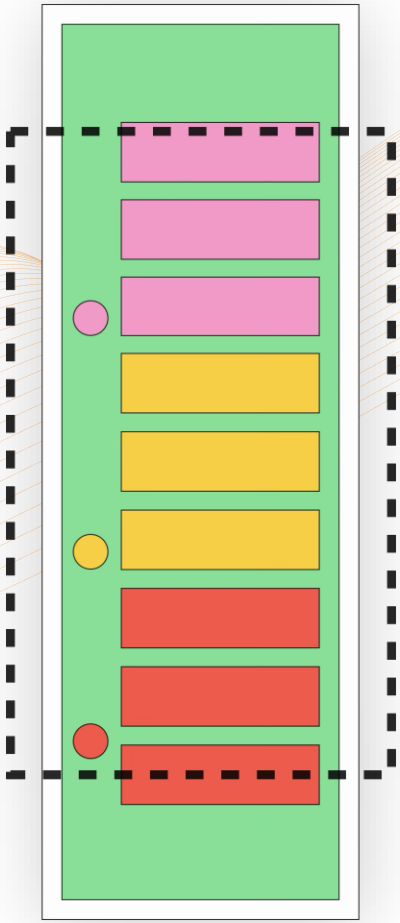
1С-Коннект
Программист iOS

- В макете рассчитаны аватарки для каждой ячейки
- В видимой области находим последовательности по автору
- Берем аватарку из нижнего сообщения последовательности



1С-Коннект
Программист iOS

- В макете рассчитаны аватарки для каждой ячейки
- В видимой области находим последовательности по автору
- Берем аватарку из нижнего сообщения последовательности
- Пересчитываем положение аваратарки из нижней последовательности



1С-Коннект
Программист iOS

1С-Коннект
Программист iOS

На этом всё?

1С-Коннект
Программист iOS

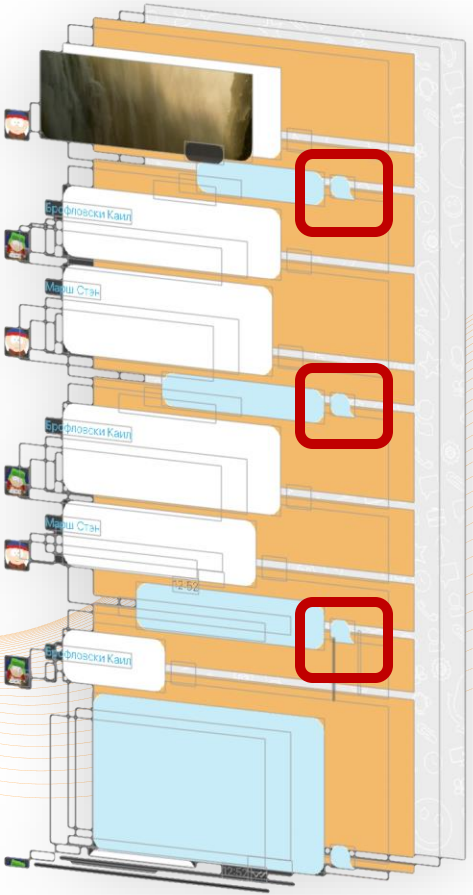
Нет...

1С-Коннект
Программист iOS

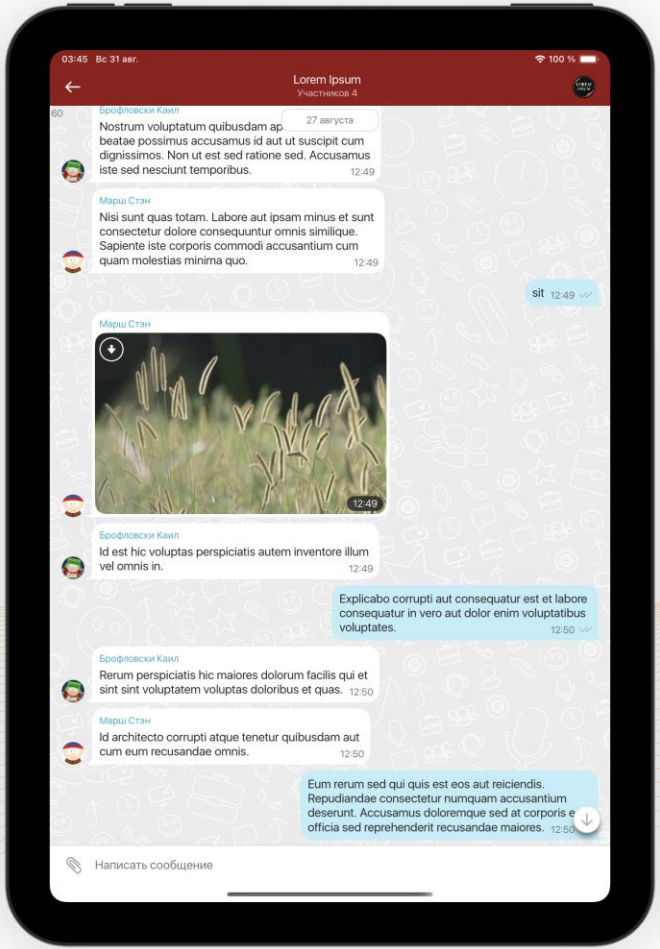
Остались пузыри и хвосты

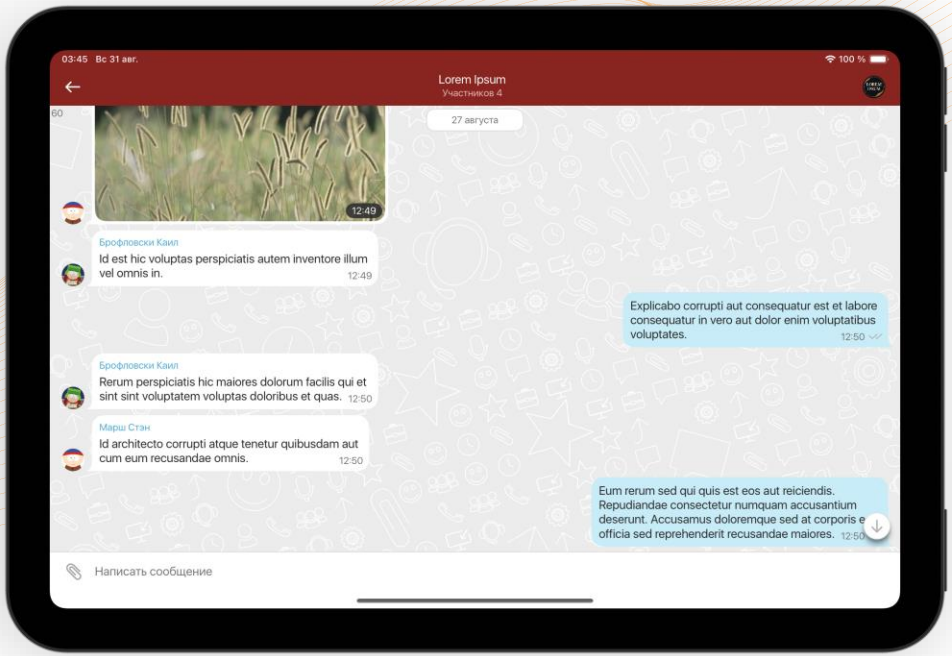
1С-Коннект
Программист iOS

1С-Коннект
Программист iOS

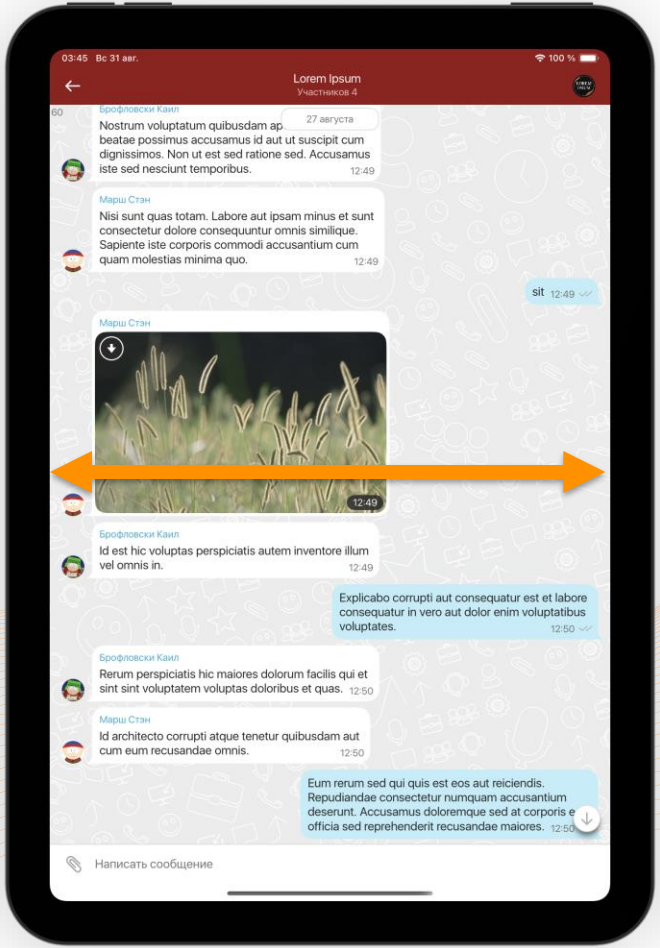


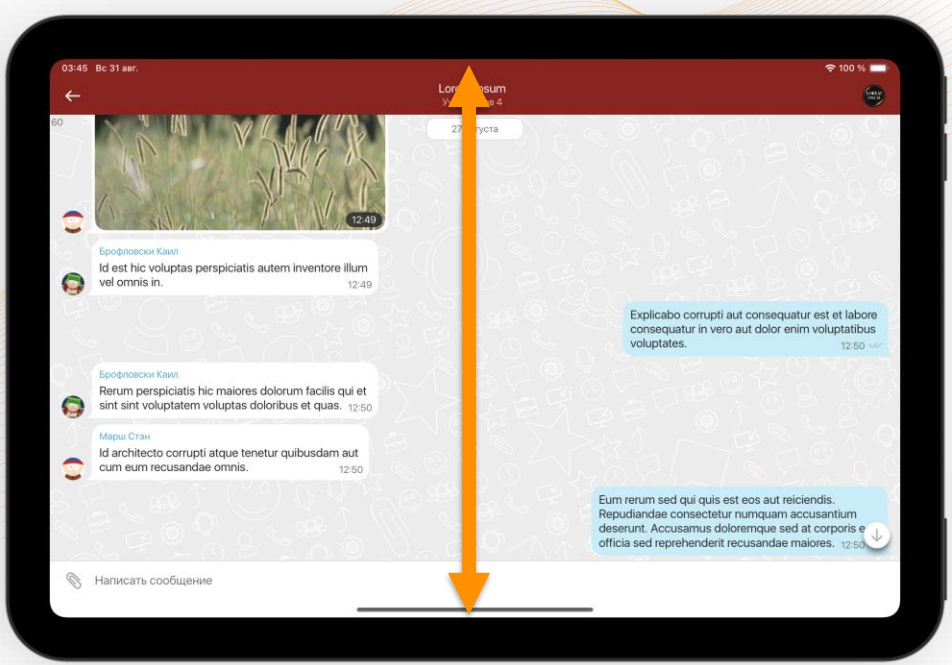
1С-Коннект
Программист iOS



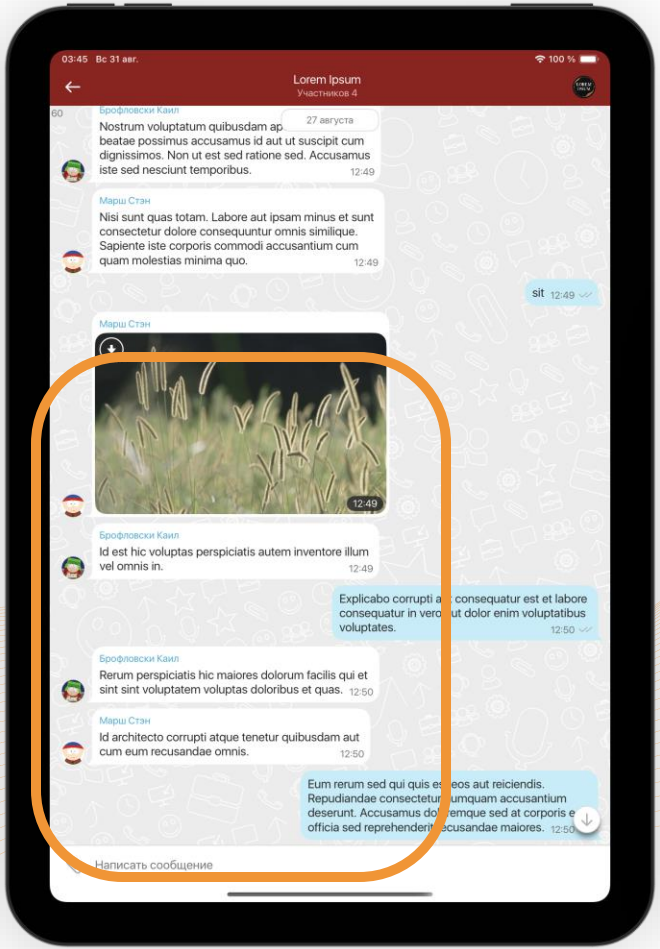


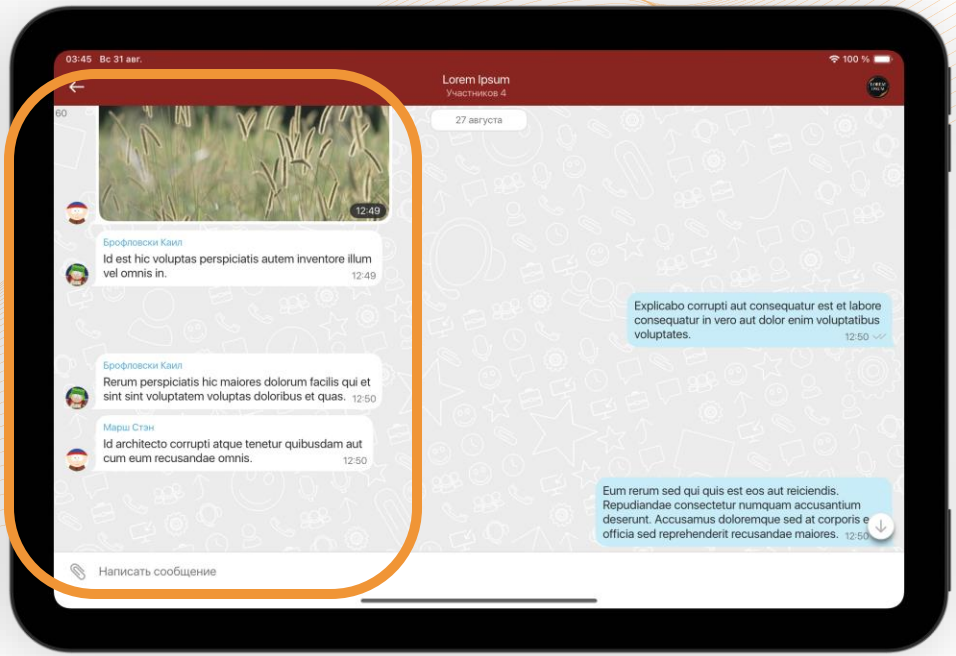
1С-Коннект
Программист iOS



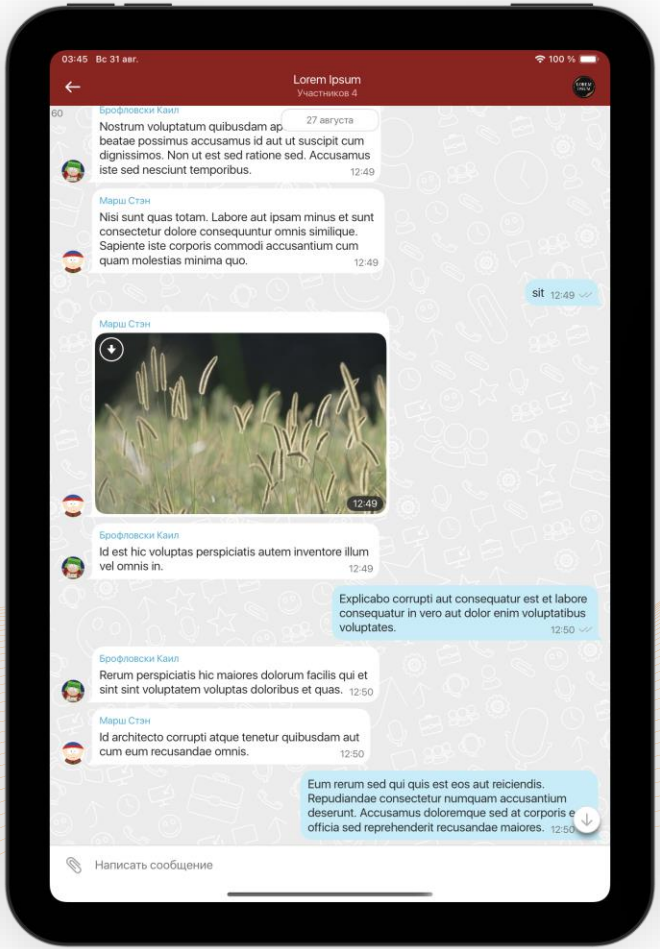


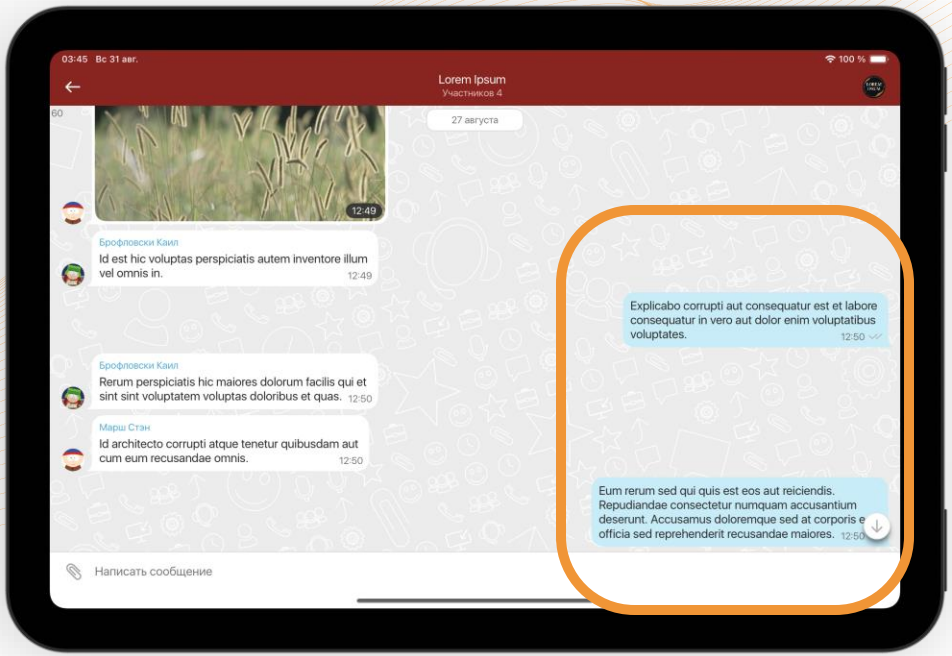
1С-Коннект
Программист iOS





1С-Коннект
Программист iOS





1С-Коннект
Программист iOS

Выводы и советы

1С-Коннект
Программист iOS

Не делайте кастомный лейаут

1С-Коннект
Программист iOS

Закладывайте время на это

1С-Коннект
Программист iOS

Возьмите паузу, и сделайте небольшой релиз

1С-Коннект
Программист iOS

На этом всё

1С-Коннект
Программист iOS

Вопросы?

Материалы к докладу,
ссылки на источники

Оценка доклада