Отчет по Домашнему заданию по курсу
"Разработка Интернет-Приложений"

Выполнил:
Студент группы
ИУ5-55Б
Шкарин Егор Алексеевич

Москва, МГТУ – 2021

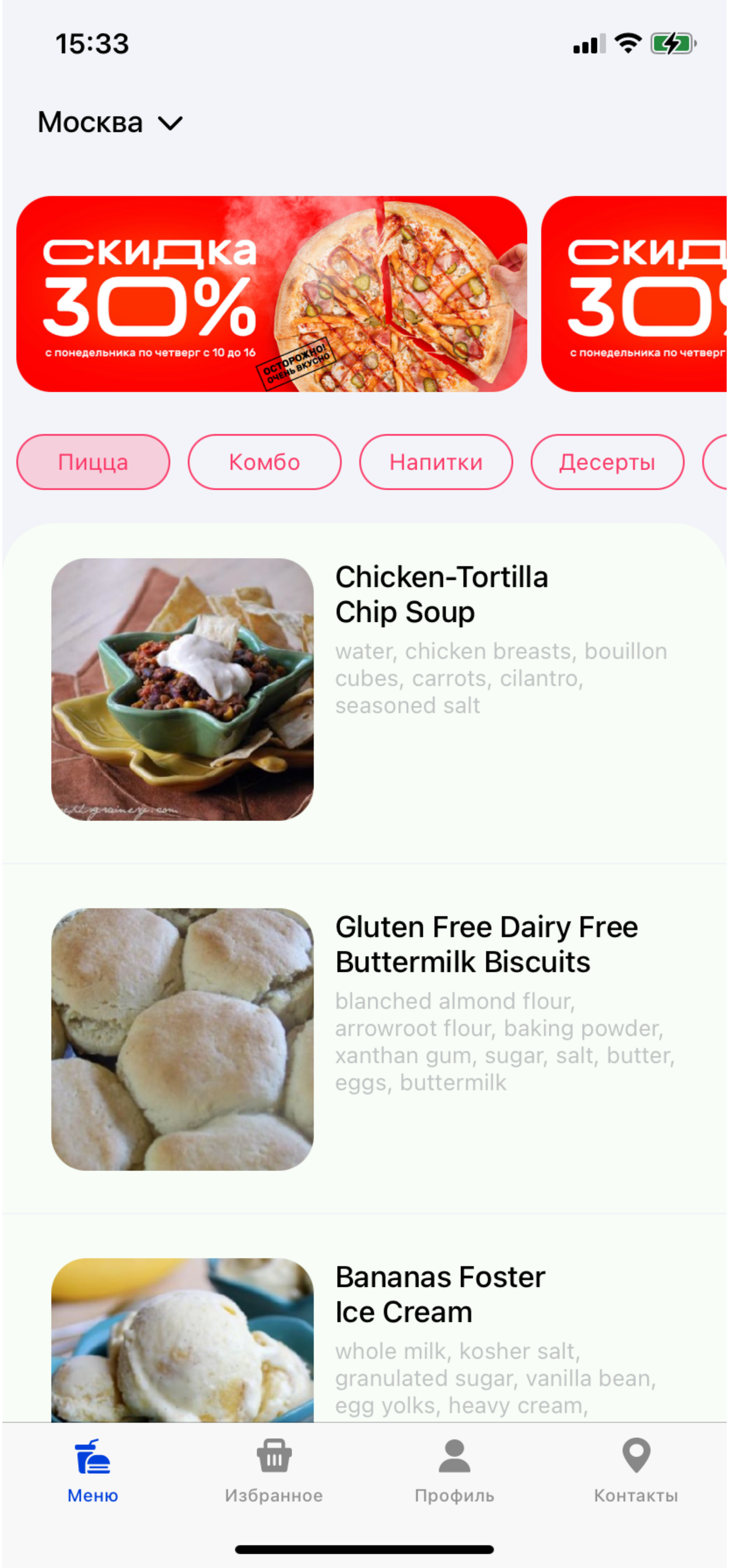
Задание:

На основе методических указаний разработайте мобильное приложение на языке Kotlin (платформа Android) или языке Swift (платформа iOS). Для создания приложения необходимо решить следующие задачи:

Создать стартовый проект мобильного приложения.
Добавить компонент таблицы в Ваше приложение.
Добавить подключение из мобильного приложения к веб-сервису, разработанному в лабораторной работе №6.
Наполнить таблицу элементами, полученными из вашего веб-сервиса: с помощьюзапросов к API получить из него данные.
Разработать дополнительный экран с UI компонентами для отображения детальной информации об элементе таблицы при касании по нему.

Данное домашнее задание было выполнено с небольшими отклонениями от самого задания. В частности: вместо архитектуры MVC была взята архитектура VIPER, работа с была полностью со стороннем API, а верстка производилась кодом. Также были установлены дополнительные зависимости в виде фреймворка KingFisher.

Результат работы программы



Chicken-Tortilla Chip Soup

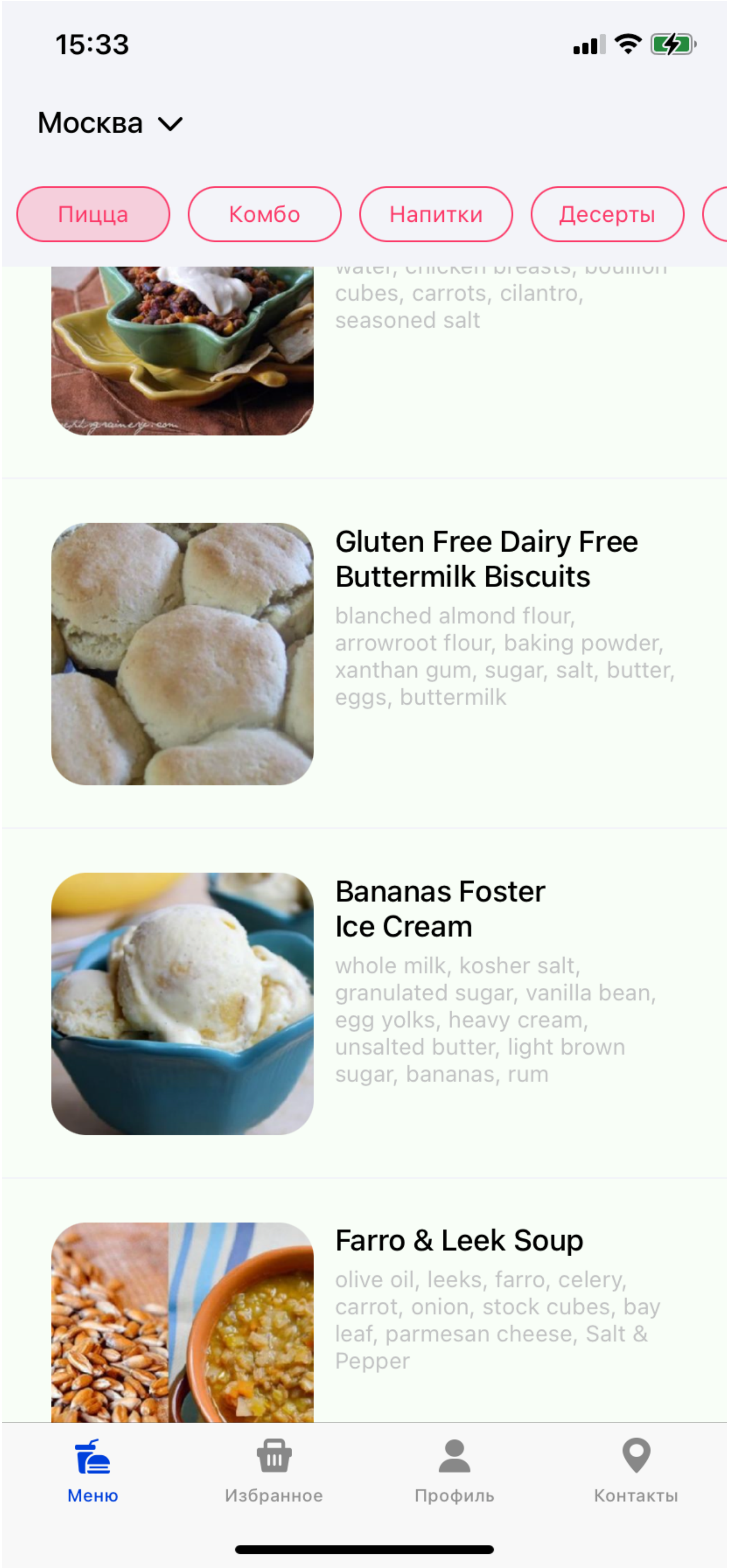
water, chicken breasts, bouillon cubes, carrots, cilantro, seasoned salt

Gluten Free Dairy Free Buttermilk Biscuits

blanched almond flour, arrowroot flour, baking powder, xanthan gum, sugar, salt, butter, eggs, buttermilk

Bananas Foster Ice Cream

whole milk, kosher salt, granulated sugar, vanilla bean, egg yolks, heavy cream,



water, chicken breasts, bouillon cubes, carrots, cilantro, seasoned salt

Gluten Free Dairy Free Buttermilk Biscuits

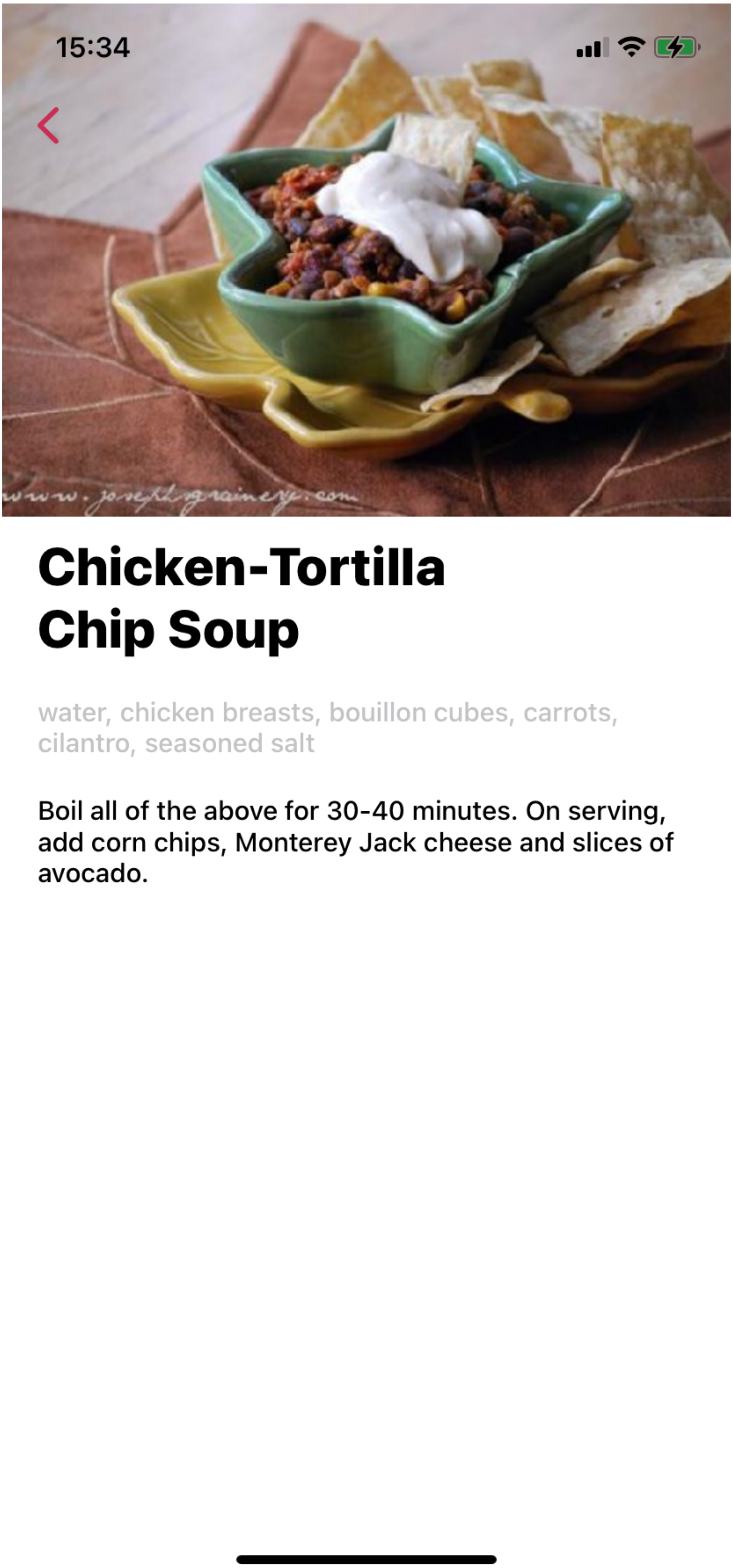
blanched almond flour, arrowroot flour, baking powder, xanthan gum, sugar, salt, butter, eggs, buttermilk

Bananas Foster Ice Cream

whole milk, kosher salt, granulated sugar, vanilla bean, egg yolks, heavy cream, unsalted butter, light brown sugar, bananas, rum

Farro & Leek Soup

olive oil, leeks, farro, celery, carrot, onion, stock cubes, bay leaf, parmesan cheese, Salt & Pepper



Chicken-Tortilla Chip Soup

water, chicken breasts, bouillon cubes, carrots, cilantro, seasoned salt

Boil all of the above for 30-40 minutes. On serving, add corn chips, Monterey Jack cheese and slices of avocado.

Код программы:

SceneDelegate.swift

```
//
//  SceneDelegate.swift
//  test
//
//  Created by Егор Шкарин on 05.01.2022.
//

import

class SceneDelegate  UIResponder  UIWindowSceneDelegate

    var window  UIWindow
    var coordinator  CoordinatorProtocol

    func scene _        UIScene  willConnectTo          UISceneSession
options                    UIScene
        guard let                      as  UIWindowScene  else   return
        window   UIWindow windowScene
        guard let          window else   return
               overrideUserInterfaceStyle    light
        coordinator   Coordinator window
        coordinator  start


    func sceneDidDisconnect _        UIScene
        // Called as the scene is being released by the system.
        // This occurs shortly after the scene enters the background, or when
its session is discarded.
        // Release any resources associated with this scene that can be re-
created the next time the scene connects.
        // The scene may re-connect later, as its session was not necessarily
discarded (see `application:didDiscardSceneSessions` instead).


    func sceneDidBecomeActive _        UIScene
        // Called when the scene has moved from an inactive state to an active
state.
        // Use this method to restart any tasks that were paused (or not yet
started) when the scene was inactive.


    func sceneWillResignActive _        UIScene
        // Called when the scene will move from an active state to an inactive
state.
        // This may occur due to temporary interruptions (ex. an incoming
phone call).


    func sceneWillEnterForeground _        UIScene
        // Called as the scene transitions from the background to the
foreground.
```

```
        // Use this method to undo the changes made on entering the
background.


    func sceneDidEnterBackground _        UIScene
        // Called as the scene transitions from the foreground to the
background.
        // Use this method to save data, release shared resources, and store
enough scene-specific state information
        // to restore the scene back to its current state.
```

Coordinator.swift

```
//
//  Coordinator.swift
//  test
//
//  Created by Егор Шкарин on 05.01.2022.
//

import
import

protocol CoordinatorProtocol
    func start


final class Coordinator  CoordinatorProtocol
    private let window  UIWindow
    private lazy var tabBarController   UITabBarController
    private lazy var navigationControllers
Coordinator makeNavigationControllers
    init window  UIWindow
        self window


    func start
        setupMenu
        setupContacts
        setupProfile
        setupBasket

        let                         NavControllerType allCases compactMap
            self navigationControllers

        tabBarController setViewControllers                        animated
true
        tabBarController tabBar barTintColor   UIColor named  "tabBarColor"
        window rootViewController   tabBarController
        window makeKeyAndVisible


extension Coordinator

    private func setupMenu
        guard let                 navigationControllers  menu  else
            fatalError "No navController"

        let               MenuContext moduleOutput  nil
        let                 MenuContainer assemble with
                      setViewControllers                viewController
animated  true


    private func setupBasket
        guard let                 navigationControllers  basket  else
            fatalError "No navController"

        let                 BasketContext moduleOutput  nil
        let                   BasketContainer assemble with
                      setViewControllers                  viewController
animated  true


    private func setupProfile
        guard let                 navigationControllers  profile  else
            fatalError "No navController"

        let                  ProfileContext moduleOutput  nil
        let                    ProfileContainer assemble with
                      setViewControllers                   viewController
animated  true


    private func setupContacts
        guard let                 navigationControllers  contacts  else
            fatalError "No navController"

        let                   ContactsContext moduleOutput  nil
        let                    ContactsContainer assemble with

                      setViewControllers                   viewController
animated  true


    fileprivate static func makeNavigationControllers       NavControllerType
UINavigationController
        var          NavControllerType  UINavigationController
        NavControllerType allCases forEach                     in
            let                        UINavigationController
            let              UITabBarItem title
                                          image
                                          tag                     rawValue
                                 tabBarItem


                                 isNavigationBarHidden   true

        return



fileprivate enum NavControllerType  Int  CaseIterable
    case menu  basket  profile  contacts

    var title  String
        switch self
        case  menu
            return "Меню"
        case  basket
            return "Избранное"
        case  profile
            return "Профиль"
        case  contacts
            return "Контакты"



    var image  UIImage
        switch self
        case  menu
            return UIImage named  "menu"
        case  basket
            return UIImage named  "basket"
        case  profile
            return UIImage named  "profile"
        case  contacts
            return UIImage named  "contacts"
```

NetworkManager.swift

```
//
//  NetworkManager.swift
//  test
//
//  Created by Егор Шкарин on 06.01.2022.
//

import

protocol NetworkManagerProtocol
```

```
final class NetworkManager
    func getData completion  @escaping  Result Data  Error
        guard let       URL string
"https://api.spoonacular.com/recipes/random?number=50&apiKey=f6b4e0c5cc9842728
3e493660a3e3d4c"  else   return
        let           URLRequest url
        URLSession shared dataTask with                                       in
            if let
                            failure

            guard let
            else
                return

                        success
          resume
```

KingFisherManager.swift

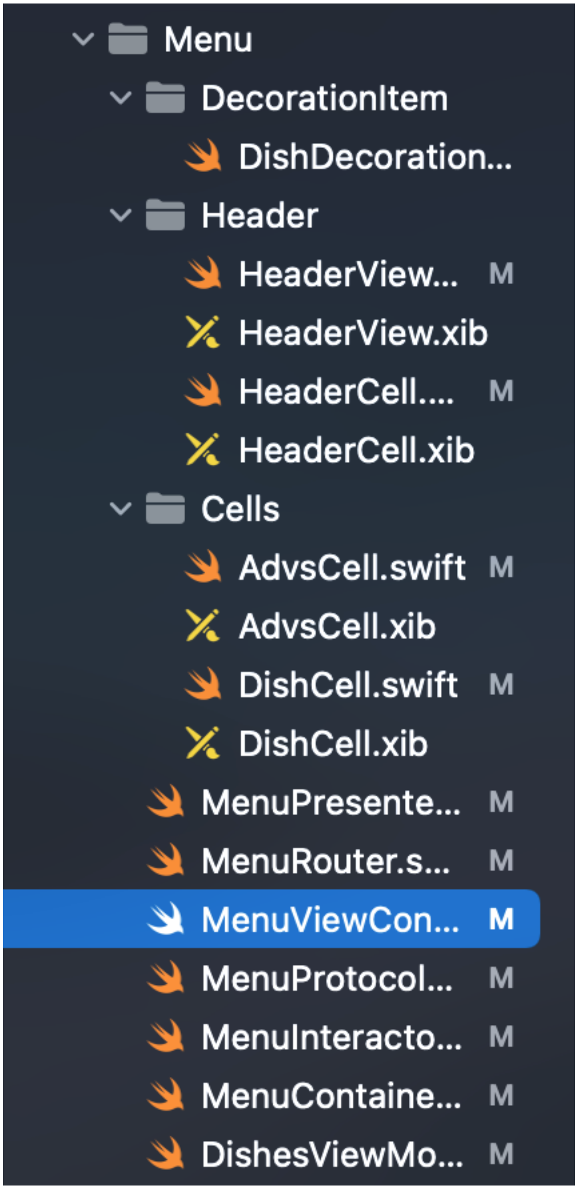
```
//
//  NetworkManager.swift
//  test
//
//  Created by Егор Шкарин on 06.01.2022.
//

import

protocol NetworkManagerProtocol



final class NetworkManager
    func getData completion  @escaping  Result Data  Error
        guard let       URL string
"https://api.spoonacular.com/recipes/random?number=50&apiKey=f6b4e0c5cc9842728
3e493660a3e3d4c"  else   return
        let           URLRequest url
        URLSession shared dataTask with                                       in
            if let
                            failure

            guard let
            else
                return

                        success
          resume
```

ReuseIdentifier.swift

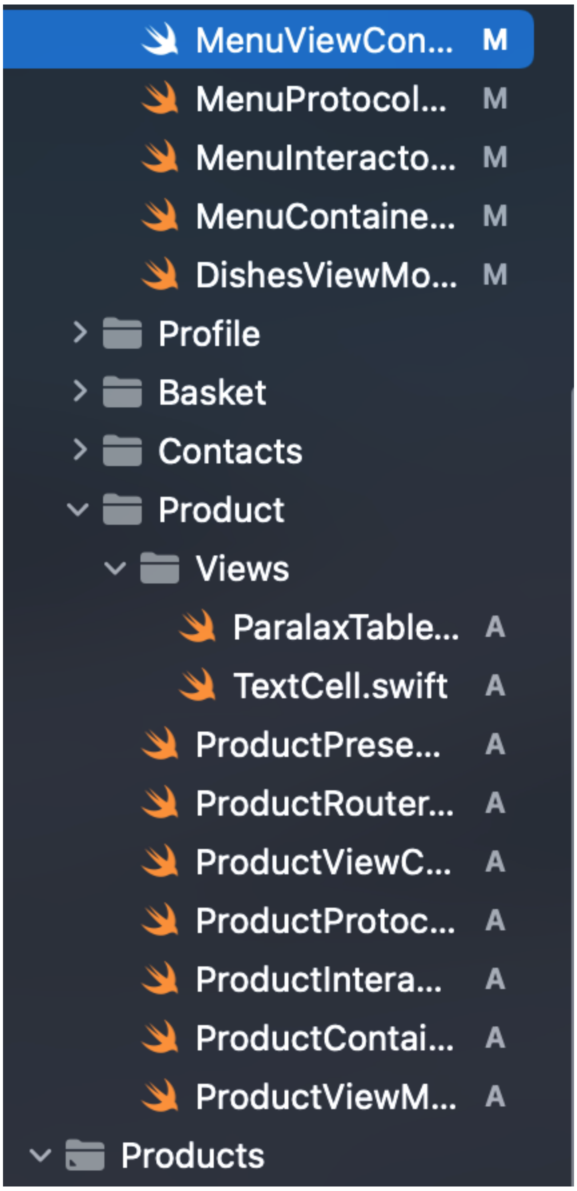
```
//
//  ReuseIdentifire.swift
//  test
//
//  Created by Егор Шкарин on 11.01.2022.
//

import

protocol ReuseIdentifiable
    static var reuseIdentifier  String   get


extension ReuseIdentifiable
    static var reuseIdentifier  String
        return String describing  self

    static var nibName  String
        return String describing  self



extension UITableViewCell  ReuseIdentifiable

extension UITableViewHeaderFooterView  ReuseIdentifiable

extension UICollectionReusableView  ReuseIdentifiable

extension UICollectionView
    func dequeueCell    UICollectionViewCell  cellType    Type  for
IndexPath
        guard let        self dequeueReusableCell withReuseIdentifier
  reuseIdentifier
                for             as  T else
            fatalError "Can't deque"

        return



extension UITableView
    func dequeueCell    UITableViewCell  cellType    Type  for
IndexPath
        guard let        self dequeueReusableCell withIdentifier
  reuseIdentifier
                for             as  T else
            fatalError "Can't deque"

        return
```

```
func register    UITableViewCell  _             Type
    self register           forCellReuseIdentifier
     reuseIdentifier
```

Последующие файлы копировать очень затруднительно, поэтому я приложу скрины из иерархии проекта.



```
Menu
  DecorationItem
    DishDecoration...
  Header
    HeaderView...       M
    HeaderView.xib
    HeaderCell....      M
    HeaderCell.xib
  Cells
    AdvsCell.swift      M
    AdvsCell.xib
    DishCell.swift      M
    DishCell.xib
  MenuPresente...       M
  MenuRouter.s...       M
  MenuViewCon...        M
  MenuProtocol...       M
  MenuInteracto...      M
  MenuContaine...       M
  DishesViewMo...       M
```



```
        MenuViewCon...    M
        MenuProtocol...   M
        MenuInteracto...  M
        MenuContaine...   M
        DishesViewMo...   M
  > Profile
  > Basket
  > Contacts
  v Product
    v Views
          ParalaxTable... A
          TextCell.swift  A
        ProductPrese...   A
        ProductRouter...  A
        ProductViewC...   A
        ProductProtoc...  A
        ProductIntera...  A
        ProductContai...  A
        ProductViewM...   A
v Products
```