**Final Project Report**

Mobile Application Programming

April 28 2017

App developed : CycPop

Author: Yaodong Wang and Xiaomeng Cao

1 Introduction

CycPop is a simple and user-friendly application. Minimalism is the philosophy concept means “less input, accurate information outcome”. Entering the activities once, and the app will automatically record time and location related to your activities. Once you require some information similar to your past activities, it will automatically represent useful information related to your past time and location. Moreover, CycPop provides more convenient ways to browse your activity, including timeline, map view, and list view, so the user can easily be reminded what they did in the past in order to schedule their activities.

2 Architecture

The app structure of CycPop conforms to Apple’s development MVC specification, that the GUI and data model are separated and synchronized via UIViewControllers.

2.1 View

The Views are mainly implemented through graphic component in Interface Builder, and sparsely programmatically generated in corresponding UIViewControllers. There are 4 Views that provide various function and interaction for uses.

* HomeView that is the entry of the app presenting the initial content. Use can record new message, and a default records retrieving runs  automatically.
* TimeView provides users with timed records. When a user needs retrieve records based on time cycle, he/she can choose various pattern of cycle, such as daily, weekly, monthly.
* MapView, on that a MapView component is deployed on the super view, offers uses the location-based records. Users can choose a location visualized on the map, that results in relative records coming out.

2.2 Custom Layout and Animation

Since there are views that require custom layout or use identical display animation. Layout and Animation is decoupled from Views, on that views can focus on interaction and display, while other layout or animation can be programmatically implemented as an individual package. We use itemCell to specify a custom Table Item that display uses memo message, relative time and location.

The process of representing the result - as bubbles pops up to the view - is in charge of animation package. And because most of views use this animation, an individual package is provided.

2.3 Controller

Each view is owned by a view controller. In CycPop, there are 4 controllers that are responsible for their views: HomeViewController, TimeViewController, MapViewController, ItemsViewController. In order to organize them to display, the development takes adavatange of UITabBarController that embed all views as its subviews. Another upside using tab bar controller is that shared constants or variables are easily to be accessible between tab bar’s subveiws via index.

Another point of controller is to synchronize data between UI and Model. In this app, HomeViewController and ItemsViewController have the sync responsibility. HomeViewController synchronizes input message from user interface and data items in back-end model, while ItemsViewController sync removing manipulation on the items in the Table View and back-end data model.

2.4 Model

The model takes charge of data provider and archiving. The model is represented as swift package “ItemStore”, “Item”. The Item class is a plain data model that recording kinds of information, e.g., property memo stores user input message, property date stores NSDate object that facilitating time-based retrieving, and property location using GPS to provide location-based retrieving. Each time the app switches to background mode or gets terminated, the archiving happens in the item model. ItemStore provides add, deleted semantics on items.

3 Tech details

3.1 Shared Object via AppDelegate

AppDelegate.swift:  in the “ didFinishLaunching “, we downcast window to tabBarController, and from tabBar, we can subscribe sub controller. And shared variables or constants are assigned to sub controller objects, such as itemStore (shared between all sub controllers), bubleAnimiationHander (shared between ¾  controllers) .

3.2 Delegates, Protocols

* Many delegates or protocols used, e.g., for text input, UITextFieldDelegate to handle like “textFieldShouldReturn”
* Package item conform to NSCoding protocol to archive and unarchive itself. Package itemStore persist data via NSRUL and  “func saveChanges()” with archivieRootObject for all items
* CLLocationManagerDelegate to update user’s location.

3.3 NSNotificationCenter and Animation

* Retrieved messages are animated as bubbles, which use UIView.animationWithDruation to change the frame.x and frame.y. This is encapsulated as package “BubbleAnimaiton”, that it can play the animation on selected “memo bubbles” based on time or location parameter.
* Dynamic layout of Text Field, that not be overlapped by pop up keyboard.
* addObserver of “UIKeyboardWillShowNotification” and “UIKeyboardWillHideNotification” into NSNotificationCenter.defaultCenter.  And in this notification function, we program TextField anchor constraints and moving up animation with keyboard.

3.4 Map and Location

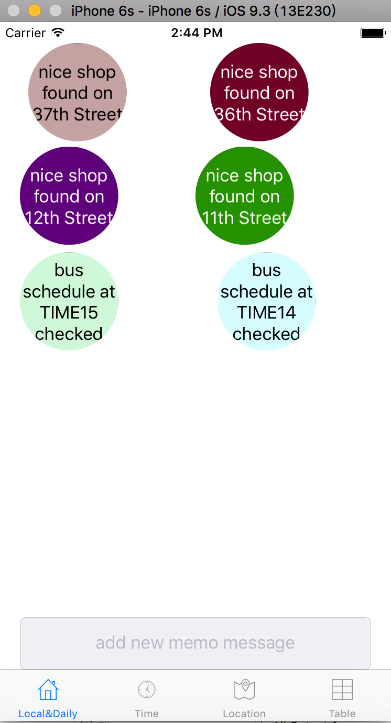
* To get user’s current location, we use CLLocationManager. First, we add new information property into info.plist - NSLocationWhenInUseUsage to offer app permission to user’s location.
* Then, in viewDidLoad, set up locationMgr delegate, and in viewWillApear, we request WhenInUserAuthorization and start update location, after that, we set mapView to show the updated locaiton, via latitude and longitude.
* To update new location on the mapView, we delegate method “func mapView( : regionDidChangeAnimated: ) to renew map’s new center coordinate, and pass the renewed location as argument to animation handler, to display new location-based pop-up bubbles.

3.5 Time cycle pattern

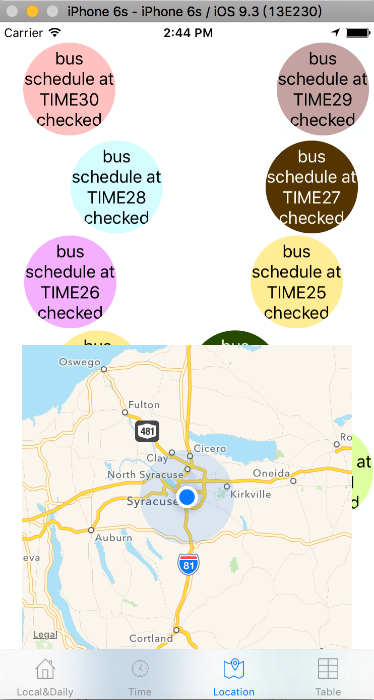
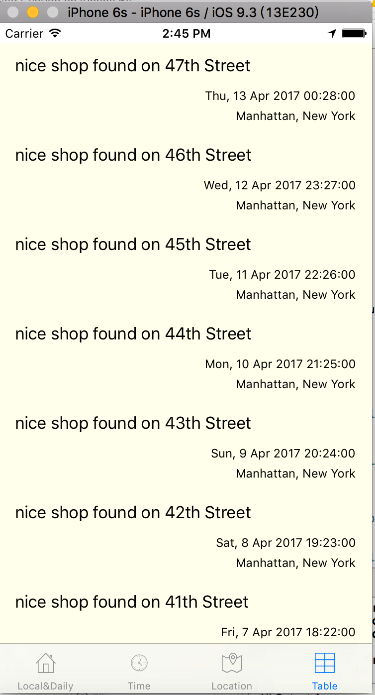
Time pattern is to tell which cycle pattern that user selected, such as show me the memos in Daily, Weekly, or Monthly cycle. For example, if it was daily pattern, that a memo is retrieved if it meets 24, 48, 72... hours from now.

We use the “NSDate.timeIntervalSinceNow” to return the second interval. And then according enum pattern { Daily, Weekly, Monthly, None }, the cycle is calculated respectively. For example, we get hour difference ( ti / 3600 ) , day diff ( hours/24 ),  daily cycle = hours%24, Weekly cycle = days%7, Monthly cycle = days%30,  and if pattern argument is None, cyc = 0.

4 Demonstration

HomeView TimeView

MapView ItemsView

5 Summary and Future Work

In conclusion, even the CycPop is not a app with complex function, it can reminder users what they want to do and what they need to do.You can quickly add and organize your activities while you’re on the go. It is simple and efficient application for the daily use, We are trying to avoid too much reminder, so the CycPop only operate when the user open it. Once you start it, you will love it. Small changes can make a big different for the daily life, let’s the CycPop instead of your physical reminder and notebook. Saving paper is saving our life.

For the next version of CycPop, we will add notification function in order to remind user without open the app. In addition, sharing is always the best thing is the life, developing a bluetooth function help the users be more engaged.

If our app can be accepted by the customers. We can develop an app for Apple Watch, so users can view their activities without taking their phone.