

# The Design and Implementation of tourism APP Based on X-Code

Shi-jun Yang<sup>1, a</sup> and Ning Wei<sup>1, b</sup>

<sup>1</sup>College of Computer and Information Technology ,China Three Gorges University

Yichang, Hubei ,China,443002

<sup>a</sup>yangsjun@163.com, <sup>b</sup>16660020@qq.com

**Keywords:** Object-C language; The mobile Internet; Tourism; X-code

**Abstract:** This topic mainly explains the application about travel through X-code development platform. The APP provide some advice to users.The main content of this article explain how to implement the application. The works studied different user require such as framework, function and the final rendering. The application also shared social media including third-party login and sharing.

## 1 Introduction

With the rapid development of intelligent terminals and mobile networks, the fusion of traditional tourism industry and the mobile Internet industry accelerated dramatically. Moving your finger, you can always grasp the latest travel information, travel guides, scenic spots, and real-time query, buy air tickets, hotels, tickets and other services[1].

With the popularity of mobile apps, mobile apps is becoming more and more mature, and the concept of "budget travel" advocated by young people is becoming more and more popular.The APP is easy to operate, the interface is simple and the function is complete. It will provide tourists with a great deal of information and advice on tourism.For developers, apple's APP store is relatively standard, and its market management is strictly regulated, and apple's APP review for APP developers is strict and serious compared with android.In apple store,APP quality can really be guaranteed.

## 2 Develop technology and environment

We use the object-c language to develop the APP.Object-c is characterized by object-oriented, inheritance, encapsulation and polymorphous.it can automatically recycle objects (ARC)

The development of apple's APP is focused on the x-code platform, which supports compiling and running, as well as testing software performance and memory consumption.

In the Simulator, we view the results of the program.Simulators are not able to locate, voice and photograph, but you can use the iPhone to run the installer.

## 3 System module design

This APP is aimed at iPhone users.Novelty, fun, cheap or activity discounts are more attractive to young people.It is divided into four parts[2]: recommendation, destination, community and mine.Demand function module is shown as Fig.1.

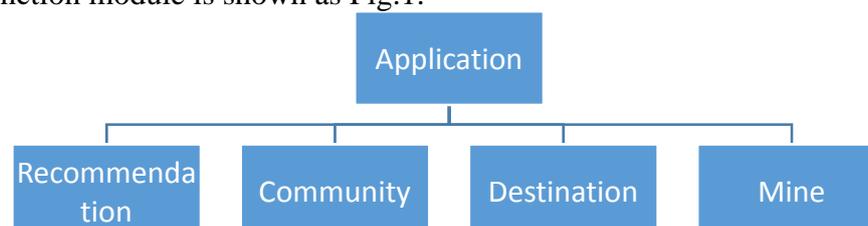


Fig.1. Demand function module

The main contents of the front page are recommended to broadcast, to look at the book, to break the discount, to book the hotel, to find the next stop, to get a discount and to watch the popular travel notes. It is mainly to recommend some popular posts, popular tourist destinations and discount routes[3][4]. All essence is condensed to the homepage, give the user the fastest and most convenient browsing. Recommended function module is shown as Fig.2.

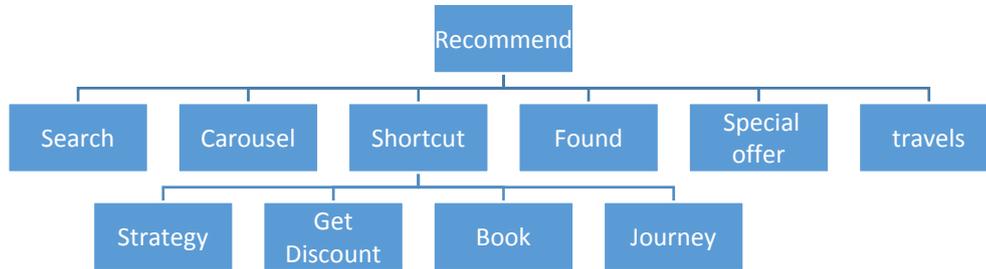


Fig.2. Recommended function module

The main function of the community is to share where you have been, to share experiences, or to travel together. The posts in BBS are classified by place names or other classification. The text is divided into three modules: all, latest and essence. We can click the corresponding module button to display the corresponding contents and display them in the form of lists.

Upper part of plate "my page" shows the avatars and nickname, if they are not logged in, click the image into the login screen, fill the account and password in the middle of login screen, or select the third-party login below, sina, WeChat and QQ. My function module is shown as Fig.3.

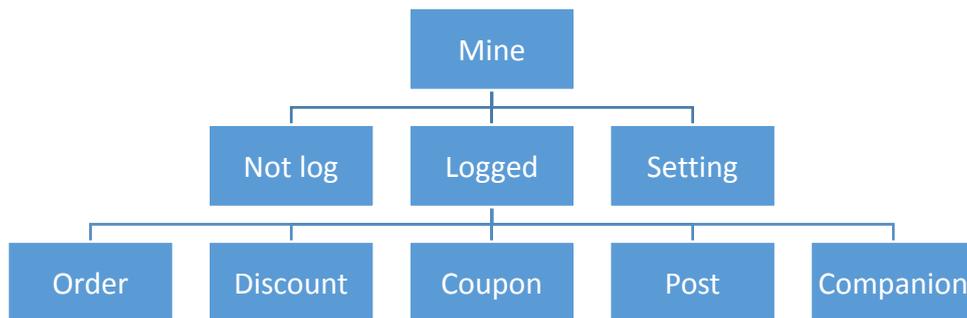


Fig.3. My function module

#### 4 The Implementation of tourism APP

Interface layout relates to how the interface elements are rendered in the APP. Your layout lets users know which things are most important, what they can do and the relationship between them. MVC relationship structure is shown as Fig.4.

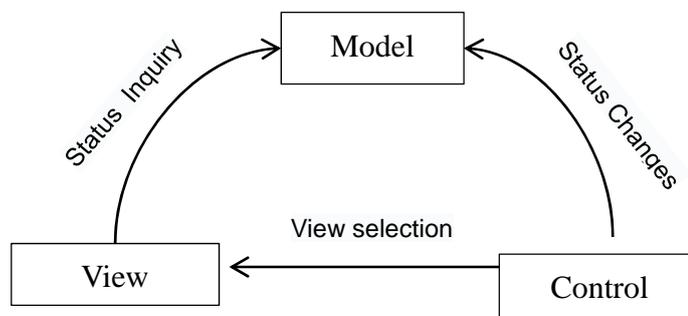


Fig.4. MVC relationship structure

When developing an APP, we often use many third-party open source libraries, such as AFNetworking, JSONKit, etc. Here, we use CocoaPods to import third-party libraries. I found ruby technology to be the most appropriate[5].

There are plenty of social media in the APP. We integrate social media and user-sharing content into the APP, so we can give users a way. Customers can add information without having to exit the APP.

The BasicTabController inherits from the system's UITabBarController, and the primary role of this class is to customize the TAB bar to implement the TAB bar in this class[6].

BasicViewController class inherits from UIViewController class in the system, the main purpose of the class is to encapsulate all classes which inherited from BasicViewController, added some basic common types, and create a Button, for example, to create a label, data analytical tools and so on. TAB bar module core code is shown as Fig.5.

```

- (BasicTabController *)createTab
{
    RecommendViewController *vc1 = [[RecommendViewController alloc] init];
    DestinationViewController *vc2 = [[DestinationViewController alloc] init];
    CommunityViewController *vc3 = [[CommunityViewController alloc] init];
    MineViewController *vc4 = [[MineViewController alloc] init];
    vc1.tabBarItem.image = imageRender(@"1-1");
    vc1.tabBarItem.selectedImage = imageRender(@"1");
    vc2.tabBarItem.image = imageRender(@"2");
    vc2.tabBarItem.selectedImage = imageRender(@"2-1");
    vc3.tabBarItem.image = imageRender(@"3");
    vc3.tabBarItem.selectedImage = imageRender(@"3-1");
    vc4.tabBarItem.image = imageRender(@"4-1");
    vc4.tabBarItem.selectedImage = imageRender(@"4");

    UINavigationController *nvc1 = [[UINavigationController alloc] initWithRootViewController:vc1];
    UINavigationController *nvc2 = [[UINavigationController alloc] initWithRootViewController:vc2];
    UINavigationController *nvc3 = [[UINavigationController alloc] initWithRootViewController:vc3];
    UINavigationController *nvc4 = [[UINavigationController alloc] initWithRootViewController:vc4];

    BasicTabController *main = [[BasicTabController alloc] init];
    main.viewControllers = @[nvc1, nvc2, nvc3, nvc4];
    return main;
}

```

Fig.5. TAB bar module core code

All data comes from the APP background server[7]. After the request, the data is analyzed by Model, and the data type is written on the Model. Data request code is shown as Fig.6.

```

//请求数据
-(void)loadData:(NSInteger)page{
    AFHTTPRequestOperation *manager1 = [AFHTTPRequestOperation manager];
    NSDictionary *dict = @{@"page": [NSString stringWithFormat:@"%d", page]};
    [manager1 GET:@"Recommend_Hot" parameters:dict success:^(AFHTTPRequestOperation *operation, id responseObject) {
        [self hiddenhud:_hud];
        if ([responseObject[@"status"] intValue]==1) {
            for (NSDictionary *sub in responseObject[@"data"]) {
                HotModel *model = [[HotModel alloc] init];
                [model setValuesForKeysWithDictionary:sub];
                [_hotArray addObject:model];
            }
        }

        //刷新UI
        [_indexTableView reloadData];

        //结束刷新动画
        [_indexTableView.header endRefreshing];
        [_indexTableView.footer endRefreshing];
    }
    } failure:^(AFHTTPRequestOperation *operation, NSError *error) {
    }
};
}

```

Fig.6. Data request

The APP supports map positioning, display maps, display location tagging, custom positioning map ICONS and precision ring styles. Map Initialization is shown as Fig.7.

```

//配置用户Key
[MAMapServices sharedServices].apiKey = @"2ef42467e406f573f311c40d6c0fd118";
_mapView = [[MAMapView alloc] initWithFrame:CGRectMake(0, 0, CGRectGetGetWidth(self.view.bounds), CGRectGetHeight(self.view.bounds))];
_mapView.pausesLocationUpdatesAutomatically = NO;
_mapView.allowsBackgroundLocationUpdates = YES; //iOS9以上系统必须配置
_mapView.delegate = self;
_mapView.showsUserLocation = YES;
[self.view addSubview:_mapView];
[_mapView setUserTrackingMode: MAUserTrackingModeFollow animated:YES];

```

Fig.7. Map Initialization

## 5 Conclusion

Mobile APP is the trend of future software applications. We chose light green as the main color of the APP. It also meets the trend of low-carbon travel. The APP realizes the basic functions of tourism. But the UI interface needs to be further refined. APP can also add some health data interface in subsequent development.

## Acknowledgement

In this paper, the research was sponsored by Teaching research project of CTGU (Project No. J2017016).

## References

- [1] ZHOU Chunyan, ZHANG Xinxin. Study on Travel APP Users' Experience[J]. Journal of Daxian Teachers College. 2015 25(6) 99-103.
- [2] Jie Tong, Ji-ping Wang. Travel APP Design Under The Mobile Internet Thinking - Taking LVBAN APP For an Example[J]. Design. 2016(15) 135-136.
- [3] Li Li, Daoyuan Li. Understanding Android app Piggybacking: A Systematic Study of Malicious code Grafting[J]. IEEE Transactions on Information Forensics and Security. 2017 6(12) 1269-1284.
- [4] ZHANG Si-yang, HUANG Tong-cheng. The Design and Implementation of the Android APP for the Self-service Travel[J]. Computer Knowledge and Technology. 2017 13(1) 95-97.
- [5] Lihao xu, J. Bruck. X-code: MDS array codes with optimal encoding[J]. IEEE Transactions on Information Theory. 1999 45(1) 272-276.
- [6] JIN Chao, FENG Dan, LIU Jing-ning. Extending and analysis of X-Code[J]. Advances in Manufacturing. 2011(3) 194-200.
- [7] YANG Yang, OU Dongxiu, HE Xiangjun. A Travel Data Collection Method Based on Big-data from a Smart Phone APP[J]. Journal of Transport Information and Safety. 2015 33(6) 40-47.