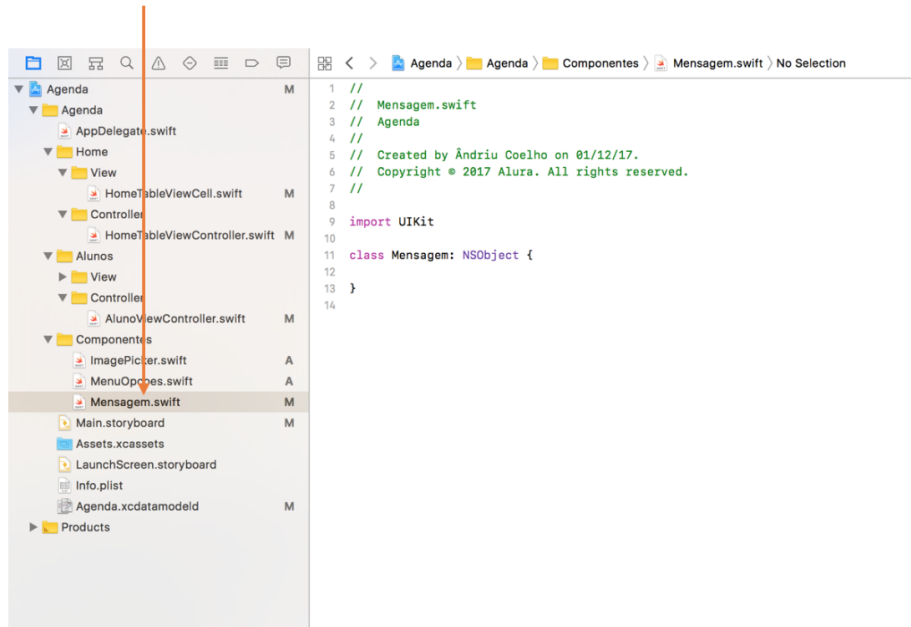


Configurando envio de SMS

Com o Action Sheet implementado, agora podemos continuar a nova feature de envio de SMS.

Vamos criar a classe Mensagem que implementará o protocolo *MFMessageComposeViewControllerDelegate*.



Para exibir o componente de mensagem nativo do iOS, utilizaremos a classe *MFMessageComposeViewController*.

Para ter acesso a essa classe, é necessário implementar o framework *MessageUI*:

```
import UIKit
import MessageUI
```

Com o framework implementado, podemos utilizar o protocolo *MFMessageComposeViewControllerDelegate*.

Esse protocolo nos obriga a utilizar o método `messageComposeViewController`:

```
// MARK: - MessageComposeDelegate

func messageComposeViewController(_ controller: MFMessageComposeViewController, didFinishWith
    controller.dismiss(animated: true, completion: nil)
}
```

Esse método é disparado quando o botão cancelar da tela de mensagem for pressionado.

Vamos criar um método que retorne um *MFMessageComposeViewController* (componente de mensagem)

```
// MARK: - Metodos
```

```
func configuraSMS(_ aluno:Aluno) -> MFMessageComposeViewController? {  
    if MFMessageComposeViewController.canSendText() {  
        let componenteMensagem = MFMessageComposeViewController()  
        guard let numeroDoAluno = aluno.telefone else { return componenteMensagem }  
        componenteMensagem.recipients = [numeroDoAluno]  
        componenteMensagem.messageComposeDelegate = self  
  
        return componenteMensagem  
    }  
    return nil  
}
```

Como o recurso de envio de SMS não funciona no simulador, funciona apenas em devices de verdade, precisamos fazer uma verificação para validar se é possível abrir o componente de SMS. Por isso vamos deixar o retorno como optional. Se for possível criaremos o componente **MessageComposeViewController**, se não for possível, o método retornará nil.

Repare que na assinatura do método passamos o aluno como parâmetro, assim conseguimos configurar o número em que o SMS será enviado através da propriedade **recipients**.

Agora é a sua vez:

- Crie uma constante do tipo **Mensagem()** e implemente sua chamada no **case** de SMS do switch