Smartphone Development

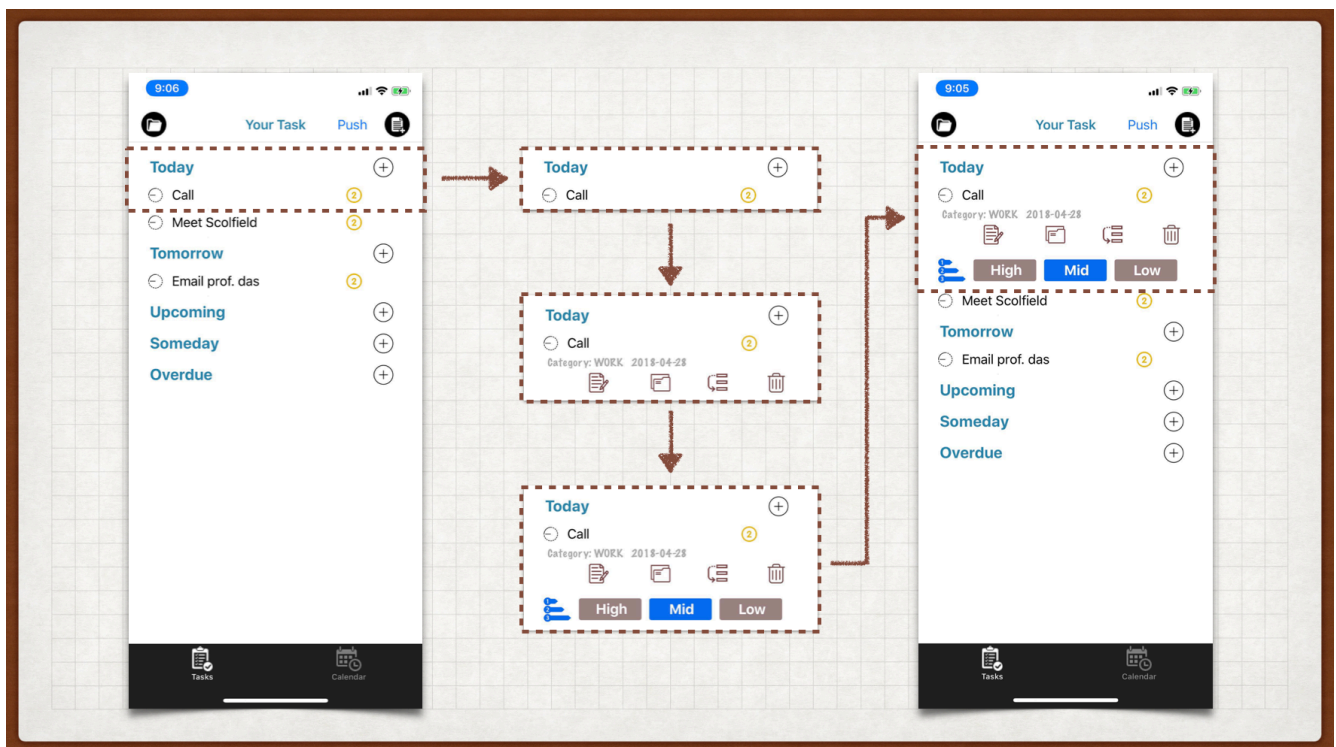*INFO_6350 To do list Software*

Ruohua Yin

NUID:001822668

2018 Spring

INFO_6350 Final Project Report

Introduction

There are plenty of to-do-list or productive app in Apple Stores, but most of the person can't get used to manage their schedule on a mobile device even if almost everybody keep a smart phone with them. So I'm trying to explore the reason and the shortcoming of existing mobile app in the market, and then make a app myself trying to combine the advantages they have.

Detail Description

Description

TASK

Task ID

Title

Type **Priority** **Is Selected** **SetupTime**

Icon

categoryNam
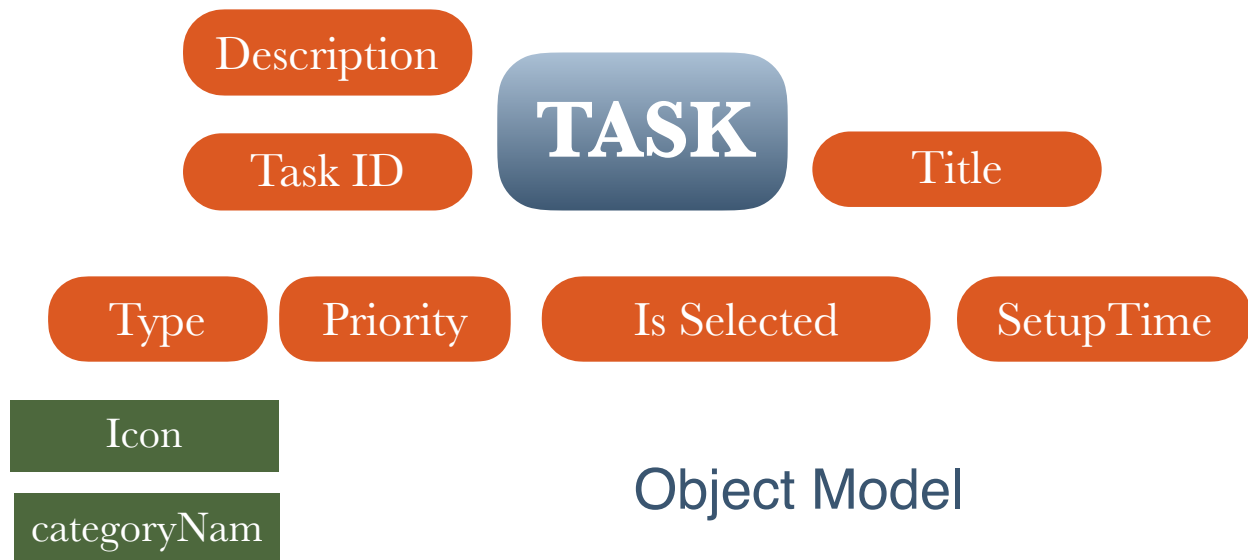
Object Model

The main data structure could be really easy for a to do list to use. The above data structure could nearly support all the data store demand. But 3 or 4 more helper data structure is going to support the search, manage, expandable view functions and data visualizations.

```swift
struct ExpandableTasks{

    var isExpanded: Bool
    var tasks: [Task] = []

}

struct Task{

    var taskID : Int
    static var taskCounter: Int = 1
    var title : String
    var setupTime : Date
    var isFinished : Bool = false
    var taskType: String
    var taskDescription: String
    var taskPriority:String

    init(titleName:String, time:Date, type:String) {
        taskID = Task.taskCounter
        Task.taskCounter += 1
        title = titleName
        setupTime = time
        isFinished = false
        taskType = type
        taskDescription = ""
        taskPriority = "Mid"
    }
}
```
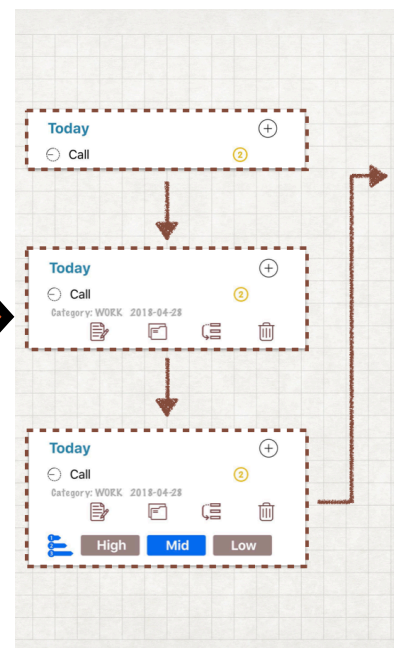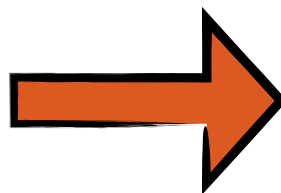
The implement of expandable TableViewCell is with the help of a ExpandableTask Structure. The TableView will reload data by reading the boolean value of the structure. All the true value structure will be inserted to the tableView, in the opposite, the false value structure will be delete when user click on the header.
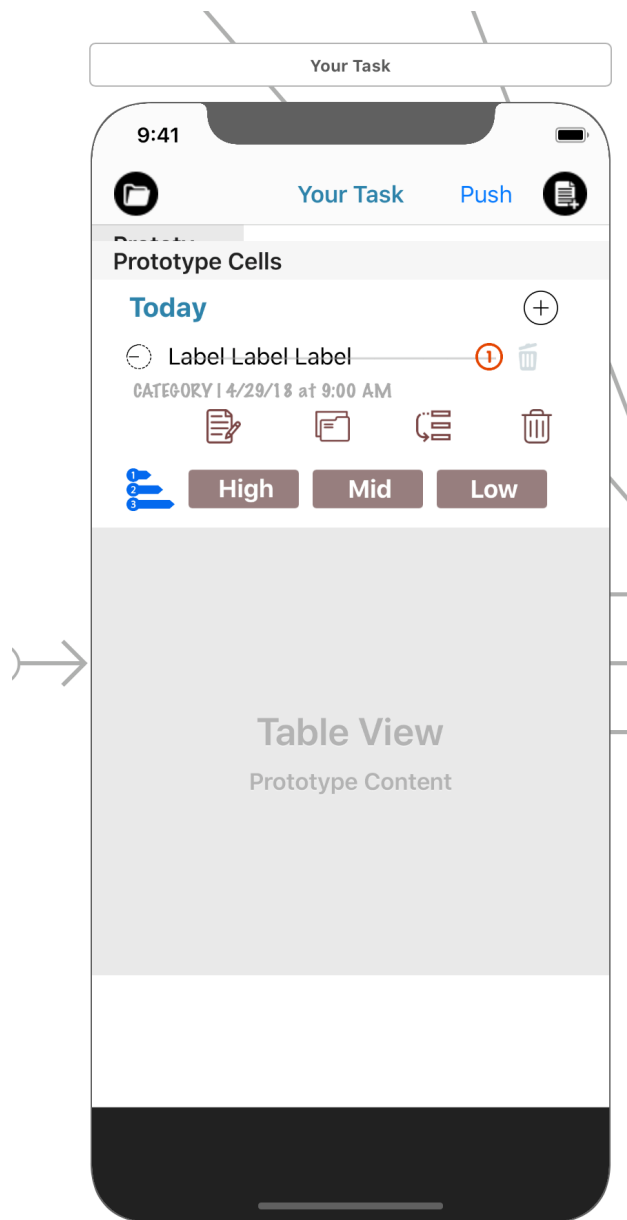
TableView is in charge of most of the data visualize work. The header here is a cell with a distinct identifier
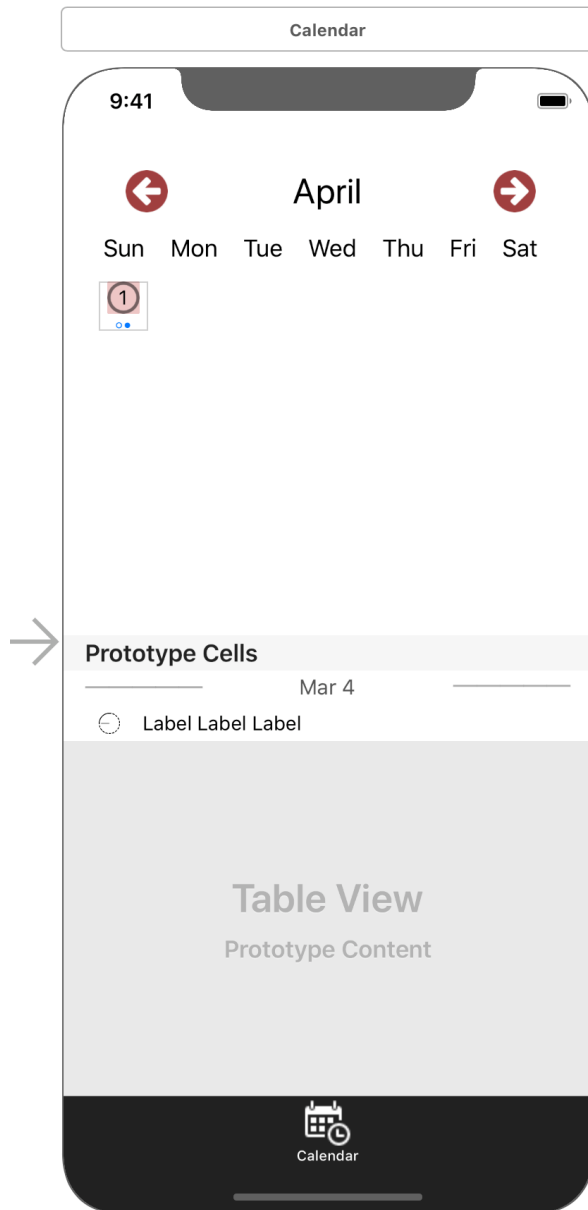
Collection view is in charge of the Calendar implementation. The cell is generate with indexPath, with some basic math tricks. The date will reveal on the board.

1 There are task indicator on the bottom of the date number

2 The selected view will have a red back ground

3 The table view will scroll to the date in selected Calendar cell

Design

The is one function I have 90 percent finished, which is add a skill to Amazon Echo. The process is shown as followed. We use the Alexa developing kit to configure skills. There could have several intent and corresponding utterance to invoke the skill. The Skill trigger could deploy on AWS function or HTTP request. The echo will decode the JSON file in return and read them out.

I plan to use the firebase as the cloud storage because it's perfectly worked with IOS, and the most importantly, it provide a RESTful API which the data in firebase can be access directly.

I want the Alexa have at least two intent in the Skill kit. The first one is to read my to do list for today. The second one is to add a Item by voice to today task which is trigger by the "Alexa, add to my super awesome to do list that I'm going to football match today"
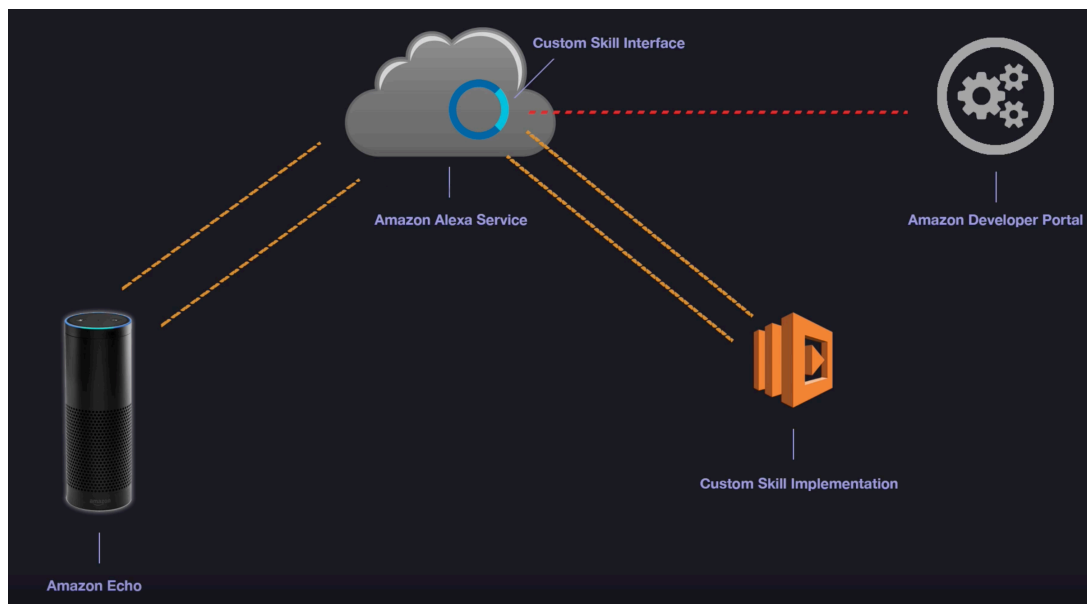
Future Enhancement

1 Custom Settings： Theme Color， Category Management， task type icon choice

2 AutoLayout

3 AlexaSkill