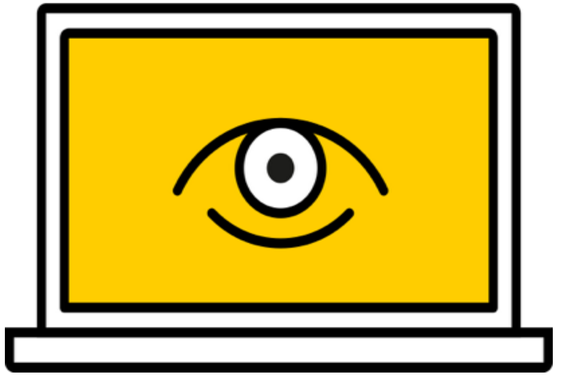
Яндекс

Лекция 3.0

View Layer

Слой представления

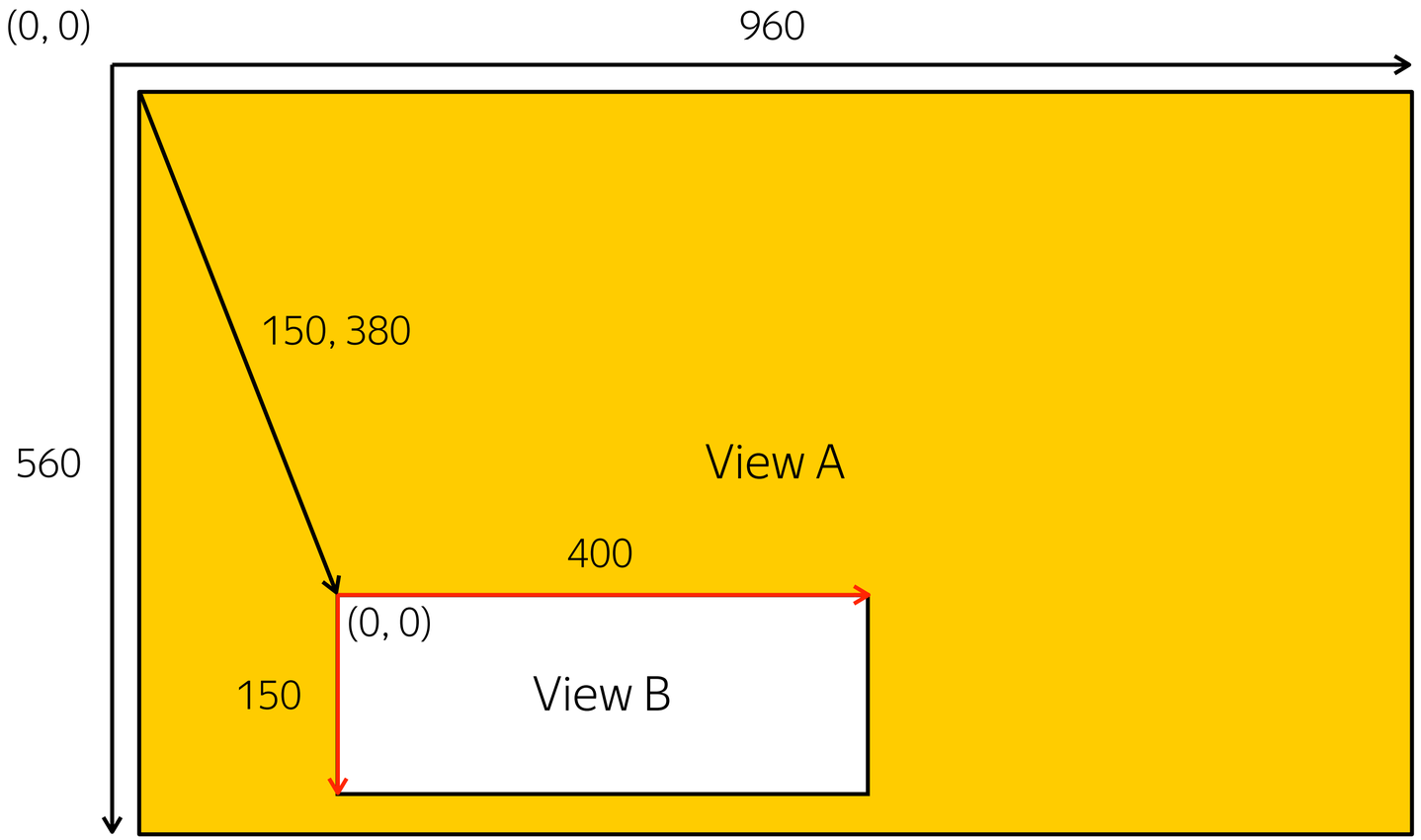


О чем будем сегодня говорить

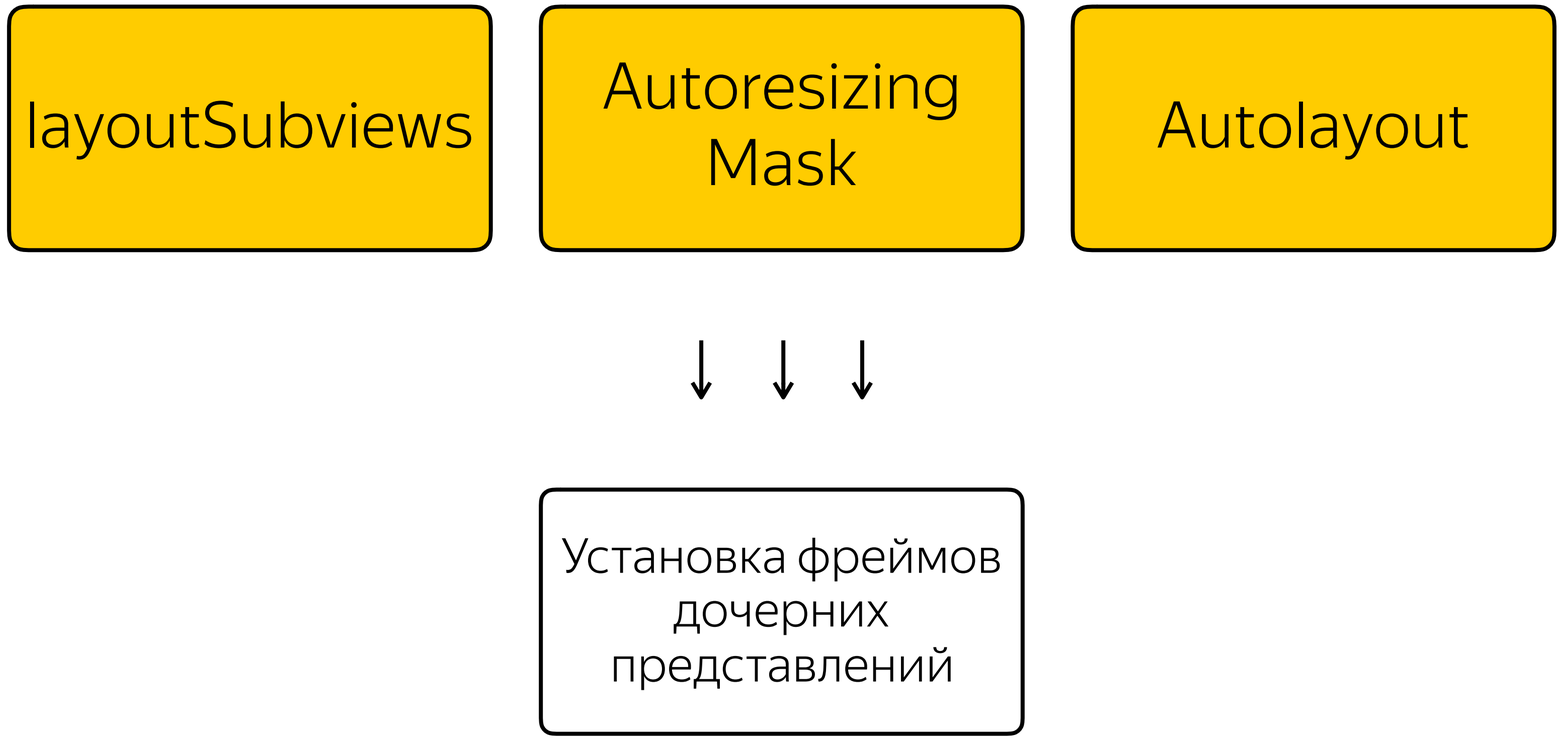
1. UIView
2. Autolayout
3. Interface Builder
4. CoreGraphics
5. UIViewController Lifecycle
6. Обработка событий и Gestures
7. Анимации и переходы

- UIView -> UIResponder -> NSObject
- CALayer -> NSObject

Frame != Bounds



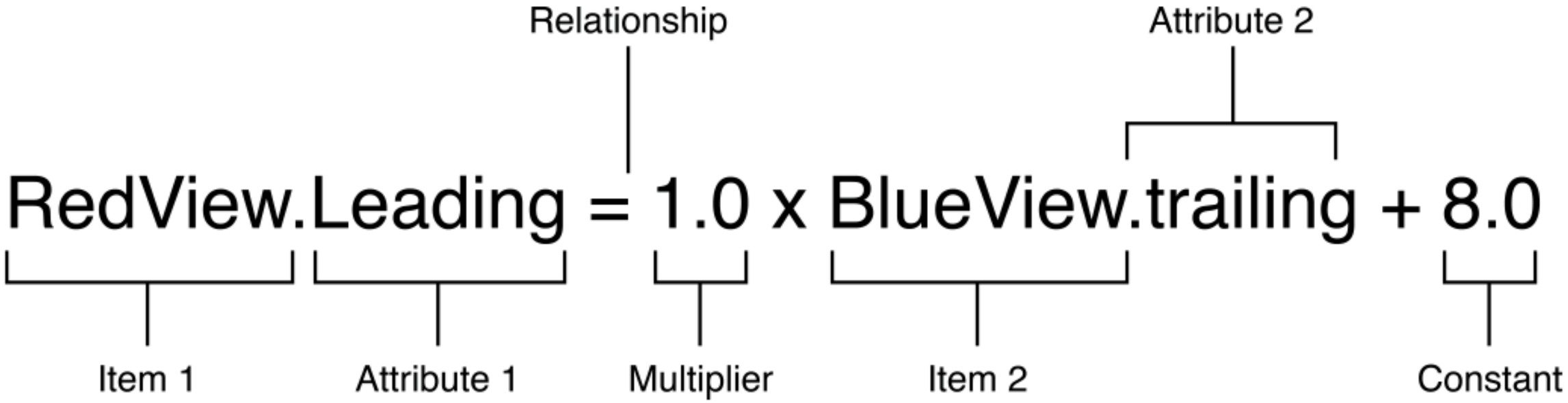
View A frame:
origin: (0, 0)
size: 960 x 560
View A bounds:
origin: (0, 0)
size: 960 x 560
View B frame:
origin: (150, 380)
size: 400 x 150
View B bounds:
origin: (0, 0)
size: 400 x 150


layoutSubviews
Autoresizing Mask
Autolayout
Установка фреймов дочерних представлений

It's a DEMO time

Пример верстки фреймами

NSLayoutConstraint



It's a DEMO time

Пример верстки constraint'ами

intrinsicContentSize

Content Hugging + Priority

- proposedSize > intrinsicContentSize
- "Hug" - англ. обнимать
- ContentHuggingPriority - первым увеличивается тот, у кого меньше этот приоритет

Compression Resistance + Priority

〉 proposedSize < intrinsicContentSize

〉 "Resist" - англ. сопротивляться

〉 CompressionResistancePriority - первым сжимается тот, у кого меньше приоритет

Content Hugging vs Compression Resistance

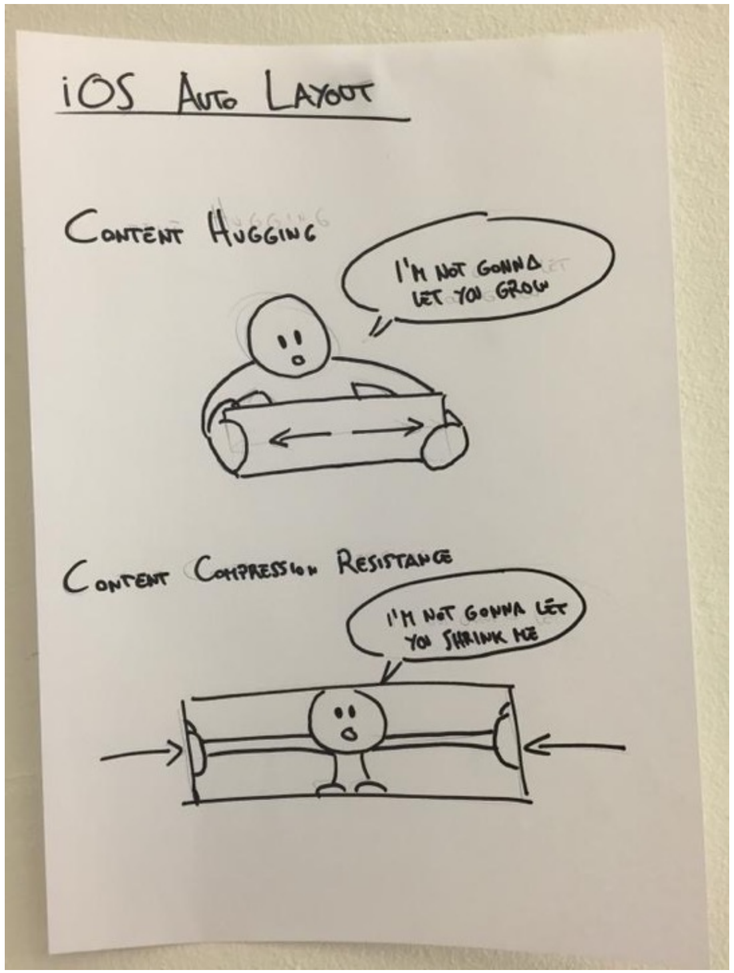


Image Credit: krakendev.io

Autolayout. Рекомендации

- Autolayout - полезный инструмент
- Используйте минимальное количество правил
- Минимизируйте количество неочевидных констант
- По возможности не решайте проблемы при помощи приоритетов
- **Используйте leading/trailing вместо left/right**
- Используйте контейнеры для более очевидной верстки
- Стремитесь к адаптивной верстке

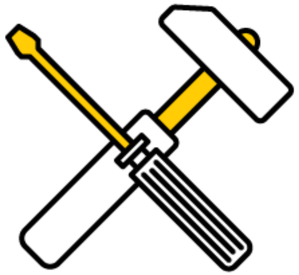
Autolayout. Проблемы

〉 Работает дольше, чем верстка "руками" на фреймах (в некоторых приложениях может быть критично)

〉 Требует layoutIfNeeded в анимационных блоках ⚠️

Interface Builder

- Все что можно сделать при помощи IB можно сделать без него. Обратное не верно.
- Два вида файлов: .xib и .storyboard
- Степень использования крайне отличается в разных командах/проектах



It's a DEMO time

Interface Builder + Custom Views

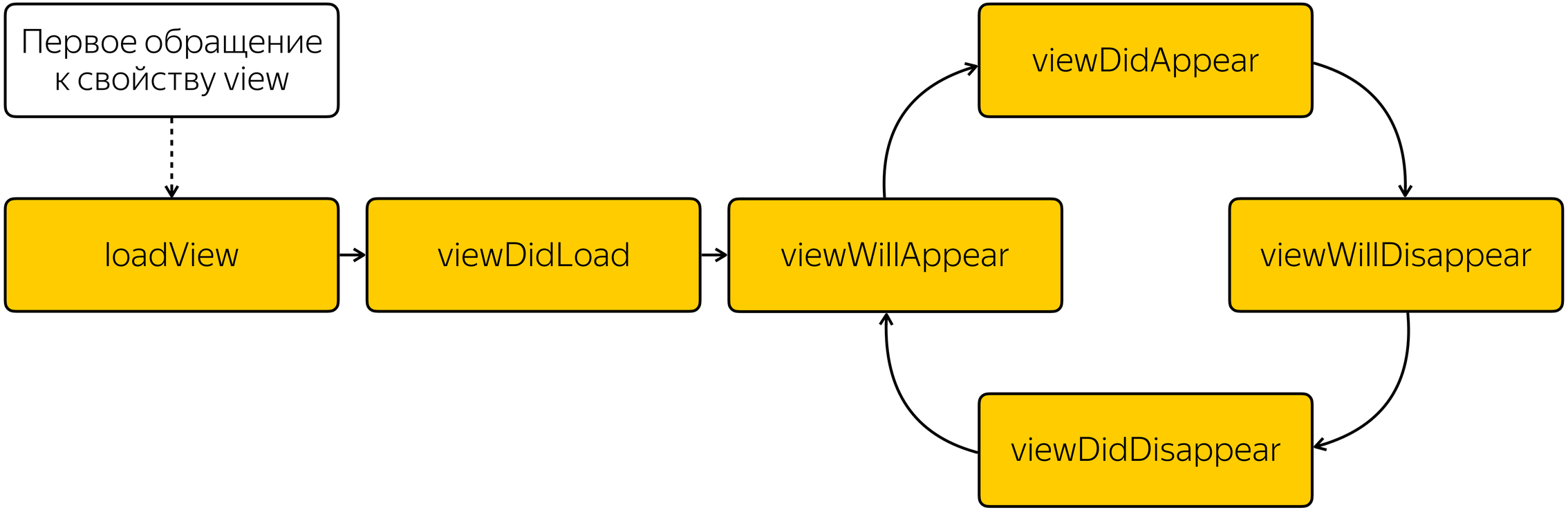
Core Graphics

- Рисование при помощи графических примитивов
- Метод drawRect у UIView
- Контекст имеет состояние
- Перерисовка представления вызывается через setNeedsDisplay

It's a DEMO time

Рисуем флаг РФ

UIViewController Lifecycle



It's a DEMO time

UIStoryboardSegue

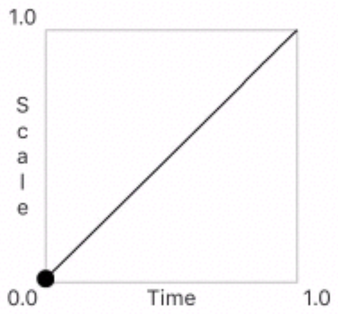
Обработка событий и жесты

- UIResponder
  - touchesBegan
  - touchesMoved
  - touchesEnded
  - touchesCancelled
- UIGestureRecognizer
- target-action

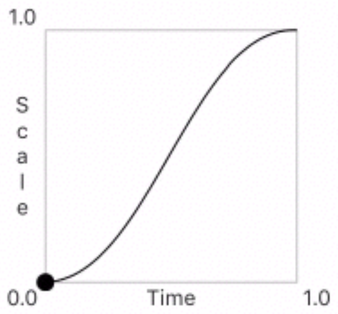
It's a DEMO time

target + action

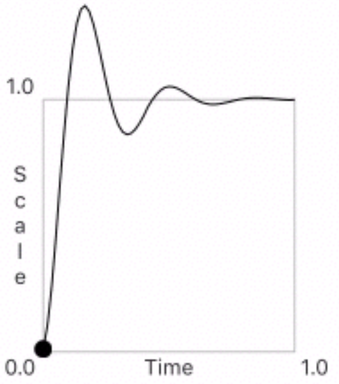
Анимации. Таймингговые функции



Linear

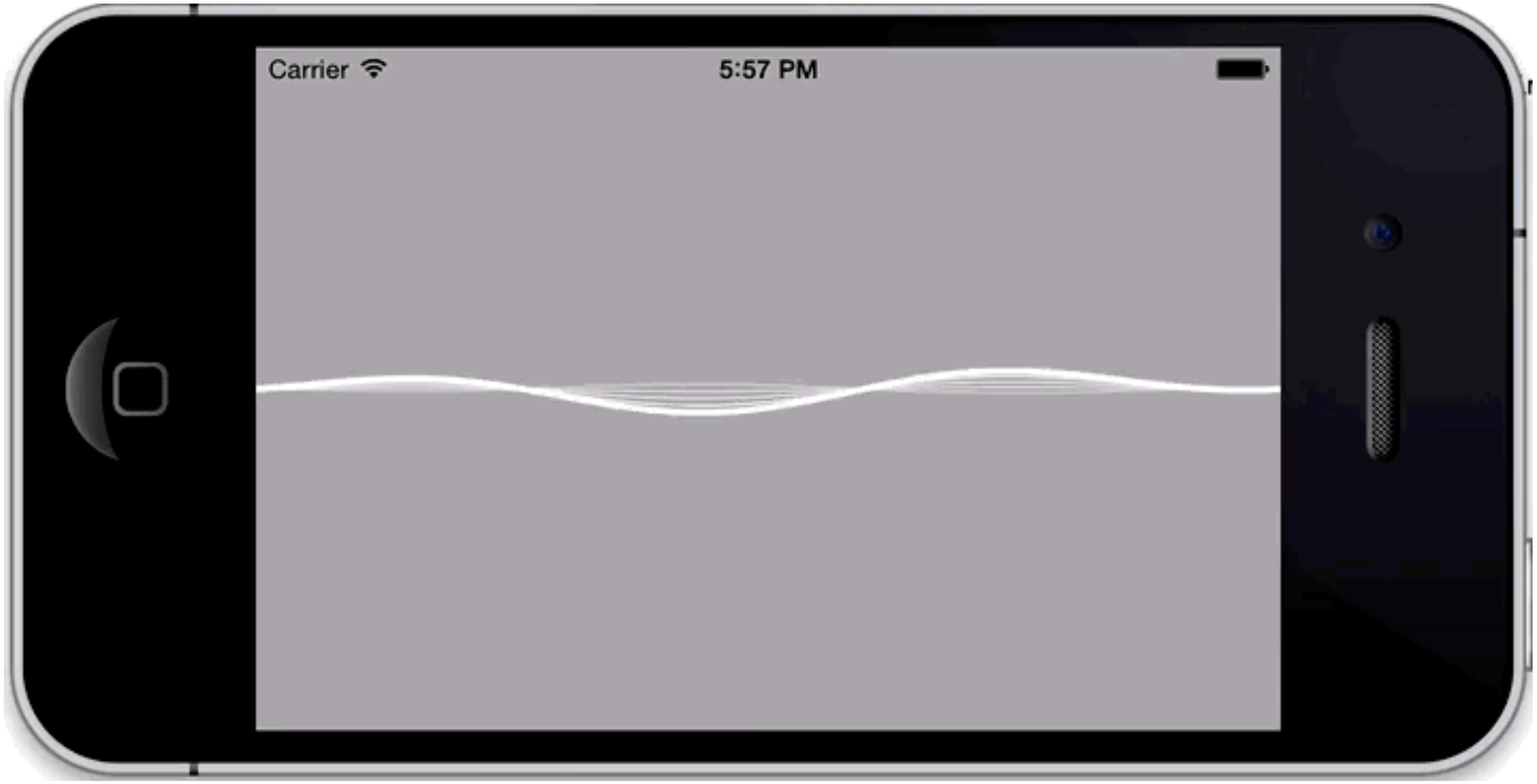


EaseInOut



Spring

Анимации. CADisplayLink



Анимации. Пара замечаний

- **Если анимируете констрейнты, то вызывайте layoutIfNeeded**
- Если анимируете layer, то учитывайте что есть model и presentation layer
- Используйте keyframes для сложных анимаций

Вопросы ?

Задача

Общие требования

〉 Сверстать экран редактирования заметки. Можно верстать на UIViewController, а можно создать отдельный класс наследник UIView.

〉 Верстка должна быть адаптивной: на разных рахмерах экрана и в разных ориентациях должно выглядеть аккуратно

Усложнение *

〉 Контент должен скроллироваться

〉 DatePicker должен появляться/исчезать при смене состояния свитчера

〉 Высота поля для ввода текста заметки должна динамически меняться в зависимости от содержимого (с некоторым минимальным размером)

〉 При появлении клавиатуры на экране контент должен оставаться просматриваемым (текст не попадает "под" клавиатуру)

〉 Выбор цвета должен быть реализован в виде цветных квадратиков с черной рамочкой. Текущий выбранный цвет должен помечаться галочкой. Галочка должна быть отрисована с помощью CoreGraphics

Усложнение **

- На основном экране редактирования заметки в секции выбора цвета должен быть дополнительный квадратик для выбора кастомного цвета. Изначально должен выглядеть как палитра цветов
- По долгому нажатию на доп. квадратик должен открываться экран с компонентом ColorPicker
- ColorPicker обязательно должен быть выполнен в виде отчуждаемого компонента (то есть класс ColorPickerView)
- Выбор цвета осуществляется путем перемещения пальца по палитре
- Элемент указывающий на текущий цвет в палитре в центре должен быть залит в текущий цвет
- Элемент отображающий текущий цвет (в левом верхнем углу) должен иметь скругление углов (радиус выбираете сами)
- Выбранный цет должен сохраняться при повторном заходе на экран ColorPicker

Материалы

- Основные требования - https://yadi.sk/i/7wNgzGSx3HR4D2
- С усложнением (*) - https://yadi.sk/i/CKNjpMo23HR4D8
- С усложнением (**) - https://yadi.sk/i/G2KDl9G83HR4DY
- ColorPicker - https://yadi.sk/i/w5FR44FO3HR4Dg

Контакты

Малых Денис

iOS Developer

 mrdekk@yandex.ru

 @mrdekk

 mrdekk

Спасибо за внимание