

Soft Blue Soft Blue Soft Blue Soft Blue Soft Blue

Soft Blue Soft Blue Soft Blue Soft Blue Soft Blue

Soft Blue Soft Blue Soft Blue Soft Blue Soft Blue

iOS

Criando Aplicações para iPhone e iPad

Cocoa Touch


SoftBlue

www.softblue.com.br

Todos os direitos de cópia reservados. Não é permitida a distribuição física ou eletrônica deste material sem a permissão expressa e por escrito do autor.

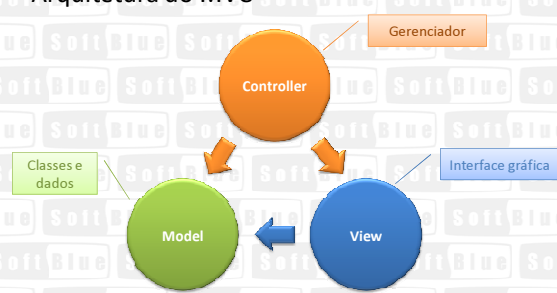
Introdução ao Cocoa Touch

- Framework desenvolvido para iOS
 - Suporte a imagens, animações, 3D, áudio, rede, câmera, acelerômetro, giroscópio, GPS, outros
- Construída em Objective-C (OO)
- Baseada no padrão MVC
 - Model: Classes e dados
 - View: Interface gráfica
 - Controller: Gerenciador / conector
- Abordagem no decorrer do curso
- Garbage Collector melhorado no iOS 5.0



Model View Controller

- Arquitetura do MVC



Outlets e Actions

- Outlets (**IBOutlet**)
 - Variáveis
 - Fácil gerenciamento pelo Interface Builder
 - `[@property([Atributos])] IBOutlet Tipo Variável;`

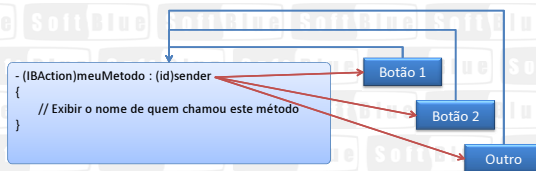
```
@property (nonatomic, retain) IBOutlet UIButton *meuBotao;
```

- Actions (**IBAction**)
 - Métodos (funções)
 - Fácil gerenciamento pelo Interface Builder
 - `-(IBAction) NomeDaAction [:Argumento];`

```
-(IBAction)meuMetodo : (id)sender;
```

(id) sender

- Referência do objeto que invocou a action
- Permite tratamento específico para cada sender
- Evita duplicação de código



@property e @synthesize

- **@property**
 - Criação de propriedades

```
@property (nonatomic, retain) IBOutlet UIButton *meuBotao;
```

```

▪ getter      // @property(getter=getMinhaVar) int minhaVar;
▪ setter      // @property(setter=setMinhaVar) int minhaVar;
▪ readwrite   // @property(readwrite) int minhaVar;
▪ readonly    // @property(readonly) int minhaVar;
▪ assign       // @property(assign) int minhaVar;
▪ retain      // @property(retain) ClasseObj *meuObj;
▪ copy        // @property(copy) ClasseObj *meuObj;
▪ nonatomic   // @property(nonatomic) ClasseObj *meuObj;
```

- **@synthesize**
 - Criação de métodos **setters** e **getters**

Métodos automatizados

- Método **getter** obsoleto

```
-(id)minhaVar {
    return minhaVar;
}
```

- Método **setter** obsoleto

```
-(void)setMinhaVar:(id)minhaVarTemp {
    if(minhaVarTemp != minhaVar) {
        [minhaVarTemp retain];
        [minhaVar release];
        minhaVar = minhaVarTemp;
    }
}
```

Arquivo de configurações

- xxx-Info.plist
 - **Bundle Display Name:** Permite alterar o nome que aparece para o aplicativo na tela dos dispositivos móveis
 - **Icon File:** Permite alterar o ícone que aparece para o aplicativo na tela dos dispositivos móveis

Aulas práticas e manuais on-line



Assista agora às aulas práticas, que abordam os temas tratados nesta aula teórica, para consolidar seus conhecimentos.

[Clique aqui](#) para visualizar as aulas práticas disponíveis