|  |  |
| --- | --- |
|  | **Кислюк Игорь Витальевич**  Год рождения: 1996  Факультет ИКТ, кафедра ПС, группа K3421  Специальность: Интеллектуальные информационные системы  e-mail: igorkislyuk@icloud.com |

ОБЗОР И АНАЛИЗ БИБЛИОТЕК ДЛЯ АНИМИРОВАНИЯ

Бакалавр И. В. Кислюк

Научный руководитель – старший преподаватель Осетрова Ирина Станиславовна

В данной статье проведен обзор основных библиотек для анимации компонентов вида модели MVC. Было проведено сравнение их ключевых особенностей, недостатков и достоинств, на основании чего были сделаны выводы о удобстве и необходимости использовании данных библиотек в проектах

Ключевые слова: анимации, библиотеки для анимирования, анализ библиотек для анимирования

Анимации является неотъемлемой частью операционной системы[2] компании Apple iOS, однако их реализация может стоить некоторых усилий, поэтому стоит рассмотреть основные библиотеки, которые используются для упрощения реализации анимаций всех типов в iOS.

Фреймворк для совершения цепочных преобразований "JHChainableAnimations". Написан на языке Objective-C, ограничения на использование минимальные, поэтому спокойно интегрируется со всеми актуальными версиями операционной системы iOS. Позволяет быстро и доступно связать несколько различных анимаций, использую цепочку вызовов блоков [1].

Например, для изменения обычного компонента простым образом, таким как смещения, повороты и масштабирование, необходимо написать несколько строчек кода, однако при использовании библиотеки этот эффект может быть достигнут всего лишь с одной строчкой анимации. Что же касается вложенных анимаций, то преимущества становятся более явными. Имеет однозначный выигрыш в удобстве и простоте написания кода различной сложности, даже при расположении анимации на несколько уровней, где под уровнем анимации понимается величина задержки до воспроизведения анимации, то есть следующая анимация с определенным смещением относительно предыдущей анимации. Таким образом, если вторая анимация начнется строго после завершения первой анимации - это будет анимация второго уровня по отношению к первой.

"JHChainableAnimations" позволяет использовать различные функции для управлением временем анимации, которые значительно расширяют стандартный набор. Функции управления временем анимации - функции которые определят скорость изменения свойств компонента UIView с течением времени. Самыми популярными являются Linear, EasyIn и EasyOut, которые присутствуют в стандартной библиотеке анимации CoreAnimation[4], однако фреймворк предоставляет такие как easeInBounce, easeOutBounce и easeInOutBounce. Дополнительно предлагаются изменение точки опоры компонента, ручное изменение матрицы преобразования и другие низкоуровневые вещи для компонентов.

Немалым преимуществом можно отметить документацию к использованию и руководство по установке.

С другой стороны, нет возможности совершать анимированные переходы между различными видами контроллеров, что является определенным недостатком.

Фреймворк для выполнения показа модальных видов[3] контроллеров "Presentr". Написан на языке Swift 3. Поддерживает прошлые версии языка 2.2 и 2.3. Ограничение на версию операционной системы - с iOS 7. Главная идея заключается в том, что для того, чтобы вид нового контроллера не закрывал весь экран необходимо использовать новый программный интерфейс "Custom View Controller Presentation API", который был показан компанией Apple на презентации iOS 7.

Принцип работы: необходимо создать объект презентации типа Presentr, что уже является недостатком, поскольку необходимо постоянно следить за жизненным циклом этого объекта. После чего вызывается метод для анимационного показа вида нового контроллера, в который необходимо передать больше параметров, включая этот самый объект. Все остальные необходимые действия выполнит фреймворк.

Главным достоинством является удобство установки вида и типа анимации, всего лишь посредством изменения свойств у объекта анимации. Этот фреймворк тоже обладает подробной и красочной документацией, и обеспечен инструкцией по установке в своим проекты.

"Presentr" выигрывает удобством и минимализмом для настройки различных показов модальных контроллеров. С другой стороны, отсутствуют возможности для конфигурирования показов немодальных контроллеров. Большой недостаток в конфигурировании различных переходов, поскольку придется либо пользоваться теми, что уже доступны, либо создавать свои переходы, а также отсутствие поддержки версий системы, в которых еще не было необходимого API.

Фреймворк для презентаций "PresenterKit". Написан на языке Swift 3, так же поддерживает, как и предыдущий, прошлые версии языка. Позволяет значительно упростить показ различных видов новых контроллеров, так же присутствует ограничение на версию операционной системы, поддержка только с 7 версии.

Ключевая особенность - возможность проще настраивать показы новых контроллеров, где достаточно указать тип конфигурации. По сравнению со стандартным показом контроллеров - это получается более компактно, а также более удобно для разработчиков. Имеется наличие сгенерированной документации с относительно хорошим процентом покрытия.

Имеется ряд существенных недостатков: во-первых, он не представляет собой нововведений, а только упрощает взаимодействие со стандартными компонентами, что естественно уменьшает его ценность, поскольку каждое решение должно не только привносить удобство для взаимодействия, а увеличивать гибкость некоторых компонентов или просто расширять стандартный набор. В данном случае, этот фреймворк является лишь еще одним, возможно более удобным интерфейсом для взаимодействия. Во-вторых, сам процесс анимирования вида нового контроллера происходит в несколько этапов, а при использовании фреймворка их необходимо сохранить - в то время как было бы очень удобно избавить разработчика от необходимости писать постоянно повторяющийся код, создавать однотипные объекты.

Несмотря на все перечисленное, данный фреймворк заслуживает внимания - но его главный недостаток не может остаться незамеченным, поскольку привнесение новых видом анимации является одной из главных особенностей подобных фреймворков. С другой стороны, упрощение стандартных вещей не может не оправдывать его использование.

Фреймворк для анимации "HYBControllerTransitions". Написан на Objective-C, поддерживает версии операционной системы, начиная с iOS 7. Главная особенность - предоставляет большое количество нестандартных переходов, такие как "появление из шара", "частичное появление снизу", предоставляет различные версии для анимирования коллекций - что является существенным достоинством данного фреймворка по сравнению с другими вышеперечисленными. Из недостатков можно отметить следующие: остается большое количества кода для написания разработчика - что не позволяет легко и быстро совершать необходимые анимации, необходимость управления жизненным циклом объекта, передачей большого количества параметров в методы и блоки, отсутствие документации.

Таким образом, было рассмотрено несколько фреймворков, которые так или иначе имеют отношения к анимациям видов контроллеров или обычных компонентов. Заключить можно следующее: JHChainableAnimations является наиболее гибким и предоставляет нам оригинальное решение неудобного синтаксиса блоков, Presentr решает проблему показа модальных кон троллеров элегантным образом, PresenterKit является ничем более, чем просто удобным API над стандартным, а HYBControllerTransitions представляет необычный функционал, но требует значительных условий для его настройки. Более подробные сведения приведены в таблице.

Таблица - Сравнение основных качеств библиотек для анимирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерий | JHChainableAnimations | Presentr | PresenterKit | HYBControllerTransitions |
| Язык / ограничение на версию ОС | Objective-C / iOS 5+ | Swift (3) / iOS 9+ | Swift (3) /  iOS 9+ | Objective-C /  iOS 7+ |
| Главная задача | Анимация базовых компонент | Показ модальных окон | Удобная замена стандратному API | Нестандартный показ новых видов контроллеров |
| Достоинства | Простота, функциональность | Удобство | Удобство | Обширный функционал |
| Недостатки | Влияет только на компоненты UIView | Ограниченный функционал | Отсутствие новой функциональности | Преувеличенная сложность |
| Оценка | 9 / 10 | 8 / 10 | 7 / 10 | 9 / 10 |

**Литература**

1. Хиллегасс А. Objective-C Программирование для iOS и MacOS - СПб. : Питер, 2013. 254 с.
2. Nick Lockwood. iOS Core Animation: Advanced Techniques, [Addison-Wesley Professional](http://www.awprofessional.com/), 2013
3. Apple Inc. iOS Human Interface Guidelines, Apple Inc., 2015
4. Matt Neuburg. Programming iOS 9, O’Reilly, 2015
5. JHChainableAnimations [Электронный ресурс]. - Режим доступа: <https://github.com/jhurray/JHChainableAnimations> свободный. Язык англ. (дата обращения 28.11.2016)
6. Presentr [Электронный ресурс]. - Режим доступа: <https://github.com/IcaliaLabs/Presentr> свободный. Язык англ. (дата обращения 28.11.2016)
7. PresenterKit [Электронный ресурс]. - Режим доступа: <https://github.com/jessesquires/PresenterKit> свободный. Язык англ. (дата обращения 28.11.2016)
8. HYBControllerTransitions [Электронный ресурс]. - Режим доступа: <https://github.com/CoderJackyHuang/HYBControllerTransitions> свободный. Язык англ. (дата обращения 28.11.2016)