

### How to use **DGMenuView**:

- The only thing your **UIViewController** should do is to inherit from **DGViewController**.

You import the class as `"#import "DGViewController.h"`, and just inherit:

`@interface MyClassViewController : DGViewController`

---

### THATS IT!!!

You get a built in left navigation button, and the menu, with all of it's functionality.  
In case you want to mess around with the button just go to:

- `(void)configureMenuButton;`

under **DGViewController**, and customize the menuButton over there.

---

### THE MENU

**DGMenuView** is the actual menu view sliding out, it's customization may be done in:

- `(void)initMenuFunctionality;`

---

### THE MENU ITEMS

For the menu items, there is a special class called **DGMenuViewDataSource**.  
Notice you create all the objects in the:

- `(id)init;`

function, and the rest of the file is the Menu's tableView delegate. you can customize the header looks, footer, number of rows, actions, etc....

**NOTICE THE COMMENTED OUT ACTIONS IN THE didSelectRow FUNCTION, THEY ARE MEANT TO BE UNCOMMENTED ONLY IF YOU USE Three20 NAVIGATOR.**

---

### THE MENU ITEM

Every item in the menu consists of couple of properties.  
They are all can be seen in **DGMenuViewItem**.

The most important thing to remember when creating them is:

- If you are NOT using Three20, you should decide which action should be performed when the item is selected from the menu. this is done in the:

- `(id)initWithRunBlock:(void (^)(void))block;`

method.

See **DGMenuViewDataSource** for an example on how to use these blocks.

Built in for free is the notifications functionality, which will pop up an awesome animated icon when the notificationsBadgeCount is set to be not 0.

---

### THE MENU ITEM CELL

**DGMenuItemCell** is in charge of every cell in the menu.  
It has the:

```
- (id)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString *)reuseIdentifier
```

method, to customize the UI of the cell,  
and:

```
- (void)postSetSubject;
```

method, which is triggered once the cell's DGMenuViewItem is set. this is the place to control dynamic data such as title, image path, notifications counter, etc...

**Happy Coding!**  
**Daniel**