Segue, Landscape, Touch events, Alpha and User Defined Runtime Attributes

Yutaka Yasuda, Kyoto Sangyo University

目標

・ぱしぱし叩けるゲームを作りたい
 タッチイベントに反応
 火花が出て消える(アニメーション)
 ・ステージを IB の操作だけで簡単に増やせる



Segue での場面進行

Segue

- •場面が変わるだけのアプリを作る
- ボタンをクリックすると指定した View に進む



新規プロジェクト作成

step 1.





Device Family	iPhone	ŧ	
-	🗹 Use Story	boards	
	🗹 Use Auto	matic Refer	ence Counting
	🗹 Include U	nit Tests	

Single View Application

New Project... で

Single View App を選択

プロジェクト名指定時に

Use Storyboards をチェック

今回は iPhone 限定で作った

Storyboard 初期状態



場面の追加 (View Controller の Drag & Drop)



場面移動のためのボタンを設置



そのボタンを Modal でつなぐ



これだけで移動はOK

- ・
 往路だけならこれでOK
- 戻りは要コード



戻るためのコード

ViewController.m

}

// このビューを閉じて一つ前に戻る

- (IBAction)dismissButton:(id)sender {
 [self dismissModalViewControllerAnimated:YES];

遷移を覚えているのだけれど、、、 さてループするような場合はどうするのが?

Navigation View Controller

- 戻りのボタンは自動合成(コードも不要)
- ちょっと操作がトリッキー



 MainStoryboard.storyboa .app on iPhone 5.1 Simulator No Issues 		Editor	View	Drganizer
MainStoryboard.storyboard (En >	🛅 View Controller Scene 👌 🚺 View Controller			m 0
		Custom Cla	ss	
		Class	UIViewControlle	r Ov
	v	User Define	d Runtime Attribu	tes
	💌 💌	y Path T	pe Value	
80				
NEXT				
		5 View(Controller	を変えた
		+		
	い场口は	<i>۲</i> 、		
	1.新しい	いVCグ	フラスを作	≡って
		明た 作 -		亦百
		判`⊂ ト		- 攵丈

はじめから横向けでの起動

横長モードでの起動

常に横長モードで画面を表示するアプリケーションは、このモードで起動するよう、シス テムに対して明示的に指示しなければなりません。通常、iOSアプリケーションは縦長モー ドで起動します。そして、必要に応じてデバイスの向きに合うようにインターフェイスを回 転します。アプリケーションで縦長と横長の両方の向きをサポートする場合は、必ず縦長 モードでビューを設定してから、View Controllerを利用して任意の向きに回転する処理を行 わなければなりません。ただし、アプリケーションが縦長の向きをサポートせず、横長の みをサポートする場合は、最初から横長モードで起動したように見せるために、以下のタ スクを実行しなければなりません。

- アプリケーションのInfo.plistファイルにUIInterfaceOrientationキーを追加し、 このキーの値をUIInterfaceOrientationLandscapeLeftまたは UIInterfaceOrientationLandscapeRightに設定する。
- ●ビューを横長モードにレイアウトし、ビューの自動サイズ変更オプションを確実に正しく設定する。
- View ControllerのshouldAutorotateToInterfaceOrientation:メソッドをオーバーライドし、 横長左向き、または横長右向きに対してはYES、縦長の向きに対してはNOを返す。

iOSアプリケーション プログラミングガイド

https://developer.apple.com/jp/devcenter/ios/library/documentation/iPhoneAppProgrammingGuide.pdf

Info.plist に UlInterfaceOrientation を追加



View を横長モードにレイアウト

- info-plist の Initial Interface Orientation が Landscape
- StoryBoard で ViewController の Orientation (Attributes Inspector) が inferred (推論)
- であれば、自動的に View は横長 (480x320) になる



÷

shouldAutorotateToInterfaceOrientation

ViewController.m

}

- (B00L)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{

```
return ((interfaceOrientation == UIInterfaceOrientationLandscapeLeft) ||
    (interfaceOrientation == UIInterfaceOrientationLandscapeRight));
```

	🗰 🔺 🕨 📐 test4						
▼ 🛃 test4	PROJECT		Summary	Info	Build Settings	Build Phases	Build Rules
v test4	📩 test4	iOS Application Ta	arget				
h AppDelegate.h	TARGETS	Bundle Ident					
MainStoryhoard storyhoard	À test4	Buildle Iden	777	₅≘лф		のだかと	白新加
h ViewController.h	👕 test4Tests	ve	$C \subset C$	家化	している	のだから	、日期処
 WiewController.m Supporting Files 		Deployment Ta	理して	「くれ「	てもいいの	のに!(実験済)
test4-Info.plist InfoPlist.strings		▼ iPhone / iPod I	Deployment info				
m main.m h test4-Prefix.pch		Main Storybo	pard MainStory	board		•	
Gest4 lests Frameworks Products		Main Inter	face			•	
		Supported Devi	ce Orientations				
			Portrait	Upsic	de Landscape n Left	Landscape Right	

タッチイベントの取得

ぱしぱし叩きたい

- ・
 ・
 画面上のどの位置をタッ
 ・
 チしたか知りたい
- ボタンの TouchUpInside
 などでは座標位置の情報
 が足りない
- タッチイベントをそのま
 ま取得



タッチイベントの取得

- touchesBegan イベントがそのまま拾える
- 当該ビューでの座標位置が得られる

ViewController.m

```
- (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event {
    // タッチ開始地点を取得
    CGPoint lastTouchPoint = [[touches anyObject] locationInView:self.view];
    [self punch:lastTouchPoint]; ← 自前のアクションを呼び出すなり
}
```

★が出て消えるアニメーション

ViewController.m

```
- (void)punch:(CGPoint)point {
   // イメージビューを置く
   UIImageView *anImageView = [[UIImageView alloc] initWithImage:flashImage];
   [anImageView setCenter:point];
   [anImageView setAlpha:1.0]; ← 最初は Alpha 値 1.0 で用意
   [anImageView setUserInteractionEnabled:NO];
   [self.view addSubview:anImageView]; ← addSubview で「置く」
   // 徐々に薄くして最後は消す
   [UIView animateWithDuration:0.5f ← 0.5秒かけて、
                  animations:^(void){
                     anImageView.alpha = 0.3; ← Alpha 値 0.3 まで変化させる
                  }
                  completion:^(BOOL finished){
                     [anImageView removeFromSuperview]; ← 終了したら削除
                  }];
}
```

どこかで画像を初期処理などで flashImage プロパティに読み込み、保持しておく

flashImage = [UIImage imageNamed:@"flash01.png"];





ようやくゲーム的なこと

逃げるキャラクターを先回りして追う敵

移動するキャラクターと、どこで遭遇できるかを予想して走る追跡キャラクター



衝突予測:ベクター&三角関数

以下の状況で A, B が衝突する時間 t と、そのために必要な角θを調べる



単純化しました(B原点出発・横移動のみ)

以下の状況で A, B が衝突する時間 t と、そのために必要な角θを調べる





勉強して出直してきます・・・・

いや、我々には Alpha がある



Mathematica (Alpha) で θ について解かせる





検算

試しに下側の式に S, R, X, Y の値を入れて t について解かせてみた。 入力式は以下のとおり:

(-S X+sqrt(R^2 X^2+R^2 Y^2-S^2 Y^2))/((R-S) (R+S)), R=2, S=1, X=5, Y=5



code: 最短合流地点への θ を得る

```
float thetaToMeetFor(float bSpeed, float ax, float ay, float aSpeed)
{
   float theta;
   // 先に計算不能状態を処理する
   // もし A の y がゼロ (最初から B の真横にいた) なら theta は \pi or 0
   if( fabs(ay) < 10.0 ) {
       if(ax>0.0)
           theta = M PI;
       else
           theta = 0.0:
       return theta;
    }
   // 解
   theta = 2.0 * (atan((-1.0 * sqrt((aSpeed * aSpeed * ax * ax) +
                                    (aSpeed * aSpeed * ay * ay) -
                                    (bSpeed * bSpeed * ay * ay)) - ( bSpeed * ax ) )
                       /(ay * (aSpeed + bSpeed))
                  );
    return theta:
}
```

一般解が欲しくて

cos(R) * t * S2 + X2 = cos(r) * t * S1 + X1, sin(R) * t * S2 + Y2 = sin(r) * t * S1 + Y1 for r and t



Input interpretation:



をやらせてみたが、一般解は爆発して無理。 特殊条件のものについては並ぶが、それらではだめ。

レイアウトするだけで新しいステージを作る

レイアウトするだけでステージを作る

- scene を作り Segue で接続(勝つまで遷移ボタンは非表示)
- ・キャラクターはすべて共通のViewで作成
- キャラの性質はそのViewのpropertyとして埋め込み



User Defined Runtime Attributes



奇妙な制限事項			Boolean Number String Localized String					
 ほとんど資料なし 使える形式は右の通り 数は Number のみ、かつ整数 	女のみ	Point Size Rect Range Color Nil						
• Point, Size, Rect, Range は実数を受 け入れるが使うとエラーになる				ode 4 ごは仮	1.3 以ī 吏えた	前 ?		
Cont 👌 💭 View 👌 🔄 Char View – tori.png ٵ 📵 🕨	D		-	1000	0			
✓								
	▼ User Defined Runtime Attributes							
	Key Path	Туре	Valu	le				
	maxSpeed	Size	{1.2,	, 2.299	9999999	9		
	type	Number	; 2					

コード

ViewController.h

```
@property NSMutableArray *charViews;
```

ViewController.m

```
@synthesize charViews;
```

```
// 現在の view の上にある subviews を取り出し、そこから CharView だけを抜き出す
- (void)getCurrentCharViews
{
    NSArray *subViews = [self.view subviews];
    CharView *charView;
    [charViews removeAllObjects]; // キャラの消滅を前提に毎回一旦空にする
    int i;
    for(i=0; i<subViews.count; i++) {
        if([[subViews objectAtIndex:i] isKindOfClass:[CharView class]]) {
            charView = [subViews objectAtIndex:i];
            [charViews addObject:charView];
        }
        isKindOfClass で種別を判定し、該当するものだけ加える
    }
}</pre>
```

おしまい

- Segue での場面進行
- Landscape mode での起動
- touchesBegan の利用
- Mathematica Alpha
- 謎の User Defined Runtime Attributes
- Interface Builder だけで新ステージを作れる!