

IOS

UI-элементы в UIKit

Документация по UI-элементам: название, описание, возможности.
Описывается фреймворк UIKit, подходит для SwiftUI лишь частично.

UIButton

Обычная кнопка. Можно указывать текст, шрифты, цвет самой кнопки, цвет краев, стили для разных состояний: Enabled, Disabled, Highlighted. Можно не указывать текст и цвет, тогда кнопка будет невидимой. Можно установить картинку вместо текста и цвета.

UILabel

Текст, он и в Африке текст. Можно изменять текст, фон, шрифты. Можно использовать Attributed Text.

UITextField

Поле для ввода текста, можно изменять текст в нем, плейсхолдер, цвета, границы, стили краев. Можно сделать защищенным(звездочки вместо текста). Можно указывать максимальную длину и тип клавиатуры. При вводе текст идет в одну строку и «прокручивается» по мере ввода.

UITextView

Похоже на текстфилд, но нет плейсхолдеров и текст пишется в несколько строк. Текст можно скроллить.

UIScrollView

Контейнер для контента, который не помещается на экран. Можно делать вертикальный и горизонтальный скролл. Эффект «Пружинки» отключается.

UITableView

Таблица с ячейками. Можно использовать стандартные ячейки, можно делать кастомные. Можно отключать пружинку. СТРОГО ВЕРТИКАЛЬНЫЙ СКРОЛЛ. Ячейки всегда имеют одинаковую ширину - ширину самой таблицы.

UICollectionView

Похоже на таблицу, но с некоторыми отличиями. Можно делать не только вертикальный, но и горизонтальный скролл(одно из двух), разную ширину ячеек. В остальном та же таблица.

UIStackView

Контейнер, можно засовывать любые вьюхи. Можно выбирать тип стека - вертикальный или горизонтальный, распределение контента, отступы, центрирование.

UIPickerView

Вьюха в виде барабана, который можно прокручивать для выбора варианта(Например, выбор времени на будильнике). В качестве значений используется массив. Есть разновидность для выбора даты или времени, в таком случае нужно указать только максимальную и минимальную дату.

UIProgressView

Прогресс бар. Можно изменять цвета, уровень заполнения(с анимацией и без).

UIActivityIndicatorView

Классический лоадер в виде круга из палочек, можно изменять цвет.

UIImageView

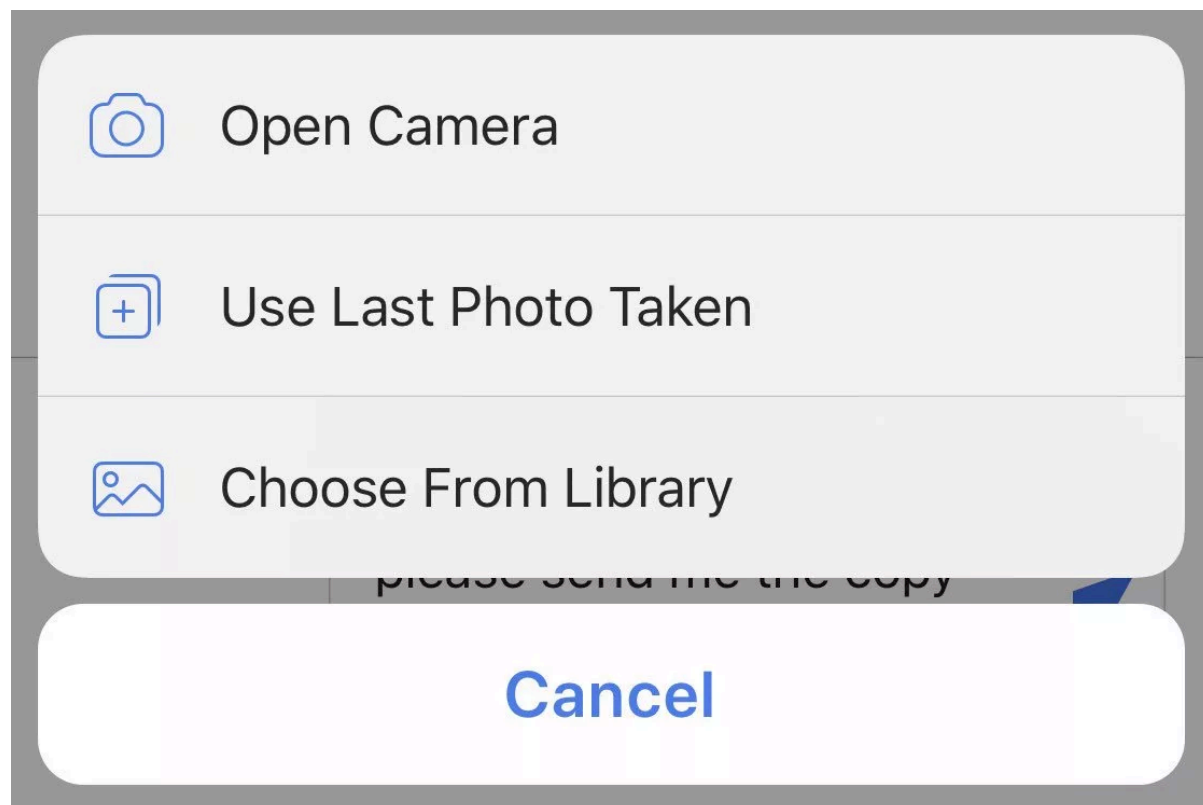
Контейнер для картинки. Можно изменять тип размещения контента(растягивать для заполнения, центрировать, отдалять картинку для умиротворения и т.д.)

UITabBar

Нижний бар с кнопками-вкладками. При переключении на другую вкладку состояние текущей сохраняется и отображается на том же месте при возврате.

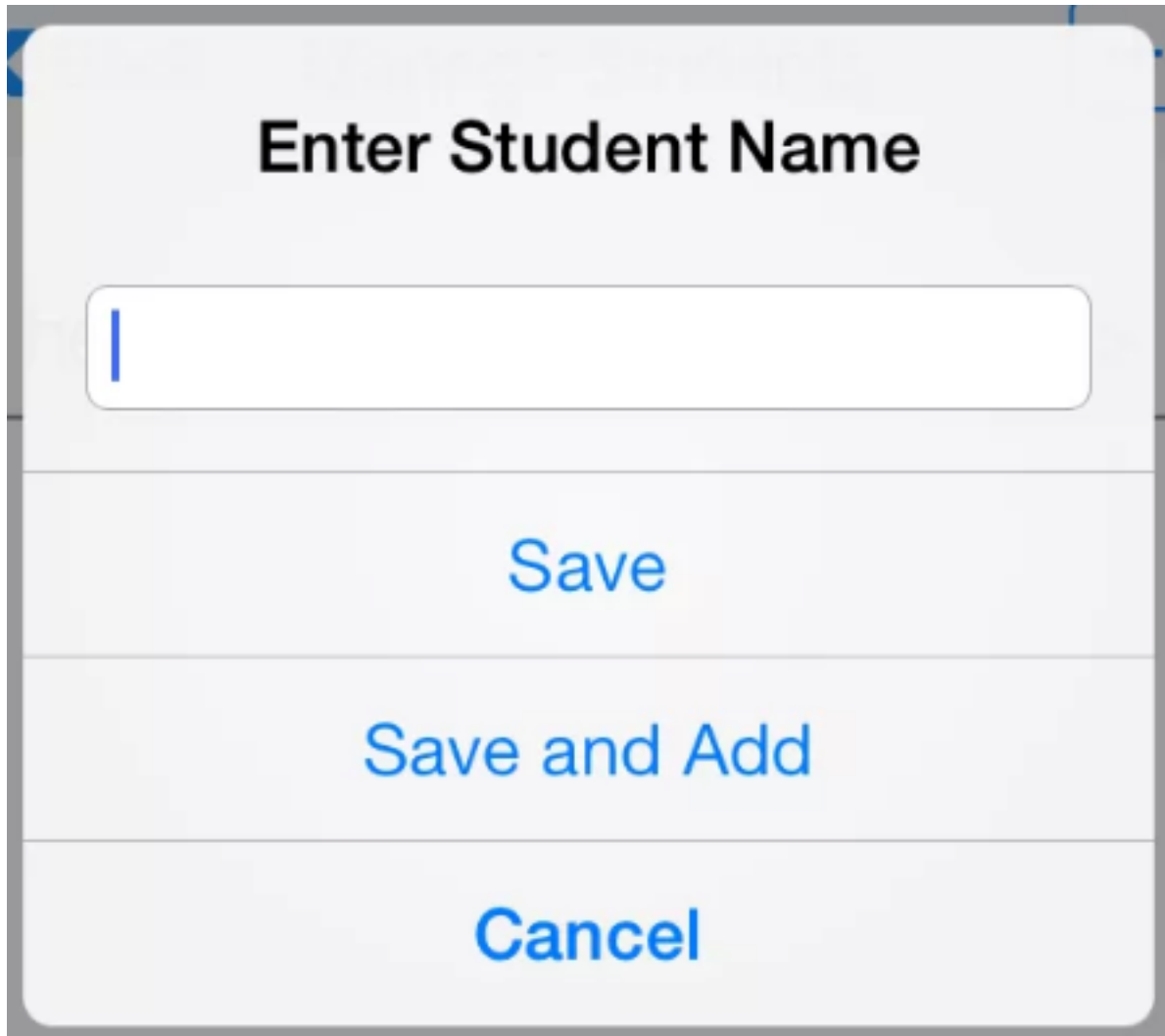
UIActionSheet

В IOS нет боттом щитов, но есть такие вьюхи. Это нативные вьюшки, рекомендуется по возможности использовать их.



UIAlertView

Нативная системная модалка. Можно добавлять в нее различные элементы - кнопки, текстфилды, лейблы и т.д.



UISearchBar

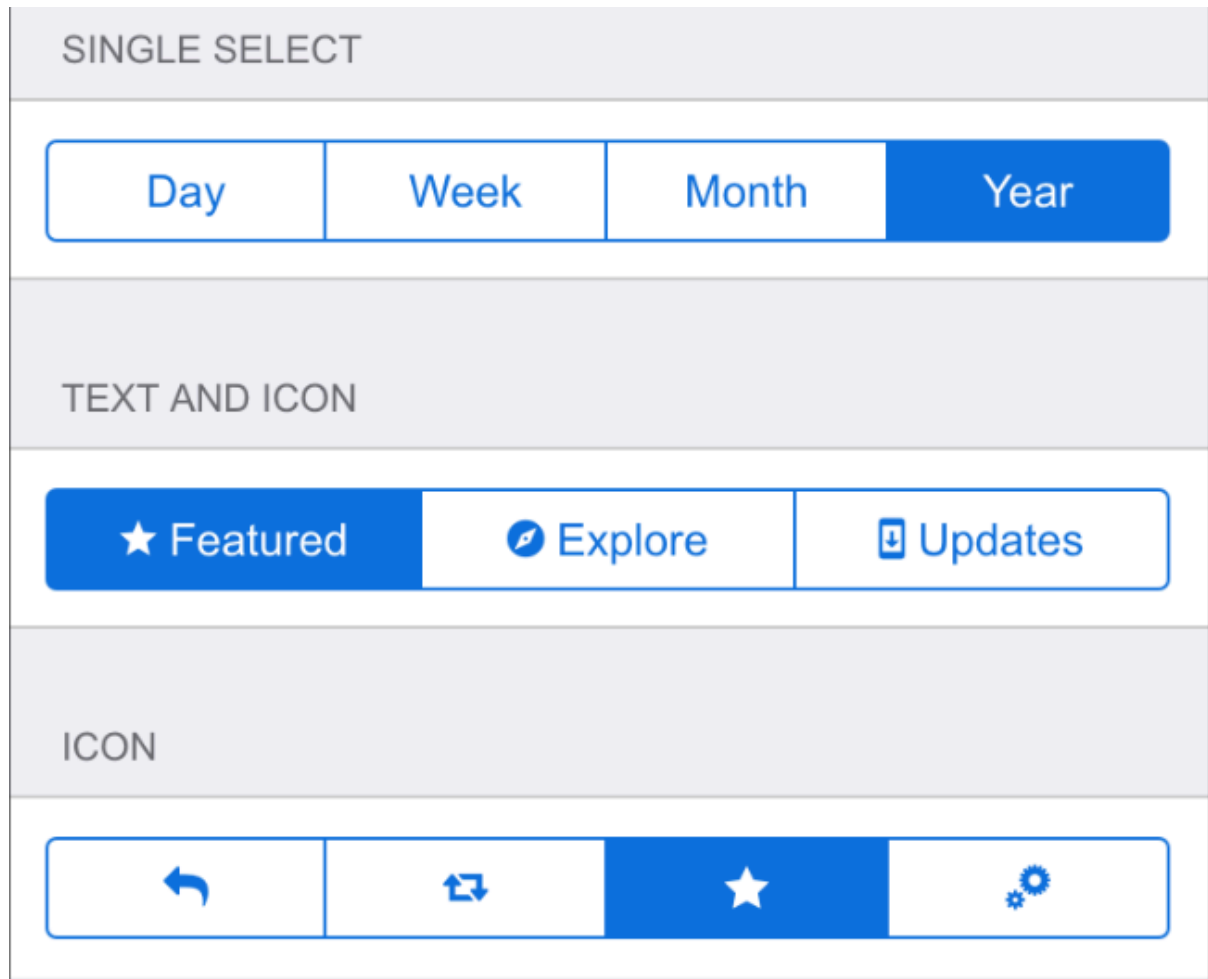
Текстфилд с небольшими надстройками для реализации поиска

UIWebView

Вебвью. Урезанный браузер внутри приложения. Часть js кода не работает

UISegmentedControl

Сегмент контроллер, можно указывать разное количество элементов, текст элементов, цвета.



UISlider

Ползунок. Можно изменять цвет заполнения, цвет самой точки контрола, цвет подложки. Точки на нем расставлять нельзя, это приходится делать вручную через другие элементы(Например, накладывая сверху UIStackView с UILabel'ами).

UISwitch

Стандартный свитч. Можно менять цвет активного и неактивного состояния, цвет контрола. НИКАКИХ ТРЕХПОЗИЦИОННЫХ СВИТЧЕЙ, НЕ ВЕРЬ ИГОРЮ.

UIGestures

Жесты, которые можно вешать на вьюхи и обрабатывать.

- UILongPressGestureRecognizer - распознавание долгого нажатия
- UIPanGestureRecognizer - распознавание жеста «нажать и потянуть»
- UIPinchGestureRecognizer - распознавание жеста зуммирования
- UIRotationGestureRecognizer - распознавание жеста кручения двумя пальцами
- UISwipeGestureRecognizer - распознавание свайпа
- UITapGestureRecognizer - распознавание тапа