

- в мобильной разработке > 6 лет
- люблю делать красивые анимации



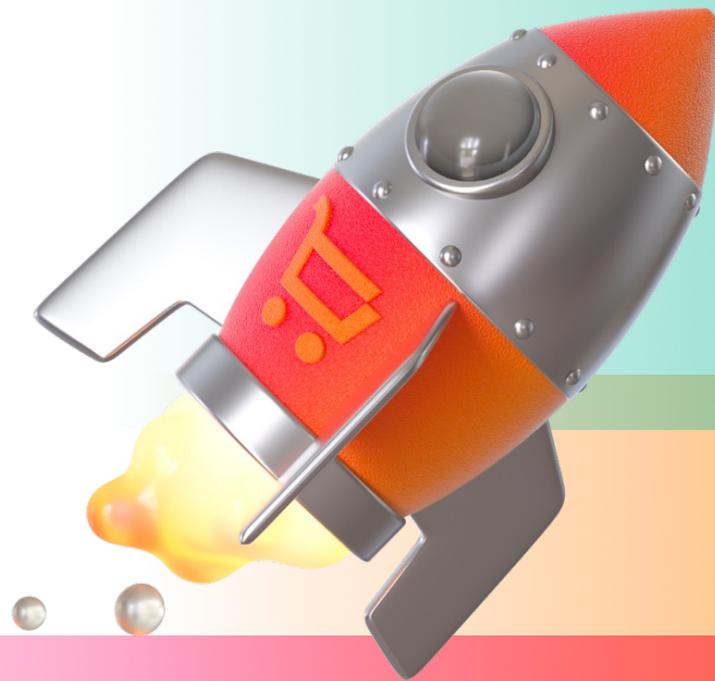
@PROKOPIEVEVGENI



Евгений Прокопьев

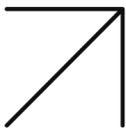
Мобильный разработчик в СберМаркете

Убиваем Lottie? Тащим Rive?



содержание

1. Обозначение проблемы
2. Lottie
3. Rive
4. Сравнение



???

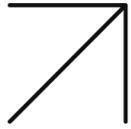
В чем вообще проблема





Надо сделать так





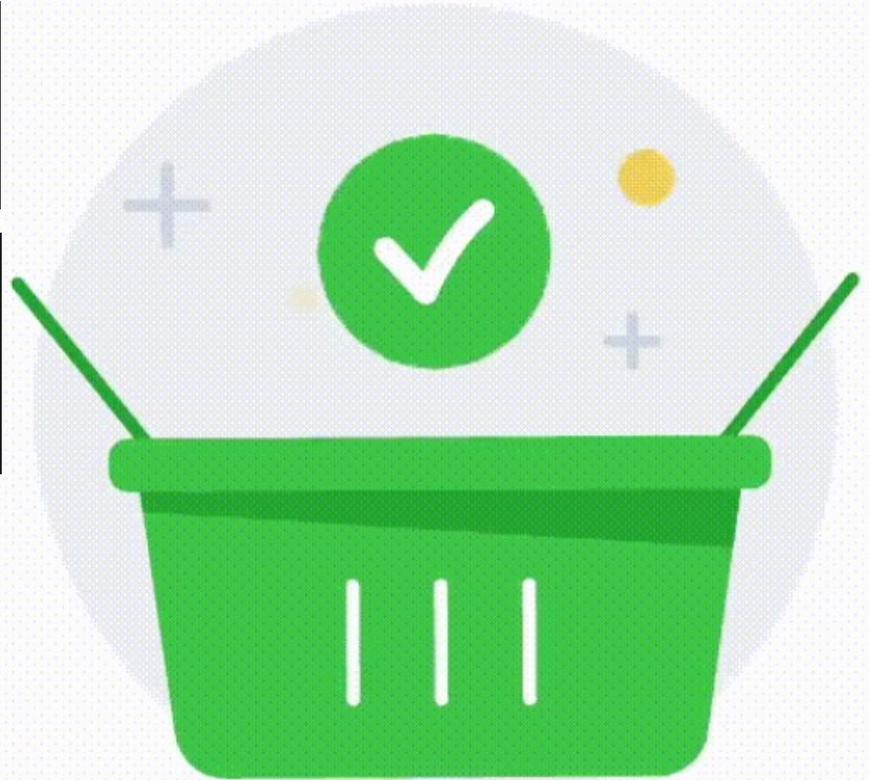
Trash vs Hard coding vs Lottie

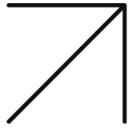
1.gif

GIF image - 28 MB

1.mp4

MPEG-4 movie - 2,1 MB





Trash vs Hard coding vs Lottie





Trash vs Hard coding vs Lottie

```
{\"v\":5.10.2,\"fr\":60,\"ip\":0,\"op\":1380,\"w\":328,\"h\":412,\"nm\":\"Comp 3\",\"ddd\":0,\"assets\":\n[{\n  \"id\":\"comp_0\",\"nm\":\"woman 2.2\",\"fr\":60,\"layers\":{\n    \"ddd\":0,\"ind\":2,\"ty\":4,\"nm\":\"teleshka\n    Outlines\",\"s\":1,\"ks\":{\n      \"a\":0,\"k\":100,\"ix\":11,\"r\":{\n        \"a\":0,\"k\":0,\"ix\":10,\"p\":\n        {\n          \"a\":0,\"k\":[795,1329,0],\"ix\":2,\"l\":2,\"a\":{\n            \"a\":0,\"k\":[545.5,307.5,0],\"ix\":1,\"l\":2,\"s\":\n            {\n              \"a\":0,\"k\":[100,100,100],\"ix\":6,\"l\":2,\"a\":0,\"shap\":{\n                \"ty\":\"gr\",\"it\":\n                {\n                  \"ind\":0,\"ty\":\"sh\",\"ix\":1,\"ks\":{\n                    \"a\":0,\"k\":{\n                      \"i\":[[0,0],[0,0],[0,0],[0,0]],\"o\":[[0,0],\n                      [0,0],[0,0],[0,0]],\"v\":[[-381.718,5.31],[381.751,-1.792],[381.718,-5.31],\n                      [-381.751,1.792]],\"c\":true,\"ix\":2,\"nm\":\"Path 1\",\"mn\":\"ADBE Vector Shape -\n                      Group\",\"hd\":false,\"ty\":\"fl\",\"c\":{\n                        \"a\":0,\"k\":[1,1,1],\"ix\":4,\"o\":\n                        {\n                          \"a\":0,\"k\":100,\"ix\":5,\"r\":1,\"bm\":0,\"nm\":\"Fill 1\",\"mn\":\"ADBE Vector Graphic -\n                          Fill\",\"hd\":false,\"ty\":\"tr\",\"p\":{\n                            \"a\":0,\"k\":[615.702,408.306],\"ix\":2,\"a\":{\n                              \"a\":0,\"k\":\n                              {\n                                \"a\":0,\"k\":100,\"ix\":7,\"sk\":{\n                                  \"a\":0,\"k\":0,\"ix\":4,\"sa\":\n                                  {\n                                    \"a\":0,\"k\":0,\"ix\":5,\"nm\":\"Transform\"}],\"nm\":\"Group\n                                  1\",\"np\":2,\"cix\":2,\"bm\":0,\"ix\":1,\"mn\":\"ADBE Vector Group\",\"hd\":false,\"ty\":\"gr\",\"it\":\n                                  {\n                                    \"ind\":0,\"ty\":\"sh\",\"ix\":1,\"ks\":{\n                                      \"a\":0,\"k\":{\n                                        \"i\":[[0,0],[0,0],[0,0],[0,0]],\"o\":[[0,0],\n                                        [0,0],[0,0],[0,0]],\"v\":[[-381.718,5.31],[381.751,-1.792],[381.718,-5.31],\n                                        [-381.751,1.792]],\"c\":true,\"ix\":2,\"nm\":\"Path 1\",\"mn\":\"ADBE Vector Shape -\n                                        Group\",\"hd\":false,\"ty\":\"fl\",\"c\":{\n                                          \"a\":0,\"k\":[1,1,1],\"ix\":4,\"o\":\n                                          {\n                                            \"a\":0,\"k\":100,\"ix\":5,\"r\":1,\"bm\":0,\"nm\":\"Fill 1\",\"mn\":\"ADBE Vector Graphic -\n                                            Fill\",\"hd\":false,\"ty\":\"tr\",\"p\":{\n                                              \"a\":0,\"k\":[561.152,302.757],\"ix\":2,\"a\":{\n                                                \"a\":0,\"k\":\n                                                {\n                                                  \"a\":0,\"k\":100,\"ix\":7,\"sk\":{\n                                                    \"a\":0,\"k\":0,\"ix\":4,\"sa\":\n                                                    {\n                                                      \"a\":0,\"k\":0,\"ix\":5,\"nm\":\"Transform\"}],\"nm\":\"Group\n                                                  2\",\"np\":2,\"cix\":2,\"bm\":0,\"ix\":2,\"mn\":\"ADBE Vector Group\",\"hd\":false,\"ty\":\"gr\",\"it\":\n                                                  {\n                                                    \"ind\":0,\"ty\":\"sh\",\"ix\":1,\"ks\":{\n                                                      \"a\":0,\"k\":{\n                                                        \"i\":[[0,0],[0,0],[0,0],[0,0]],\"o\":[[0,0],\n                                                        [0,0],[0,0],[0,0]],\"v\":[[401.101,-1.808],[401.069,-5.326],\n                                                        [-401.101,1.808]],\"c\":true,\"ix\":2,\"nm\":\"Path 1\",\"mn\":\"ADBE Vector Shape -\n                                                        Group\",\"hd\":false,\"ty\":\"fl\",\"c\":{\n                                                          \"a\":0,\"k\":[1,1,1],\"ix\":4,\"o\":\n                                                          {\n                                                            \"a\":0,\"k\":100,\"ix\":5,\"r\":1,\"bm\":0,\"nm\":\"Fill 1\",\"mn\":\"ADBE Vector Graphic -\n                                                            Fill\",\"hd\":false,\"ty\":\"tr\",\"p\":{\n                                                              \"a\":0,\"k\":[524.212,197.193],\"ix\":2,\"a\":{\n                                                                \"a\":0,\"k\":\n                                                                {\n                                                                  \"a\":0,\"k\":100,\"ix\":7,\"sk\":{\n                                                                    \"a\":0,\"k\":0,\"ix\":4,\"sa\":\n                                                                    {\n                                                                      \"a\":0,\"k\":0,\"ix\":5,\"nm\":\"Transform\"}],\"nm\":\"Group\n                                                                  3\",\"np\":2,\"cix\":2,\"bm\":0,\"ix\":3,\"mn\":\"ADBE Vector Group\",\"hd\":false,\"ty\":\"gr\",\"it\":\n                                                                  {\n                                                                    \"ind\":0,\"ty\":\"sh\",\"ix\":1,\"ks\":{\n                                                                      \"a\":0,\"k\":{\n                                                                        \"i\":[[0,0],[0,0],[0,0],[0,0]],\"o\":[[0,0],\n                                                                        [0,0],[0,0],[0,0]],\"v\":[[110.624,-206.761],[107.51,208.307],[110.624,206.761]\n                                                                      ]\n                                                                    ]\n                                                                  ]\n                                                                ]\n                                                              ]\n                                                            ]\n                                                          ]\n                                                        ]\n                                                      ]\n                                                    ]\n                                                  ]\n                                                ]\n                                              ]\n                                            ]\n                                          ]\n                                        ]\n                                      ]\n                                    ]\n                                  ]\n                                ]\n                              ]\n                            ]\n                          ]\n                        ]\n                      ]\n                    ]\n                  ]\n                ]\n              ]\n            ]\n          ]\n        ]\n      ]\n    ]\n  ]\n}
```

1.json
JSON Document - 1.8 MB

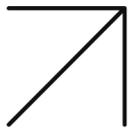
```
lottie.loadAnimation({  
  container: element,  
  renderer: 'svg',  
  loop: true,  
  autoplay: true,  
  path: 'data.json'  
});
```



???

Кто слышал о Lottie



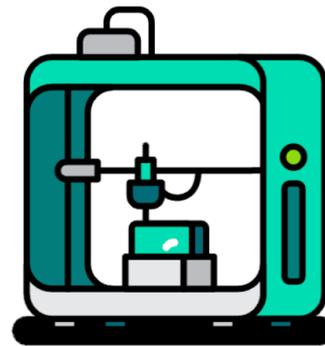


Конечно, Lottie

Lightweight, scalable animations for your websites and apps

LottieFiles takes away the complexity from motion design. It lets you create, edit, test, collaborate and ship a Lottie in the easiest way possible.

[Go to Dashboard](#)





Hernan Torrisi

Создатель Lottie, сейчас работает над Rive.app

2015: создал плагин - Bodymovin для экспорта анимаций из After Effects и первую версию плеера



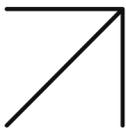
2017: релизнули плеер для android и ios



2020: создали новый формат dotLottie



2023: редактор для создания анимаций на базе проекта thorvg



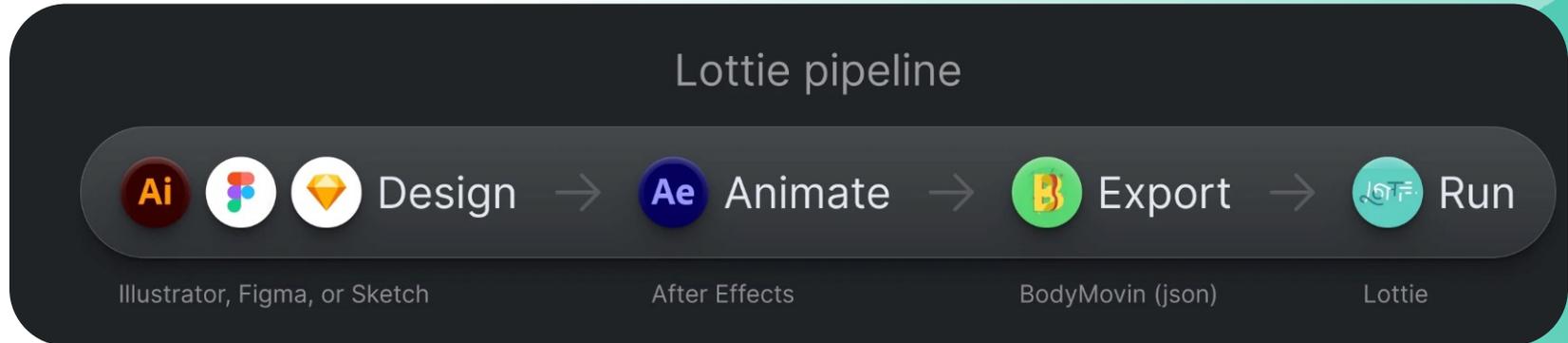
???

Сколько этапов от создания
до проигрывания анимации



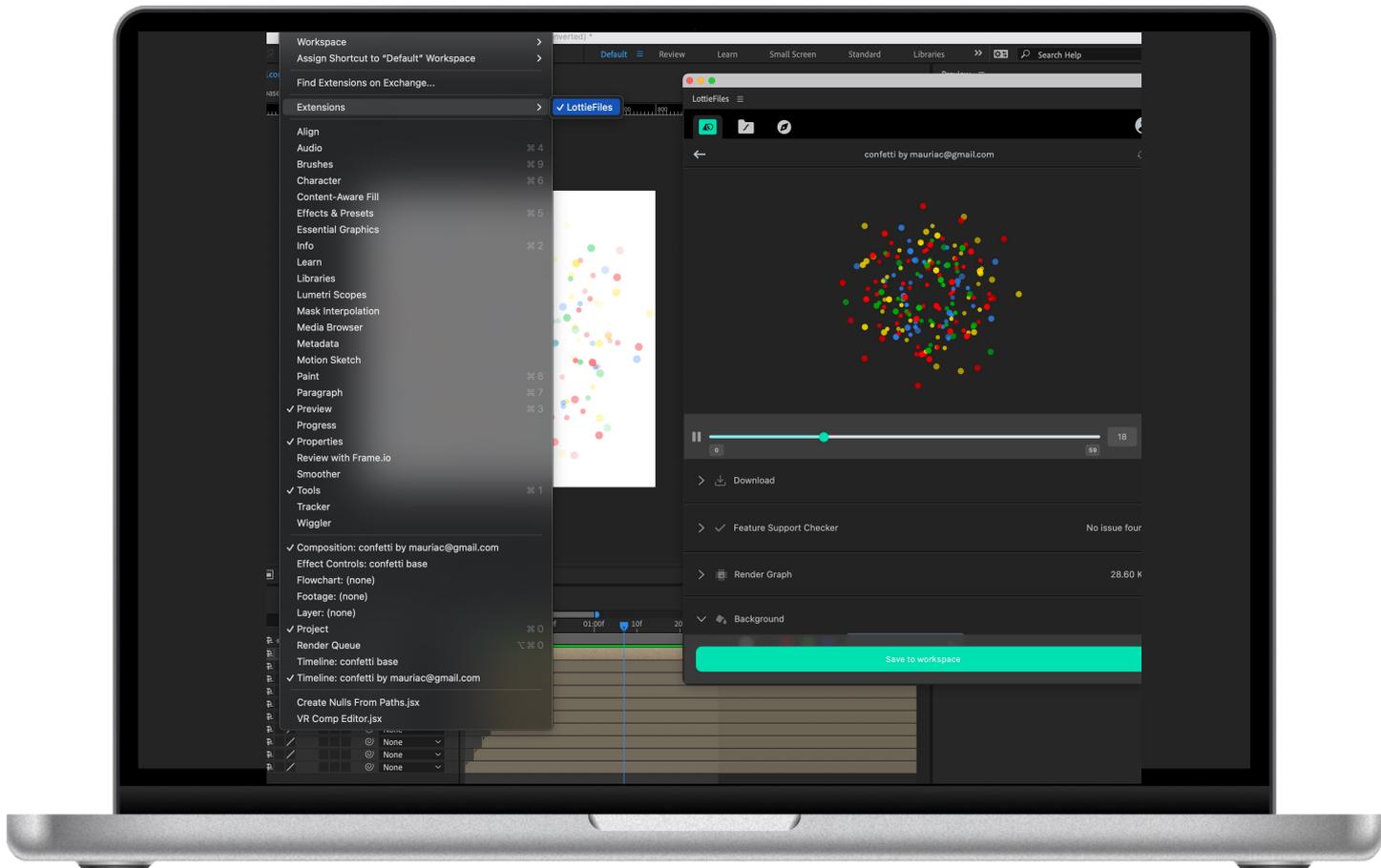


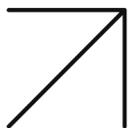
Пайплайн создания анимаций на Lottie





Плагин для AE





СПИСОК СВОЙСТВ

Lottie Supported Features

Lottie does not support all features of Adobe After Effects; please refer to the list below for details.

To test your animations with web, iOS & Android players, please use LottieFiles [online preview](#), [desktop](#) and [mobile](#) apps.

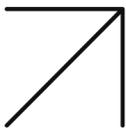
Shapes	Android	iOS	Windows	Web (SVG)	Web (Canvas)	Web (HTML)
Shape	👍	👍	👍	👍	👍	👍
Ellipse	👍	👍	👍	👍	👍	👍
Rectangle	👍	👍	👍	👍	👍	👍
Rounded Rectangle	👍	👍	👍	👍	👍	👍
Polystar	👍	👍	🚫	👍	👍	👍
Group	👍	👍	👍	👍	👍	👍
Repeater	👍	🚫	👍	👍	👍	👍
Trim Path (individually)	👍	👍	🚫	👍	👍	👍
Trim Path (simultaneously)	👍	👍	👍	👍	👍	👍



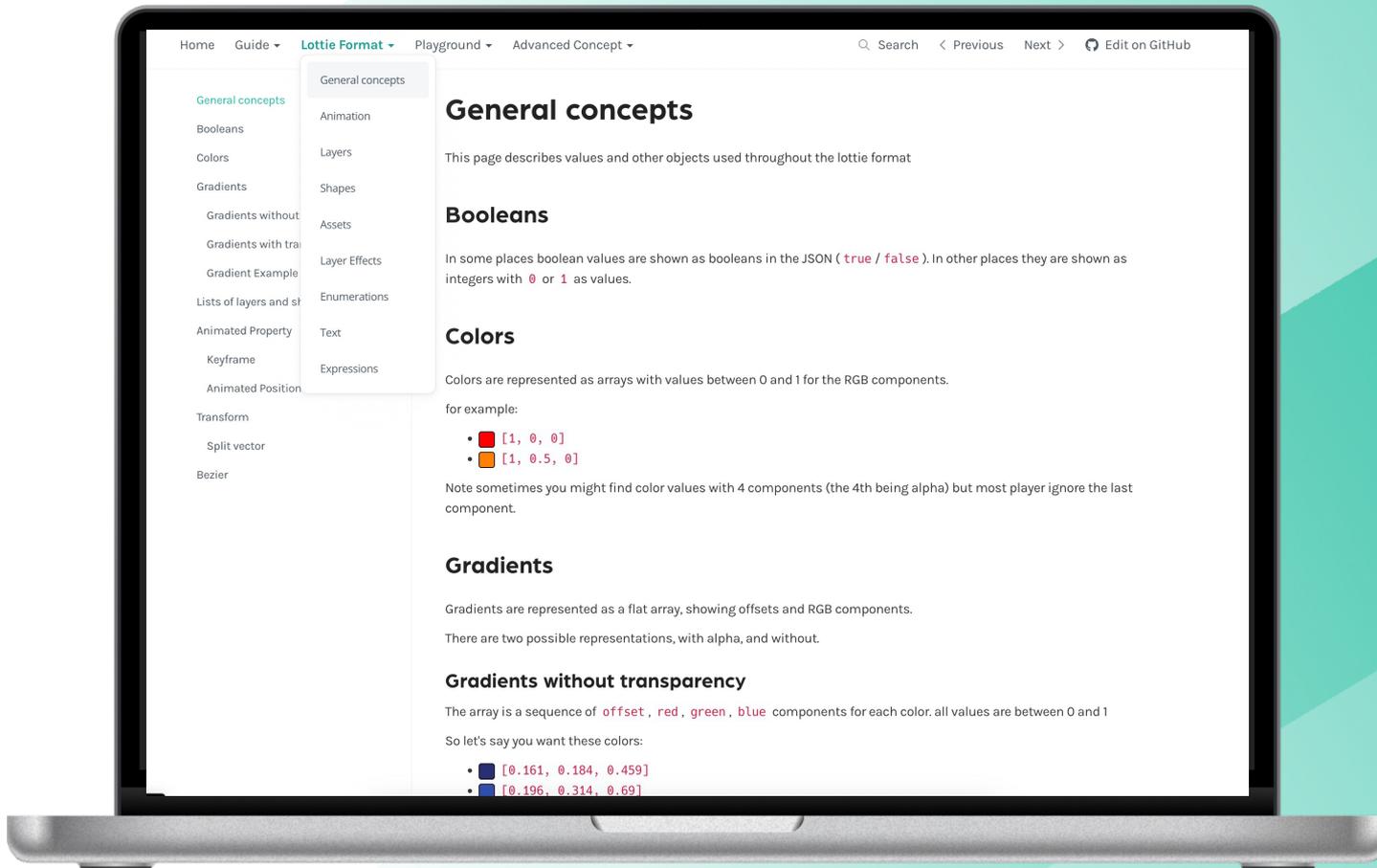
На выходе JSON, внутри инструкции

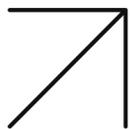
```
{
  "v": "5.7.1",
  "ip": 0,
  "op": 180,
  "nm": "Animation",
  "fr": 60,
  "w": 512,
  "h": 512,
  "layers": [
    {
      "ddd": 0,
      "ty": 4,
      "ind": 0,
      "st": 0,
      "ip": 0,
      "op": 180,
      "nm": "Layer",
      "ks": {
        "a": {
          "a": 0,
          "k": [
            256,
            256
          ]
        },
        "p": {
          "a": 0,
          "k": [
            256,
            256
          ]
        },
        "s": {
          "a": 0,

```



Внутри инструкции

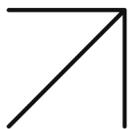




Hernan про работу над плеером

«Specifically talking about Lottie on the web, the player file size is pretty large, and it's a lot of JavaScript. It's like trying to have After Effects in 60 kilobytes of JavaScript»

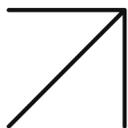
(Hernan)



Загрузка из asset

```
if
  let animationCache = animationCache,
  let animation = animationCache.animation(forKey: cacheKey)
{
  /// If found, return the animation.
  return animation
}

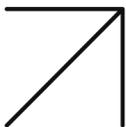
do {
  /// Load jsonData from Asset
  let json = try Data(assetName: name, in: bundle)
  /// Decode animation.
  let animation = try LottieAnimation.from(data: json)
  animationCache?.setAnimation(animation, forKey: cacheKey)
  return animation
} catch {
```



Чтение инструкций

```
startFrame = try dictionary.value(for: CodingKeys.startFrame)
endFrame = try dictionary.value(for: CodingKeys.endFrame)
framerate = try dictionary.value(for: CodingKeys.framerate)
width = try dictionary.value(for: CodingKeys.width)
height = try dictionary.value(for: CodingKeys.height)
let layerDictionaries: [[String: Any]] = try dictionary.value(for: CodingKeys.layers)
layers = try [LayerModel].fromDictionaries(layerDictionaries)
if let glyphDictionaries = dictionary[CodingKeys.glyphs.rawValue] as? [[String: Any]] {
    glyphs = try glyphDictionaries.map { try Glyph(dictionary: $0) }
} else {
    glyphs = nil
}
if let fontsDictionary = dictionary[CodingKeys.fonts.rawValue] as? [String: Any] {
    fonts = try FontList(dictionary: fontsDictionary)
} else {
    fonts = nil
}
```

```
enum CodingKeys: String, CodingKey {
    case version = "v"
    case type = "ddd"
    case startFrame = "ip"
    case endFrame = "op"
    case framerate = "fr"
    case width = "w"
    case height = "h"
    case layers
    case glyphs = "chars"
    case fonts
    case assetLibrary = "assets"
    case markers
}
```



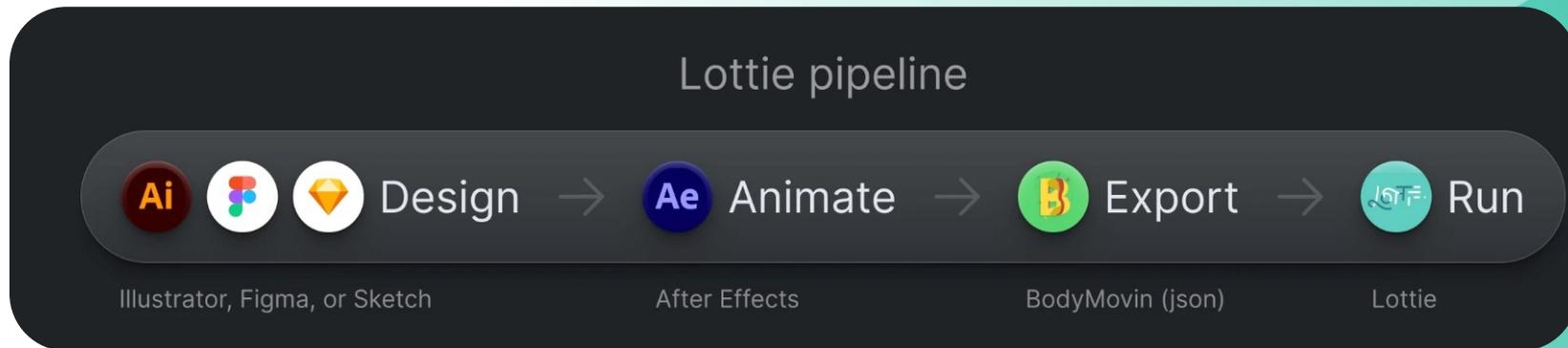
Рисование

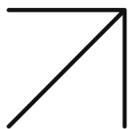
```
for layer in layers.reversed() {  
    layer.bounds = bounds  
    animationLayers.append(layer)  
    if let imageLayer = layer as? ImageCompositionLayer {  
        imageLayers.append(imageLayer)  
    }  
    if let textLayer = layer as? TextCompositionLayer {  
        textLayers.append(textLayer)  
    }  
    if let matte = mattedLayer {  
        /// The previous layer requires this layer to be its matte  
        matte.matteLayer = layer  
        mattedLayer = nil  
        continue  
    }  
    if  
        let matte = layer.matteType,  
        matte == .add || matte == .invert  
    {  
        /// We have a layer that requires a matte.  
        mattedLayer = layer  
    }  
    addSublayer(layer)  
}
```

```
layerImageProvider.addImageLayers(imageLayers)  
layerImageProvider.reloadImages()  
layerTextProvider.addTextLayers(textLayers)  
layerTextProvider.reloadTexts()  
layerFontProvider.addTextLayers(textLayers)  
layerFontProvider.reloadTexts()
```

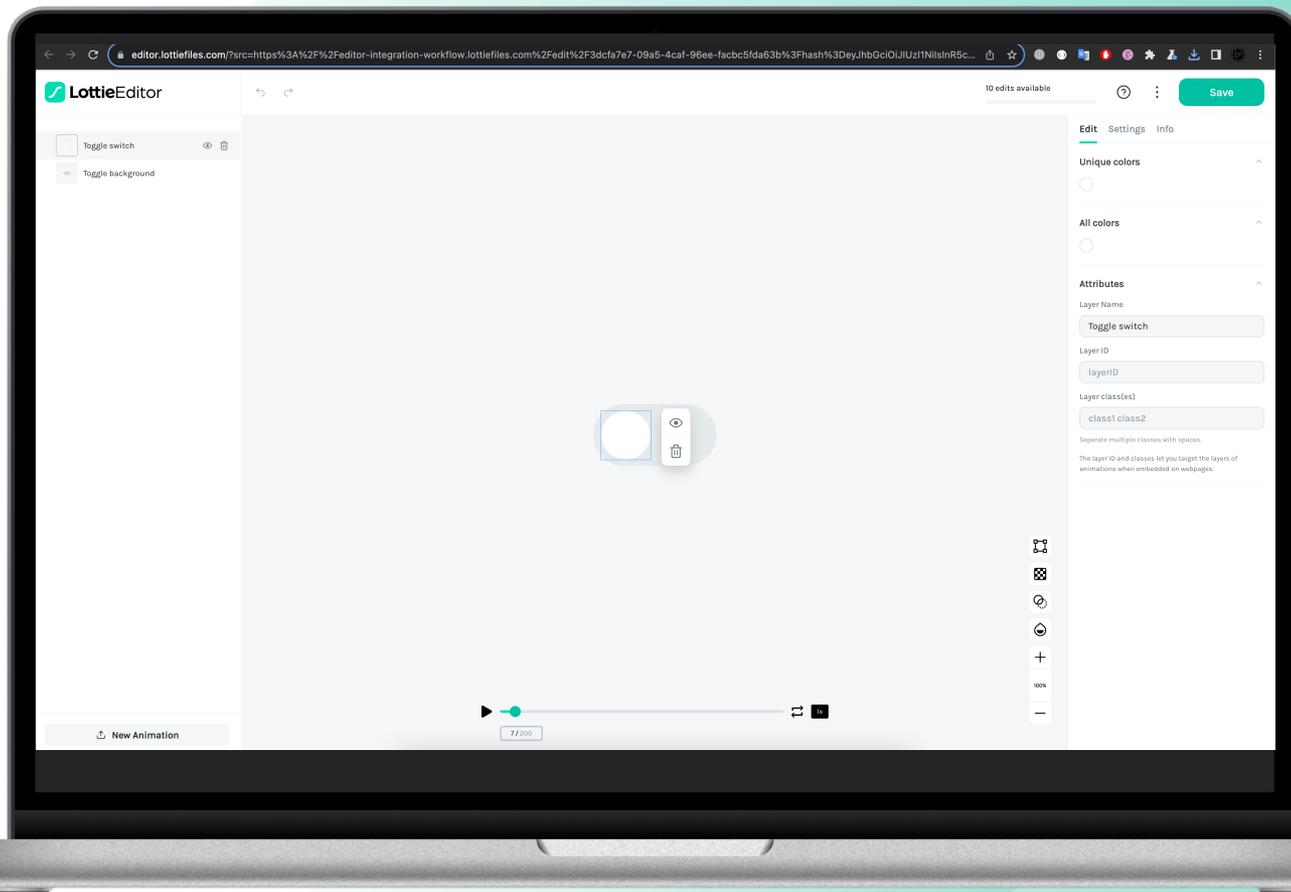


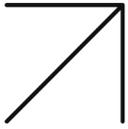
Пайплайн создания анимаций на Lottie



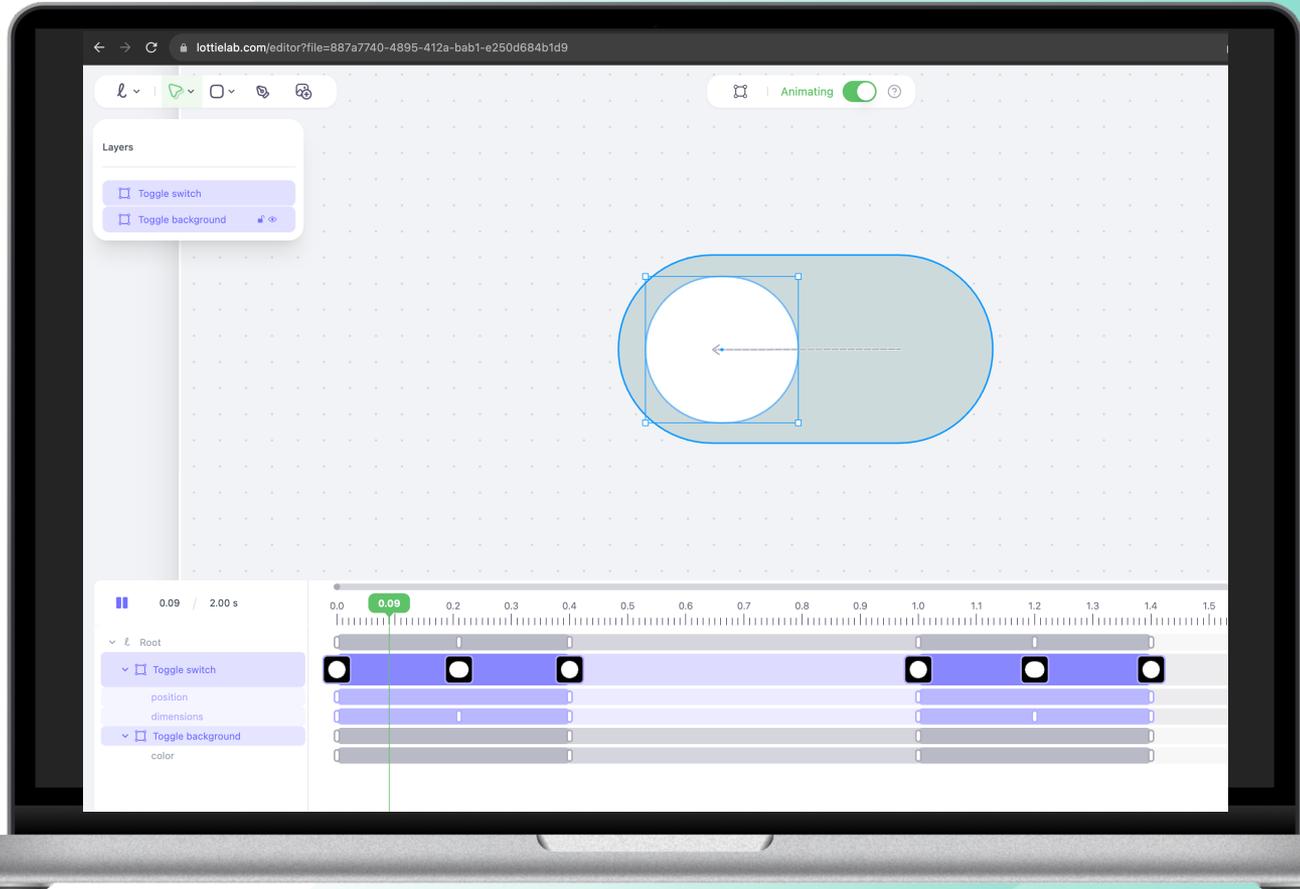


editor.lottiefiles.com





lottielab

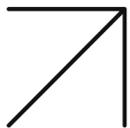




Саммари: Lottie



Нет единого механизмы для создания анимаций



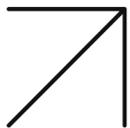
Саммари: Lottie

- Нет единого механизма для создания анимаций
- Тем более для воспроизведения



Саммари: Lottie

- Нет единого механизма для создания анимаций
- Тем более для воспроизведения
- Нельзя в рантайме влиять на анимации



Саммари: Lottie



Нет единого механизма



Тем



ить на анимации

Отличный инструмент для воспроизведения анимаций



!?!
!?!
!?!

Но!! Новая фича...



Фича

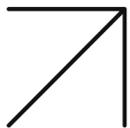




???

Как реализовать такое





Дежавю



Trash vs **Hard coding** vs Lottie





Rive, welcome

The screenshot shows the Rive website homepage. At the top, there is a navigation bar with the Rive logo, links for Products, Community, Learn, Pricing, and Downloads, and a pink 'Get Started' button. The main content area features the Rive logo, the headline 'Build interactive animations that run anywhere', and the tagline 'Blazing fast. Tiny size. Made for runtime.' Below this is another pink 'Get Started' button. Three featured examples are shown in a row: a video game character scene, an animated UI component table, and a brand logo with wings.

rive.app

Products Community Learn Pricing Downloads

Get Started

RIVE

Build interactive animations
that run anywhere

Blazing fast. Tiny size. Made for runtime.

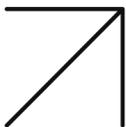
Get Started

Bring video games to life with characters, props, and UI

	Date Added	Rating
in Color	Apr 4, 2022	☆☆☆☆☆
Rigger	Mar 28, 2022	☆☆☆☆☆

Find More

Bring brands to life with animated hero moments



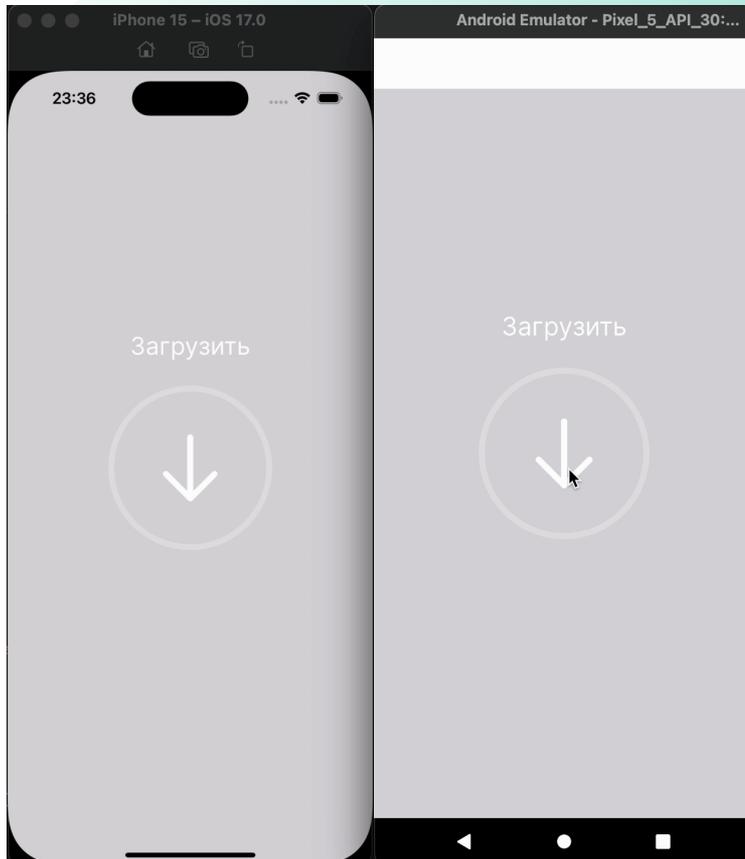
Поверим, что дальше?

```
animation.riv x
1 5249 5645 0700 cf14 a001 0000 0000 0017
2 0001 0405 5472 7563 6b07 0000 f044 0800
3 0087 4409 00e0 fcc4 0a00 00fa 4200 0204
4 0472 6f6f 7405 0010 cdcc 4c3f 11cd cc4c
5 3f0d d907 8144 0e17 fd46 4400 0204 0574
6 7275 636b 0501 0002 0502 0d80 83e8 c00e
7 bc08 9bc3 0002 0404 626f 6479 0503 108f
8 9488 3f11 2c13 8d3f 0dbc 5c06 430e a9e8
9 2543 0002 0405 6d6f 746f 7205 0410 deea
10 6f3f 1117 4668 3f00 0304 0573 6861 7065
11 0505 1001 00a0 3f11 0000 a03f 0dba af83
12 c10e afb1 d8c3 0010 0405 0450 6174 6805
13 0620 0100 0505 0718 00ca ea40 1900 d597
14 401a 0090 6941 0005 0507 1820 c94d c219
15 38db 0242 1a00 9069 4100 0505 0718 003f
16 c440 19b0 f73a 421a 0090 6941 0005 0507
17 1860 dc79 4219 e031 9a41 1a00 9069 4100
18 1205 9302 25a1 9a44 ff00 0304 0472 6f6f
19 6605 0510 9094 883f 112c 138d 3f0d 6ebc
20 b0c3 0e5c d387 c300 1004 0d0c 5041 5354
21 4544 3a20 5061 7468 050d 2001 0006 050e
22 18e8 901b c219 8e81 82c1 540c db19 4055
23 d1ee 5941 563d d33c bf57 98f6 e341 0023
24 050e 1893 c688 4319 1c67 25c3 5285 04a4
25 bd53 2bcf 1c42 0006 050e 1844 f1f9 4319
```

```
RiveViewModel(
    fileName: "animation",
    autoPlay: true
).view().frame(
    width: 200,
    height: 200,
    alignment: .center
)
```



Пробуем и все работает





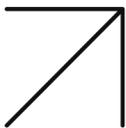
2014-2016: Max Wihlborg и Michelle Engvall
начали разрабатывать инструмент

2017: Запуск Flare, инструмент
для создания анимаций и
экспорта их в gif или Lottie

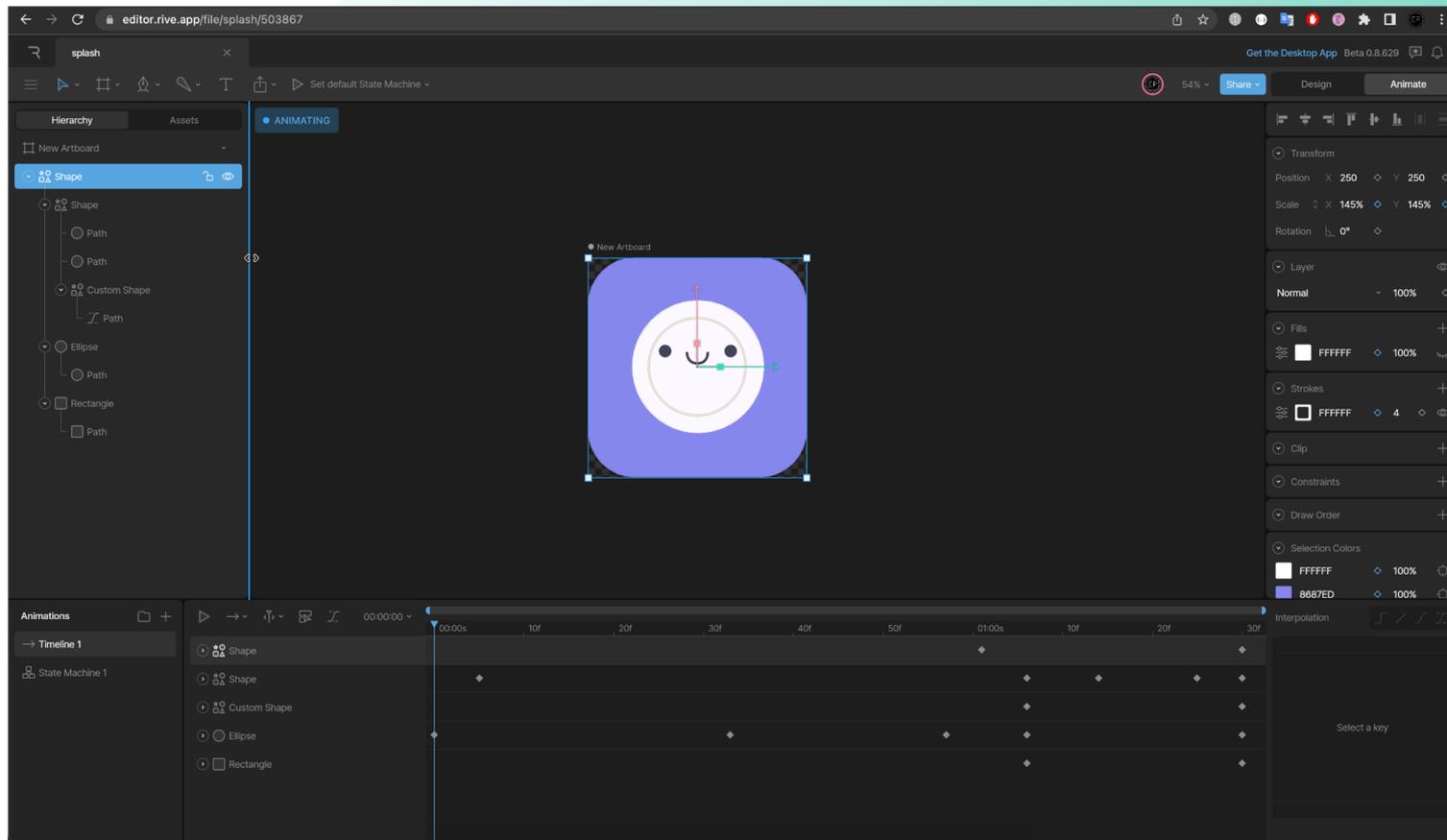
2019: Flare -> Rive

2020: Пошли в кроссплатформу,
включая Flutter, Unity, React
Native

2021+ Активно внедряют новые
фичи

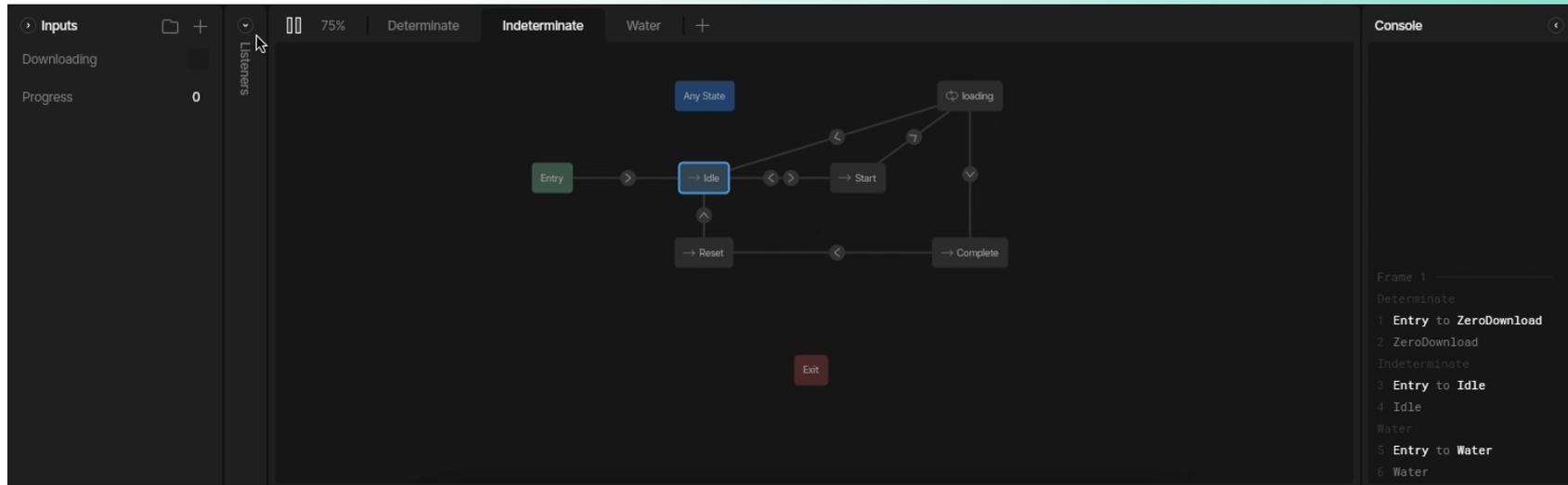


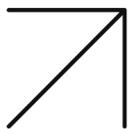
Полноценный редактор





Машина состояний





Зависимости извне - boolean

The screenshot displays the Unity Inspector for a state machine component named "Face". The state machine is configured with the following states and transitions:

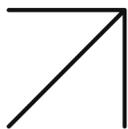
- Entry** (green box) transitions to **FaceUp** (blue box).
- FaceUp** (blue box) transitions to **FaceDown** (grey box).
- FaceDown** (grey box) transitions to **Any State** (blue box).
- Any State** (blue box) transitions to **Exit** (red box).

The right-hand panel shows the configuration for the selected state, "FaceUp to FaceDown":

- Duration:** 200 ms
- Exit Time:** 0 ms
- Pause when exiting "FaceUp":**
- Conditions:**
 - isShow:** true
- Events:**
- Interpolation:** A graph showing a smooth curve from 0 to 1.

The Console window on the right lists the following log messages:

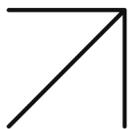
```
Frame: 1
Loop
1 Entry to Idle
2 Idle
PlayAgainButton
3 Entry to PlayAgainDefa...
4 PlayAgainDefault
Face
5 Entry to FaceUp
6 FaceUp
Eyes
7 Entry to EyesIdle
8 EyesIdle
Lights
9 Entry to IdleLigh
10 IdleLigh
DanceStop
11 Entry to Dance
12 Dance
```



Зависимости извне - trigger

The screenshot displays the Unity Inspector interface for a state machine component named "DanceStop".

- Inputs:** scrollValue (0), isShow, tap.
- Listeners:** A vertical list of listeners, with a mouse cursor pointing to the "tap" listener.
- State Machine Diagram:** A flowchart showing states: "Entry" (green box) transitions to "Dance" (blue box). From "Dance", there are transitions to "Stop" (grey box) and back to "Dance". A blue circle with a right-pointing arrow is positioned between the "Dance" and "Stop" transitions. Below the diagram is a blue box labeled "Any State".
- Console:** A log of events including:
 - 86 PlayAgain to PlayAgai...
 - 87 PlayAgainDefault
 - 88 tap fired
 - 89 Celebration to Idle
 - 90 Idle
 - 91 PlayAgainDefault to P...
 - 92 PlayAgain
 - 93 EyesDance to EyesIdle
 - 94 EyesIdle
 - 95 Stop to Dance
 - 96 Dance
 - 97 PlayAgain to PlayAgai...
 - 98 PlayAgainDefault
- Inspector (Dance to Stop):** Configuration for the transition from "Dance" to "Stop".
 - Duration: 0 ms
 - Exit Time: 0 ms
 - Pause when exiting "Dance":
 - Conditions: tap
 - Events:
 - Interpolation:



Зависимости извне - number

The screenshot displays the Animate CC software interface. On the left, the 'Inputs' panel lists 'scrollValue' (0), 'isShow', and 'tap'. The central workspace shows a state machine diagram with states: 'Entry', 'FaceUp', 'FaceDown', 'Any State', and 'Exit'. 'FaceUp' is highlighted with a blue border. The right panel shows the 'Console' with a list of 12 frame events, including 'Entry to Idle', 'Idle', 'Entry to PlayAgainDefa...', 'PlayAgainDefault', 'Face', 'Entry to FaceUp', 'FaceUp', 'Eyes', 'Entry to EyesIdle', 'EyesIdle', 'Lights', 'Entry to IdleLigh', 'IdleLigh', 'DanceStop', 'Entry to Dance', and 'Dance'. Below the console, the 'FaceUp to FaceDown' transition is configured with a duration of 200 ms, an exit time of 0 ms, and a condition 'isShow' set to 'true'. An interpolation graph is visible at the bottom right.



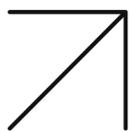
Внешние переменные

The screenshot displays a state machine editor interface. At the top center, a blue square contains a white downward-pointing arrow. Below this, the editor shows a state transition diagram with the following components:

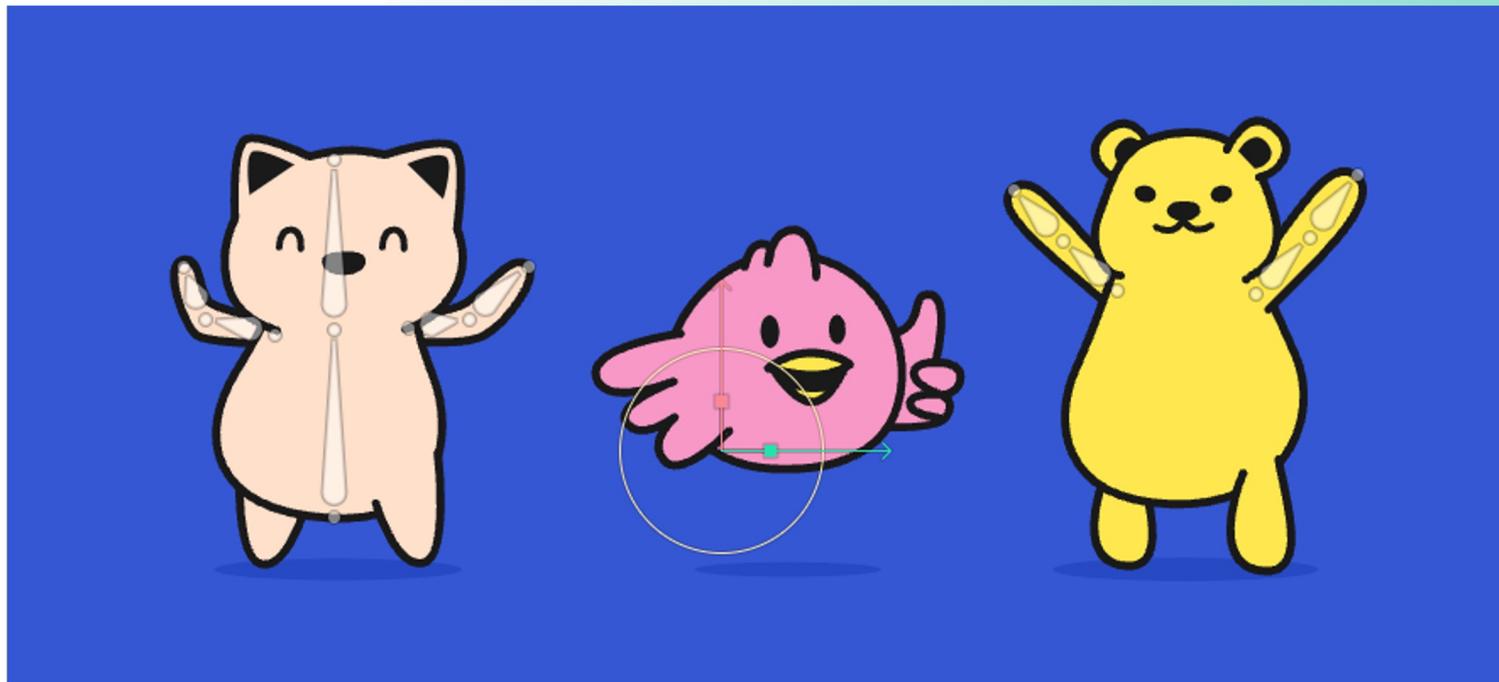
- States:** Entry (green), ZeroDownload (blue), OneHundredDownload (grey), Exit (red), and Any State (blue).
- Transitions:** An arrow points from Entry to ZeroDownload. From ZeroDownload, arrows point to Blend and OneHundredDownload. From OneHundredDownload, an arrow points to Exit. From Exit, an arrow points to Any State.
- Blend:** A central node labeled 'Blend' with an envelope icon, which branches into two paths leading to ZeroDownload and OneHundredDownload.

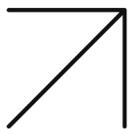
The interface includes a left sidebar with 'Inputs' (Download, Progress) and 'Listeners'. The top bar shows '78%' and tabs for 'Determinate', 'Indeterminate', and 'Water'. The right sidebar is a 'Console' with the following log:

```
Frame 1
Determinate
1 Entry to ZeroDownload
2 ZeroDownload
Indeterminate
3 Entry to Idle
4 Idle
Water
5 Entry to Water
6 Water
```

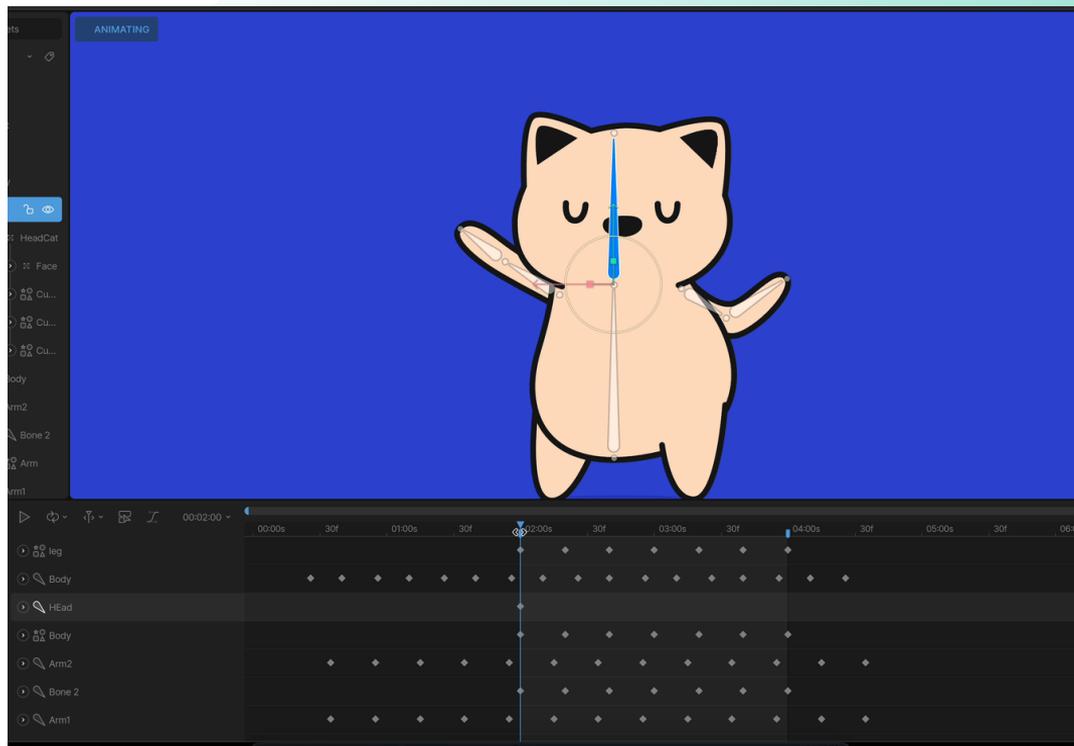


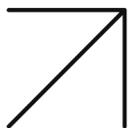
Создание костей





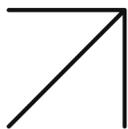
Создание костей





Lottie -> Rive

The screenshot shows a web browser window with the LottieFiles website. The main content is a preview of an animation titled "About Me" by Thomas Kiguru. The animation depicts a woman sitting inside a large, blue, stylized hand that is holding a smartphone. The interface includes a play button, a progress bar, and a "Download" button. Below the animation, there are sections for "Animation Details" and "Asset link & embed". The "Asset link & embed" section provides a URL: <https://asset-cdn.lol> and a "Generate link" button. The browser's address bar shows the URL lottiefiles.com/animations/about-me-ypck1.4A6z2. The browser's taskbar at the bottom shows various application icons.



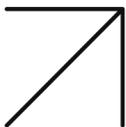
Features

State Machine

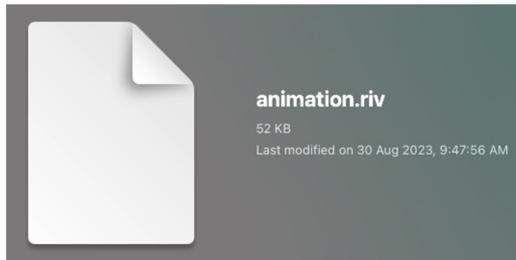
Built for interactivity

Empower designers to drive functionality that ships in the final product. The State Machine bridges the gap between design and development, making iteration a breeze.

- State Machines**
Create interactivity visually using animations as states.
- Transitions**
Connect animations with transitions and define their conditions.
- Listeners**
Make graphics respond to hovers, clicks, taps, and more.
- Inputs**
Inputs are the contract between design and engineering.
- Blend States**
Interactively layer and blend animations together.
- State Machine Layers**
Create concurrent functionality with state machine layers.



json.riv



```
animation.riv
1 5249 5645 0700 88ca 20c6 01ec 01ad 01c5
2 0100 0000 0000 1700 0107 0000 fa43 0800
3 00fa 43ec 0100 040c 4e65 7720 4172 7462
4 6f61 7264 0012 0502 2531 3131 ff00 1405
5 0000 1f37 0a54 696d 656c 696e 6520 3100
6 3537 0f53 7461 7465 204d 6163 6869 6e65
7 2031 0039 8a01 074c 6179 6572 2031 003d
8 9501 0000 3f00 4197 0100 0040 003e 0001
9 0417 616e 696d 6174 696f 6e5f 6c6c 786a
10 746b 7a70 2e6a 736f 6e07 0000 fa44 0800
11 00fa 4409 0000 1644 ec01 0000 0204 0448
12 616e 6405 000d 0000 7a44 0e00 007a 4400
13 0205 010d 0000 7ac4 0e00 007a c400 0204
14 0b4d 6964 646c 6520 4c69 6e65 0502 0d81
15 4d62 440e 7bc8 9d44 0002 0503 0df8 93bd
16 420e ec21 83c3 0002 0407 5368 6170 6520
17 3105 0400 0204 0954 7261 6e73 666f 726d
18 0505 0002 0506 0003 0406 5061 7468 2031
19 0507 0010 0508 0006 0509 1800 003b c319
20 00c0 cb43 54df cacf bf55 7a5c 8a42 56d0
21 54c2 3f57 0e4a d142 0006 0509 1800 8082
22 c319 0000 ae43 54b8 f9f1 3f55 1163 3643
23 5602 26a0 bf57 7993 1543 0006 0509 1800
24 003b 4319 0000 de42 542f 1145 c055 f03f
25 8043 56de aa7f 3d57 f03f 8043 0012 05ac
26 0125 8200 29ff 0002 0405 5468 756d 6205
27 020d c1ca 1a44 0e00 d092 4400 0205 0e0d
28 7f6a be43 0e00 802e c300 0204 0753 6861
29 7065 2031 050f 0002 0409 5472 616e 7366
30 6f72 6d05 1000 0205 1100 0304 0650 6174
31 6820 3105 1200 1005 1300 0605 1418 0000
32 2cc4 1900 0096 c254 e578 33c0 5564 7606
33 4356 c3b7 ac3e 57e7 000e 4300 0605 1418
34 0000 b0c3 1921 0092 4354 8c91 d7bf 55e0
35 9677 4356 238e ba3f 578e 9d08 4300 0605
36 1418 0080 abc3 1900 00f0 4300 1205 ad01
37 25ff 6043 ff00 0204 0753 6861 7065 2032
38 050f 0002 0409 5472 616e 7366 6f72 6d05
39 1900 0205 1a00 0304 0650 6174 6820 3105
40 1b00 1005 1c00 0605 1d18 0000 2cc4 1900
41 0096 c254 e578 33c0 5564 7606 4356 c3b7
42 ac3e 57e7 000e 4300 0605 1d18 0000 b0c3
43 1921 0092 4354 8c91 d7bf 55e0 9677 4356
44 238e ba3f 578e 9d08 4300 0605 1d18 0080
45 abc3 1900 00f0 4300 1205 ae01 2582 0029
46 ff00 2f05 ae01 730a d763 3f75 0200 0204
47 0d45 606e 6375 7365 203e 4374 736e 0502
```



engine

rive-ios / submodules /

 **mjtalbot** patch up division by zero issue on normaliz

Name

 ..

 [rive-cpp @ a0fca85](#)

rive-android / submodules /

 **mjtalbot** patch up division by zero issue on no

Name

 ..

 [rive-cpp @ a0fca85](#)

n normalizing Le

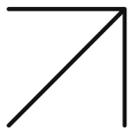
Name

 ..

 [rive-cpp @ a0fca85](#)

 ..

 [rive-cpp @ a0fca85](#)



.riv – binary

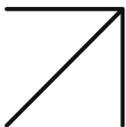
```
private:
    static constexpr char fingerprint[] = "RIVE";

    for (int i = 0; i < 4; i++) {
        auto b = reader.readByte();
        if (fingerprint[i] != b) {
            return false;
        }
    }

    header.m_MajorVersion = reader.readVarUintAs<int>();
    if (reader.didOverflow()) {
        return false;
    }
    header.m_MinorVersion = reader.readVarUintAs<int>();
    if (reader.didOverflow()) {
        return false;
    }

    header.m_FileId = reader.readVarUintAs<int>();

    if (reader.didOverflow()) {
        return false;
    }
}
```



.riv – binary

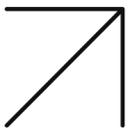
```
while (true) {
... auto propertyKey = reader.readVarUIntAs<uint16_t>();
... if (propertyKey == 0) {
...     // Terminator. https://media.giphy.com/media/7TtvTUmm9mp20/giphy.gif
...     break;
... }

... if (reader.hasError()) {
...     delete object;
...     return nullptr;
... }

... if (object == nullptr || !object->deserialize(propertyKey, reader)) {
...     // We have an unknown object or property, first see if core knows
...     // the property type.
...     int id = CoreRegistry::propertyFieldId(propertyKey);
...     if (id == -1) {
...         // No, check if it's in toc.
...         id = header.propertyFieldId(propertyKey);
...     }

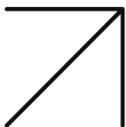
...     if (id == -1) {
...         // Still couldn't find it, give up.
...         fprintf(
...             stderr, "Unknown property key %d, missing from property ToC.\n", propertyKey);
...         delete object;
...         return nullptr;
...     }

...     switch (id) {
...     case CoreUIntType::id:
...         CoreUIntType::deserialize(reader);
...         break;
...     case CoreStringType::id:
...         CoreStringType::deserialize(reader);
...         break;
...     case CoreDoubleType::id:
...         CoreDoubleType::deserialize(reader);
...         break;
...     case CoreColorType::id:
...         CoreColorType::deserialize(reader);
...         break;
...     }
... }
}
```



.riv – binary

```
if (object->import(importStack) == StatusCode::Ok) {  
    ... switch (object->coreType()) {  
        ... case Backboard::typeKey:  
            ...     m_Backboard.reset(object->as<Backboard>());  
            ...     break;  
        ... case Artboard::typeKey: {  
            ...     Artboard* ab = object->as<Artboard>();  
            ...     ab->m_Factory = m_Factory;  
            ...     m_Artboards.push_back(std::unique_ptr<Artboard>(ab));  
            ...     } break;  
        ... case ImageAsset::typeKey: {  
            ...     auto fa = object->as<FileAsset>();  
            ...     m_FileAssets.push_back(std::unique_ptr<FileAsset>(fa));  
            ...     } break;  
        ... }  
    } else {  
        ... fprintf(stderr, "Failed to import object of type %d\n", object->coreType());  
        ... delete object;  
        ... continue;  
    }  
}
```



.riv – binary

```
case Backboard::typeKey:
    ... stackObject = new BackboardImporter(object->as<Backboard>());
    ... break;
case Artboard::typeKey:
    ... stackObject = new ArtboardImporter(object->as<Artboard>());
    ... break;
case LinearAnimation::typeKey:
    ... stackObject = new LinearAnimationImporter(object->as<LinearAnimation>());
    ... break;
case KeyedObject::typeKey:
    ... stackObject = new KeyedObjectImporter(object->as<KeyedObject>());
    ... break;
case KeyedProperty::typeKey: {
    ... auto importer =
    ...     importStack.latest<LinearAnimationImporter>(LinearAnimation::typeKey);
    ...     if (importer == nullptr) {
    ...         return ImportResult::malformed;
    ...     }
    ...     stackObject =
    ...         new KeyedPropertyImporter(importer->animation(), object->as<KeyedProperty>());
    ...     break;
}
```

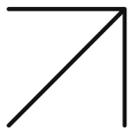


.riv – binary

```
case StateMachine::typeKey:
... stackObject = new StateMachineImporter(object->as<StateMachine>());
... break;
case StateMachineLayer::typeKey: {
... auto artboardImporter = importStack.latest<ArtboardImporter>(ArtboardBase::typeKey);
... if (artboardImporter == nullptr) {
... .. return ImportResult::malformed;
... }

... stackObject = new StateMachineLayerImporter(object->as<StateMachineLayer>(),
... .. artboardImporter->artboard());

... break;
}
case EntryState::typeKey:
case ExitState::typeKey:
case AnyState::typeKey:
case AnimationState::typeKey:
case BlendState1D::typeKey:
case BlendStateDirect::typeKey:
... stackObject = new LayerStateImporter(object->as<LayerState>());
... stackType = LayerState::typeKey;
... break;
case StateTransition::typeKey:
case BlendStateTransition::typeKey:
... stackObject = new StateTransitionImporter(object->as<StateTransition>());
... stackType = StateTransition::typeKey;
... break;
```



Саммари: Rive



Единый процесс создания и проигрывания



Саммари: Rive



Единый процесс создания и проигрывания



Много приятных фич



Саммари: Rive



Единый процесс создания и проигрывания



Много приятных фич



Поддержка большинства платформ



Саммари: Rive



Единый процесс создания и проигрывания



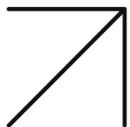
Много приятных фич



Поддержка большинства платформ



Конвертер Lottie -> Rive



Сравнение производительности

Lottie

vs

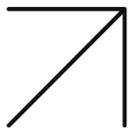
Rive



Пример 1

Простые трансформации
позиции элементов





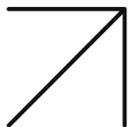
Пример 1



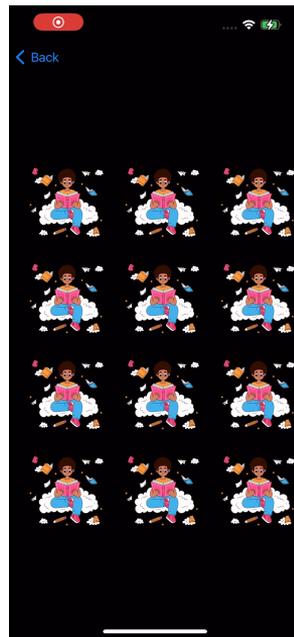
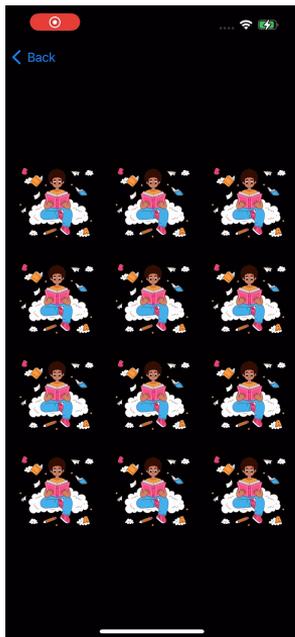
RAM:

35 MB

72 MB



Пример 1



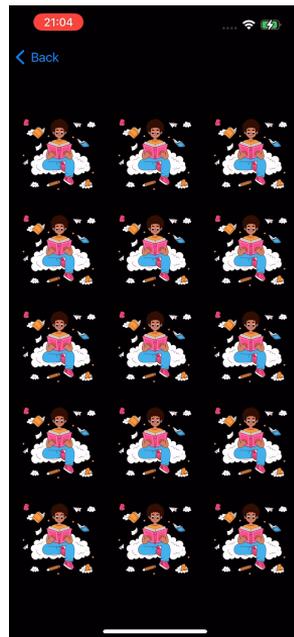
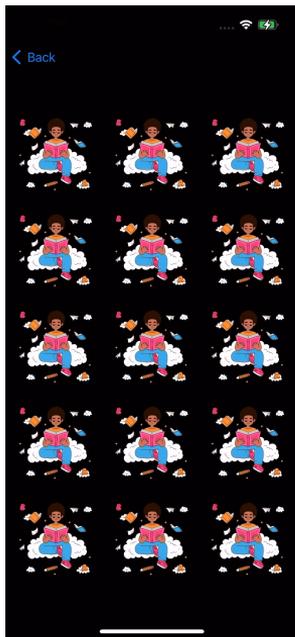
RAM:

51 MB

84 MB



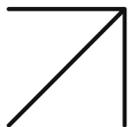
Пример 1



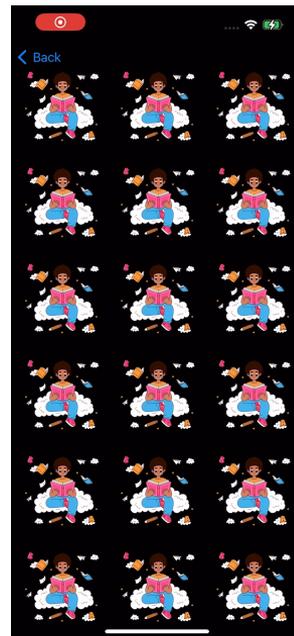
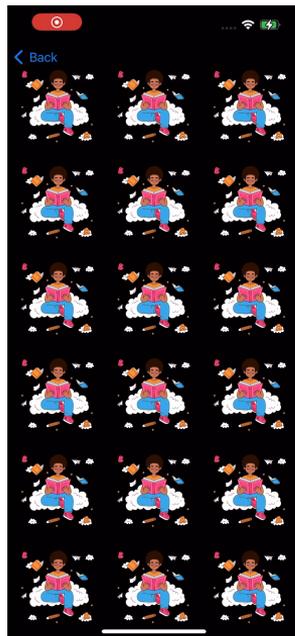
RAM:

57 MB

96 MB



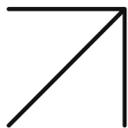
Пример 1



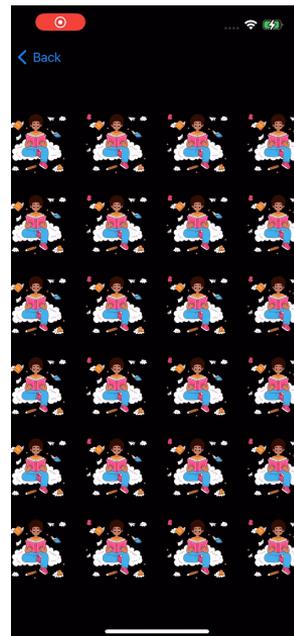
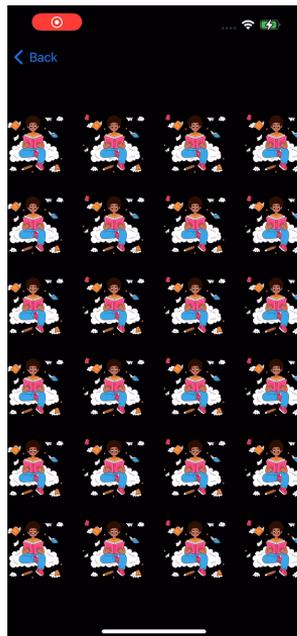
RAM:

65 MB

107 MB



Пример 1



RAM:

80 MB

110 MB

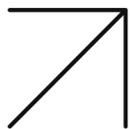


Пример 2

Простые трансформации
позиции элементов

+ изменение path у svg

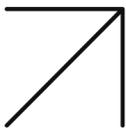




ВЫВОД

Убиваем Lottie?
Тащим Rive?





@PROKOPIEVEVGENI

Спасибо за внимание

P.S. Ловите на площадке,
поговорим про анимации