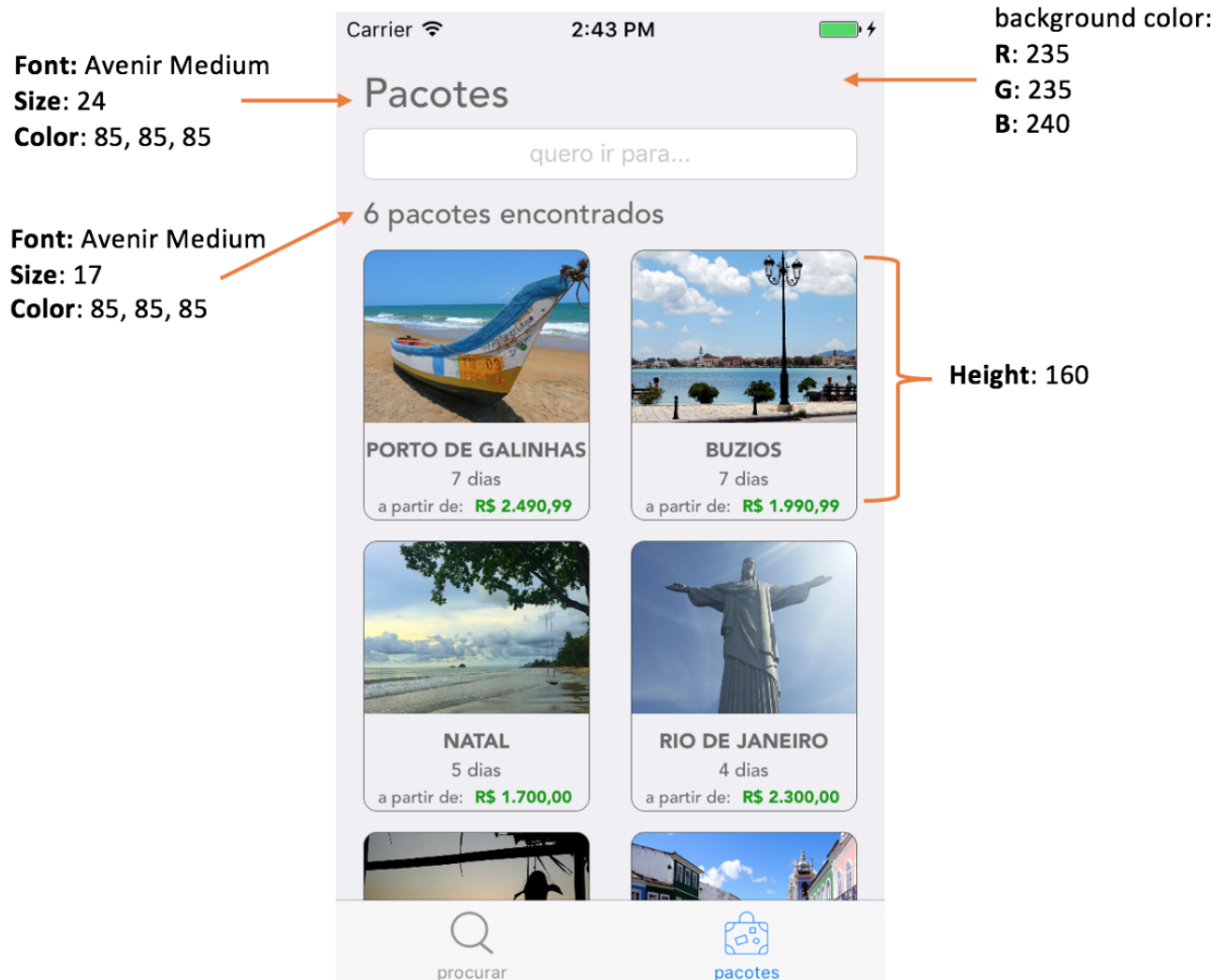


## Implementando UICollectionView

De volta ao projeto, vamos dar uma olhada novamente no layout da tela de pacotes de viagens que temos que implementar:



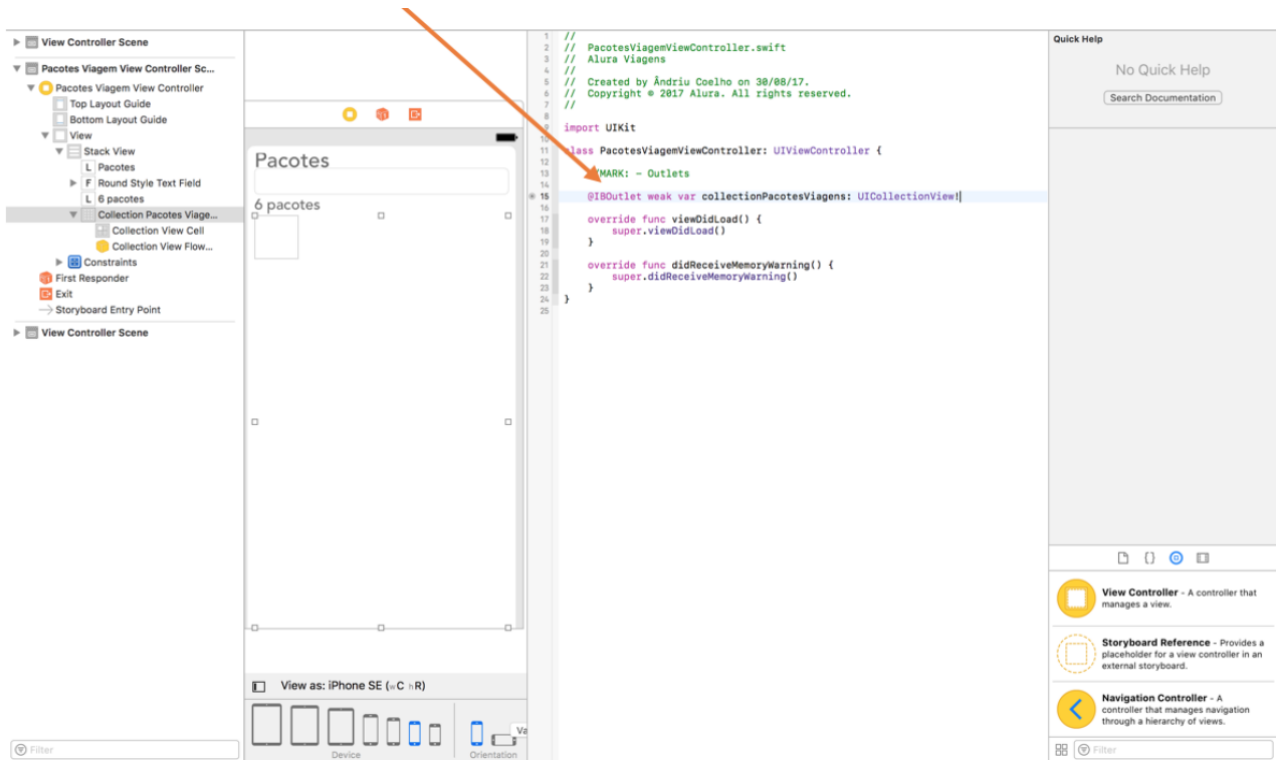
Vamos abrir o storyboard e começar a estilizar os elementos conforme o gabarito.

Repare que até agora colocamos uma 'UICollectionView' no nosso ViewController e ainda não utilizamos. Agora é hora de usá-la.

Antes de começar a implementar, precisamos criar um arquivo para controlar todos os elementos dessa nova tela de pacotes que criamos.

Agora é sua vez: Primeiro crie um novo ViewController com o nome: PacotesViagensViewController e em seguida vamos setar o arquivo no storyboard

Com o ViewController configurado, vamos continuar a implementação da 'CollectionView'. Para isso, precisamos criar seu outlet:



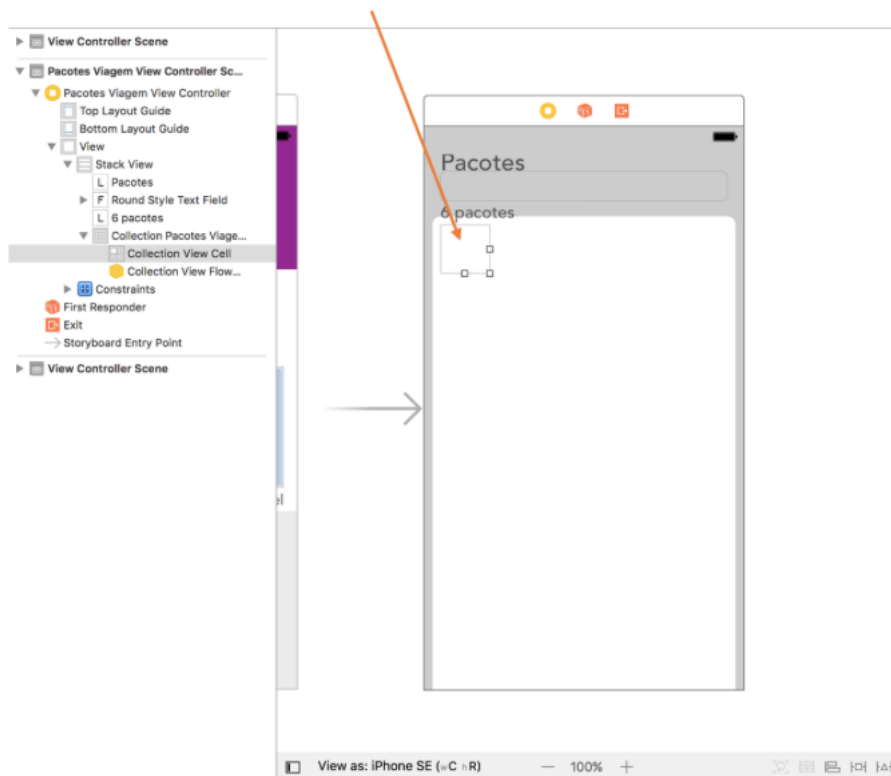
Assim como na TableView implementamos os métodos de 'DataSource' (numberOfRowsInSection, cellForRow) para configurar a quantidade de linhas e a criação da célula, na CollectionView também teremos que implementar seus métodos. Vamos começar implementando o protocolo de DataSource:

```
class PacotesViagensViewController: UIViewController, UICollectionViewDataSource {
```

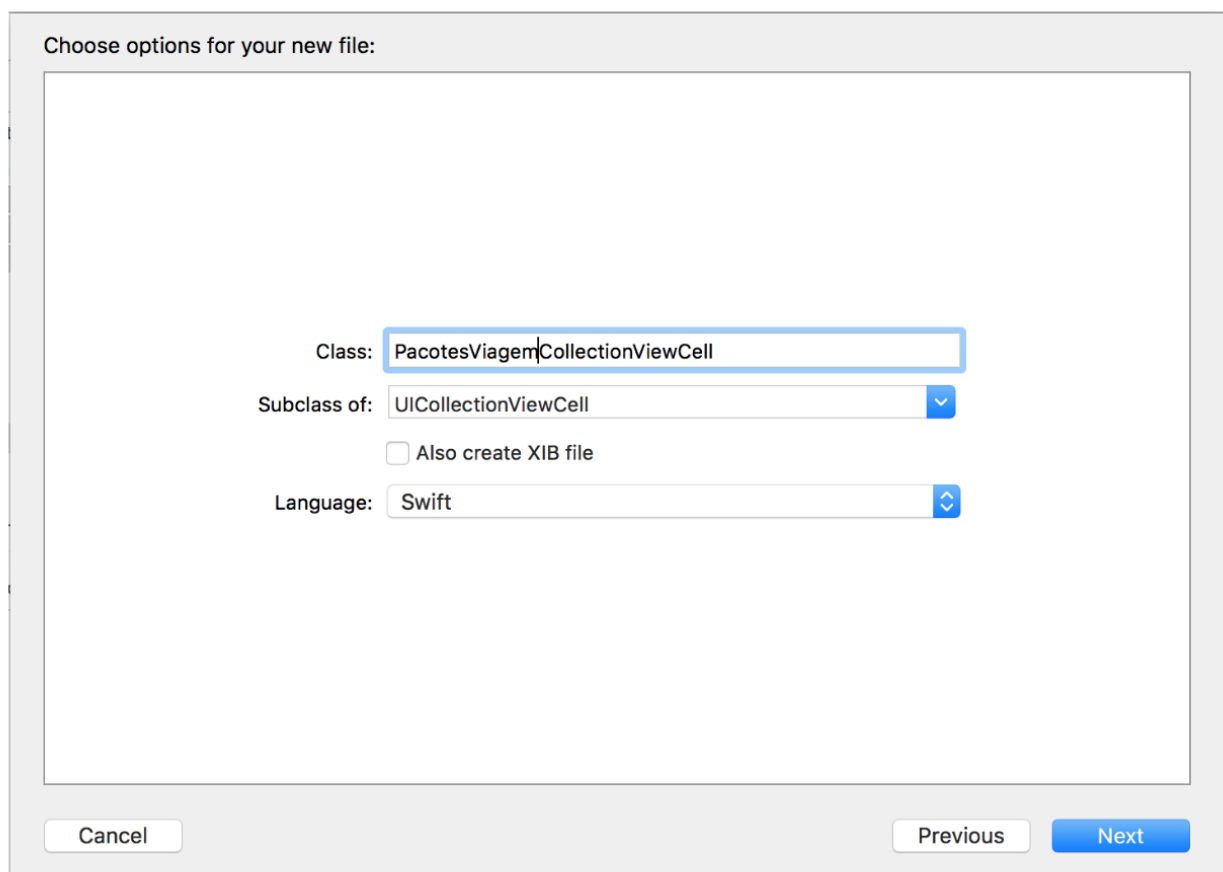
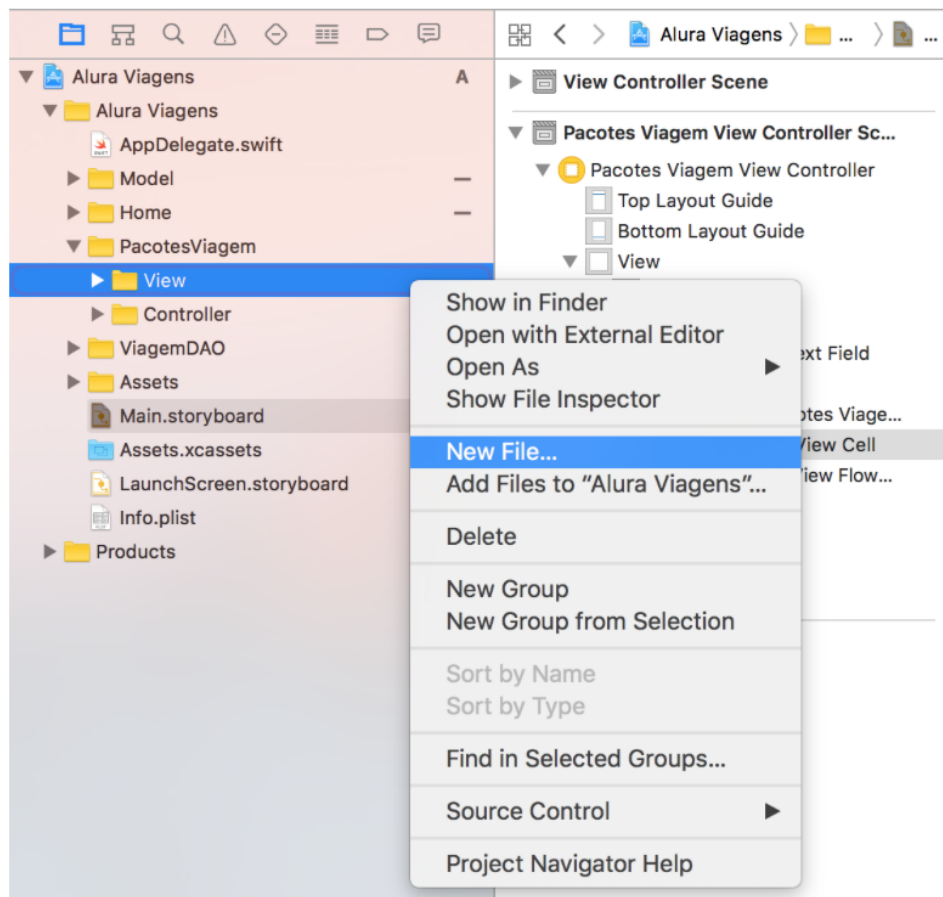
Agora é sua vez: Implemente os métodos 'numberOfItems' retornando um valor fixo de 5 itens para testarmos e também o método 'cellForItemsAtIndexPath'

Agora que já implementamos os métodos, precisamos retornar uma célula no método 'cellForItem'. Quando utilizamos uma collection, por default ela já vem com uma célula:

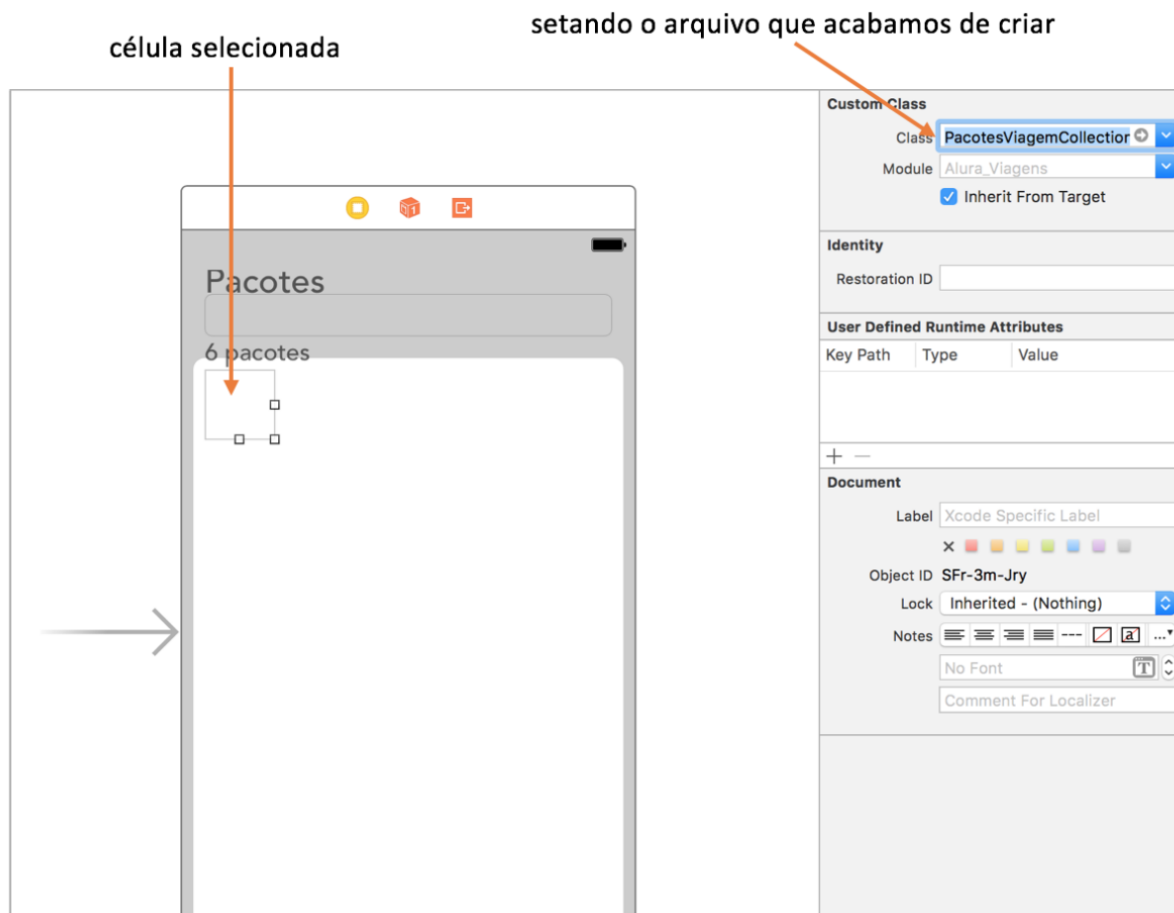
## célula da collectionView



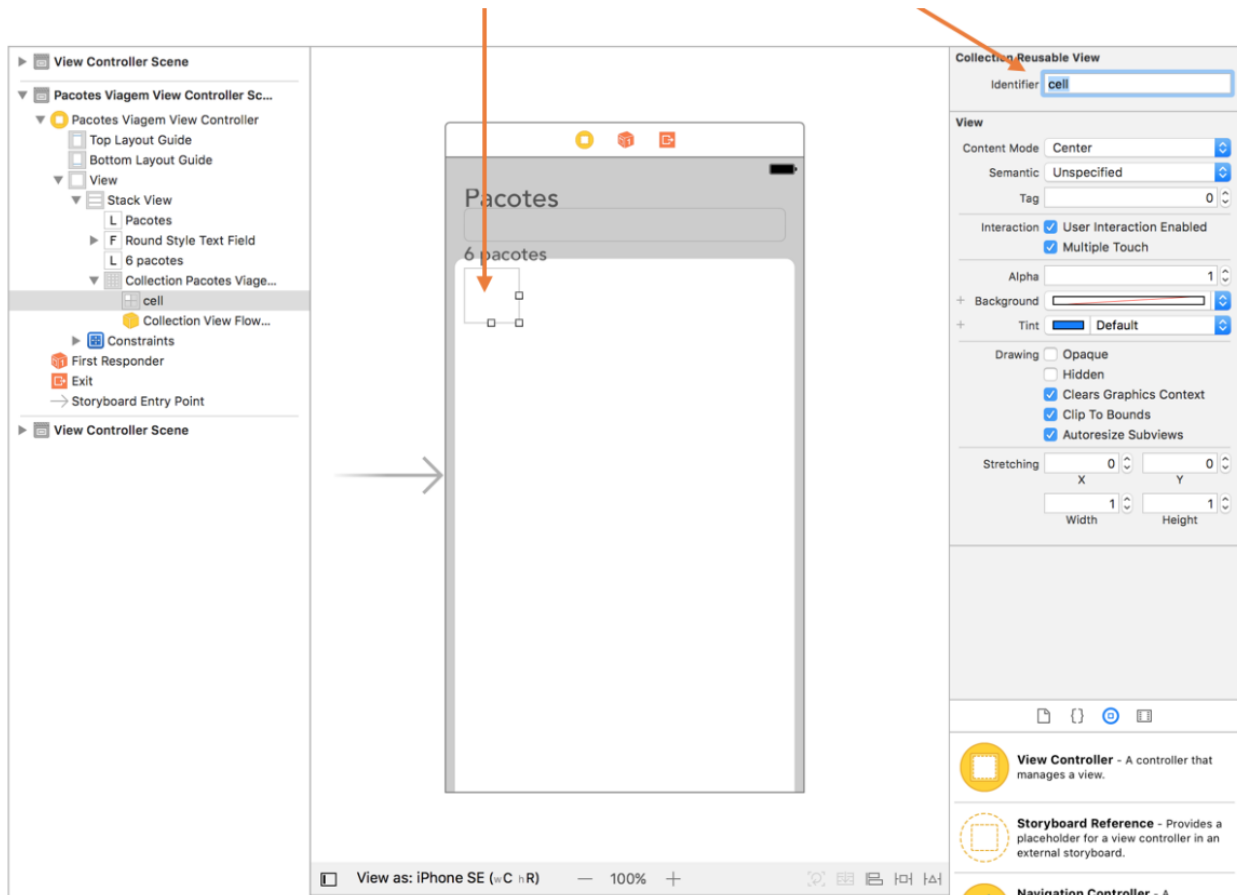
Então o que precisamos é criar um arquivo para controlá-la. Dentro da pasta 'Views' vamos criar um novo arquivo subclasse de 'UICollectionViewCell'



Pronto, já temos um arquivo para controlar a célula da collection. Vamos voltar no storyboard e setar esse arquivo na célula da collection:



Para finalizar as configurações da célula, vamos colocar um identificador para conseguirmos resgatar a célula no método 'cellForItem':



Pronto! Com isso já temos todas as configurações necessárias para criarmos a célula da UICollectionView.

Vamos voltar no ViewController e terminar a implementação da collection:

```
func collectionView(_ collectionView: UICollectionView, cellForItemAt indexPath: IndexPath) -> UICollectionViewCell {
    let celulaPacote = collectionView.dequeueReusableCell(withReuseIdentifier: "cell", for: indexPath)

    return celulaPacote
}
```

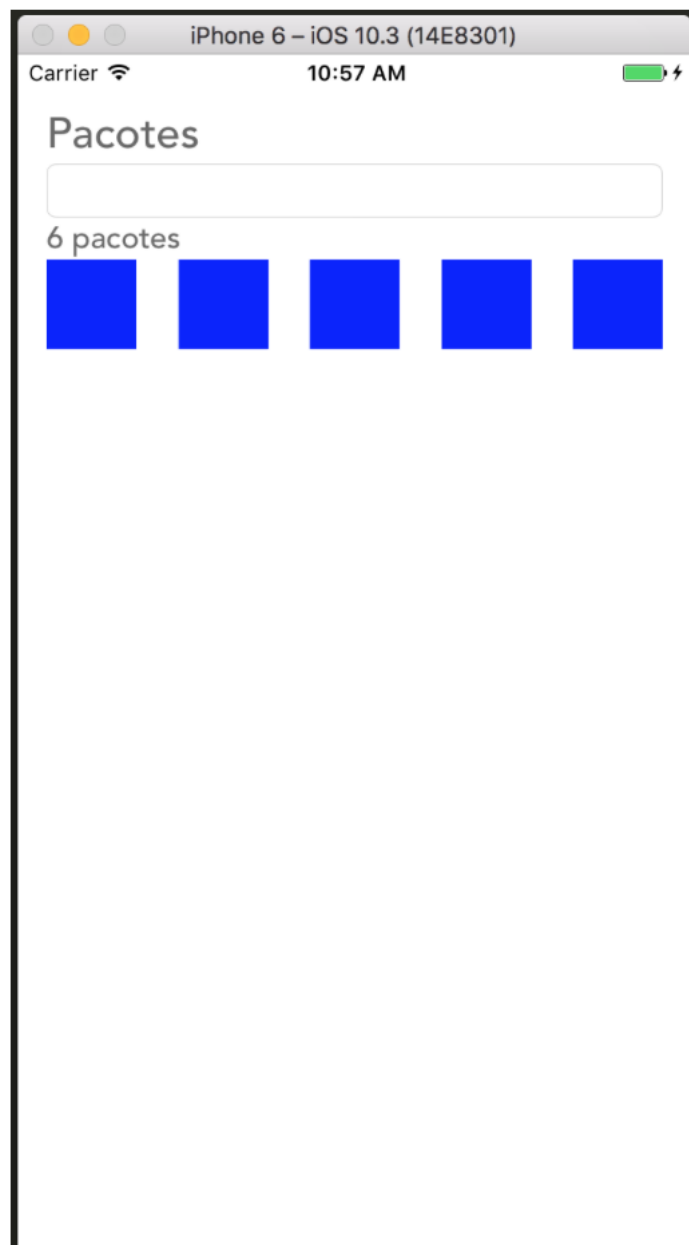
Vamos setar o 'DataSource' da CollectionView no ViewController:

```
override func viewDidLoad() {
    super.viewDidLoad()
    self.collectionPacotesViagens.dataSource = self
}
```

Para testar se a nossa implementação funcionou, vamos mudar o background da célula para azul e rodar o app:

```
func collectionView(_ collectionView: UICollectionView, cellForItemAt indexPath: IndexPath) -> UICollectionViewCell {
    let celulaPacote = collectionView.dequeueReusableCell(withReuseIdentifier: "cell", for: indexPath)
    celulaPacote.backgroundColor = UIColor.blue

    return celulaPacote
}
```



Ótimo, a implementação já está funcionando. Os próximos passos são seguir o layout do app para deixar igual ao proposto e implementar os pacotes de viagem na UICollectionView.