Assignment Cover Sheet - Group

Faculty of Science and Technology

**UNIT CODE: SIT302**

**ASSIGNMENT/PRAC No.: Assignment2**

**ASSIGNMENT/PRAC NAME:**

**Project Research and Investigation Report**

**DUE DATE: 16 January, 2016**

**Plagiarism and collusion**

Plagiarism occurs when a student passes off as the student’s own work, or copies without acknowledgment as to its authorship, the work of any other person.

Collusion occurs when a student obtains the agreement of another person for a fraudulent purpose with the intent of obtaining an advantage in submitting an assignment or other work

**Declaration**

We certify that the attached work is entirely the work of the group, except where work quoted or paraphrased is acknowledged in the text. We also certify that it has not been submitted for assessment in any other unit or course.

I agree that Deakin University may make and retain copies of this work for the purposes of marking and review, and may submit this work to an external plagiarism-detection service that may retain a copy for future plagiarism detection but will not release it or use it for any other purpose.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIRST NAME** | **FAMILY NAME** | **STUDENT ID** | **\*CONTRIBUTION**  **(%age)** | **DATE** |
| **XUN** | **SUN** | **215012871** | **100%** | **16 Jan** |
| **YUNHAO** | **ZHOU** | **215012174** | **100%** | **16 Jan** |
| **WANMIN** | **CHEN** | **215012989** | **100%** | **15 Jan** |
| **SAISANAS** | **PETER** | **213194708** | **100%** | **15 Jan** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

An assignment will not be accepted for assessment if the declaration appearing above has not been duly completed by the authors.

\* The contribution value shall be a percentage representing the amount of work completed by that group member in regards to their allocated task(s). For example, the contribution value for a particular group member shall be:

* 0% if that group member completed none of their allocated task(s),
* 50% if that group member completed half of their allocated task(s), and
* 100% if that group member completed all of their allocated task(s).

Project Proposal and Plans

SIT302 Project

Time Management and Activity Tracking

Client: Prof Paul Gastin

Supervisor: Dr. Mohamed Abdelrazek

Delivered on: Jan 16, 2017

Submitted by: SIT302 Team NO.16

Contact Email: zyunha@deakin.edu.au

Table of Contents

[1 Problem defined 5](#_Toc472322452)

[2 Systems Analysis and outcome(s) 5](#_Toc472322453)

[2.1 Analysis of several solutions 6](#_Toc472322454)

[2.2 Outcomes 6](#_Toc472322455)

[3 Application functionality clearly defined and explained 7](#_Toc472322456)

[4 Database Design 8](#_Toc472322457)

[4.1 Database Table List 8](#_Toc472322458)

[4.2 Database Model Design 8](#_Toc472322459)

[5 Security Plan 10](#_Toc472322460)

[5.1 Code Security 10](#_Toc472322461)

[5.2 Output 10](#_Toc472322462)

[5.3 Database Security 10](#_Toc472322463)

[6 User interface defined 11](#_Toc472322464)

[7 Data flow and Use case plan 17](#_Toc472322465)

[8 Plans 18](#_Toc472322466)

[8.1 Overview 18](#_Toc472322467)

[8.2 Implementation plan 18](#_Toc472322468)

[8.3 Testing Plan 20](#_Toc472322469)

[8.4 Deployment Plan 21](#_Toc472322470)

[8.5 Maintenance plan 22](#_Toc472322471)

[9 Functioning prototype 22](#_Toc472322472)

[References 24](#_Toc472322473)

## 1 Problem defined

This project is aimed at developing a working prototype of a time management and activity tracking app. The quantification of the amount of time spent by individuals in various activities is of particular interest to researchers and practitioners. For athletes it might be about how much time is spent training and recovering, and how much load is being undertaken. For adolescent students, it might be about how much time is spent studying, in physical activity, in leisure activities, screen time and sleep.

This touch screen app would allow users to identify the types of activities to track and allow these to be ‘dragged and dropped’ into a 24 hour day or weekly calendar. Activities would be positioned and resized by touch. The back-end interface would need to database the data (and allow export) for subsequent analysis and visualisation over a selected period of time.

According to our research and analysis on integrated planner and Hours in the existing system overview, we found that it is necessary to search the details of drag motion for IOS applications. There would be some changes for our application: remove timer, store activities, provide default activities, positioned and resized by touch, and ‘drag and drop’ function.

Therefore, a friendlier user interface than other time management app should be implemented.

## 2 Systems Analysis and outcome(s)

This section should detail the analysis of several solutions proposed by the team that could potentially solve the relevant problem, followed by justification for choosing one in particular.

### 2.1 Analysis of several solutions

* Solution1: Mobile app: Monthly calendar view with timeline on each day

We are suggested to make a mobile application that allow users to create/drag/drop activities into daily timeline view, and select a day from calendar view.

* Solution2: Mobile app: weekly view with timeline on each day

The main interface is a weekly view table, and user can enter the timeline view by touching the date.

* Solution 3: Mobile app: weekly view without a daily timeline view

The main interface is a weekly view table with an activity list on the right side, and all the operations are on this page.

* Solution 4: Mobile app: To-do list

A mobile app to manage and organise tasks into colourful lists. This is an application to help people deal with multiple tasks.

* Solution 5: back-end record system

Garmin has a project with Deakin University. The backend application can track and visualise the data of women basketball team.

### 2.2 Outcomes

Through the discussion with the client, we decided to use solution 1 – a mobile application that can show a monthly calendar view with timeline on each day.

Reasons for choosing solution 2:

This application should display a calendar view for the current month. TODAY and DAY which has set activities will be marked. Besides, user can view last and next month calendar by clicking “Last month” and “Next month”. Every day on the screen can be selected and the main screen will jump to OneDayView. Simultaneously, all the information about the day user selected will be downloaded from cloud and show on the OneDayView.

In OneDayView, there will be a timeline and activities will be attached on this timeline. Each activity has six properties: Date, Name, Start Time, End Time, Note, Finished (Bool). Different activities will show different background colors. The overdue activities are grey. The executing activities are red. The future activities are green. User can know the activity name, start time, end time, note. On the bottom of screen, there will be some icons and each represents one activity. What is more, user can custom new activity by clicking the “+” button. After setting, the new activity will be presented at the bottom as will. User can drag and drop these icons into the timeline to set new activities for current day. In addition, user can resize the duration of activities by using fingers. Once activity data were changed, all the new information will be uploaded and cover the old ones. Besides, activities can be removed by dragging into bin.

## 3 Application functionality clearly defined and explained

Through the discussion of our group, we have several points to emphasize (core functions):

* Allow user to create/delete/modify activities
* Provide default types of activities
* A vertical timeline view for user to allocate, delete and adjust the time of activities
* Allow user to drag and drop activities to the weekly calendar or 24 hour day view
* In the timeline view, activities can be positioned and resized by touch
* Database to store tables
* Backend to store list of users, activities per user, and activities over all users
* Automatically generating reports, and visualisation of report over a selected period of time

When user runs other programs, this application will still run at background. Once it reached the start time of some activity, this application will push notification and remind user to do this activity. When it reaches its end time, this application will notify user as well, and user can set it as Finished or Not Finished.

As for cloud, generally it has two functions. The first one is that it can store user and activity data, so the mobile application will not take too much storage of user’s phone. The other one is that it can collect big data and produce reports for users. It will provide visualized data to analyse user’s behaviour.

In summary, mobile side interact with users and generate personal data, while cloud is used for storing and analysing the data.

Database / Data model design appropriate and defined

## 4 Database Design

### 4.1 Database Table List

TIME\_MGT\_ACCOUNT\_ADMIN // restore administrator information

TIME\_MGT\_ACCOUNT\_USER // restore user account information

TIME\_MGT\_ACCOUNT\_USER\_LOG // log the operation of the user

TIME\_MGT\_ACTIVITY // table to restore the activities

### 4.2 Database Model Design

TIME\_MGT\_ACCOUNT\_ADMIN {

adminid int prime key //administrator ID should be auto gernerated by system automatically

username string //admin name policy is defined in Moodle security config

password string Optional //Plain text password consisting of any characters

createpassword int Optional //True if password should be created and mailed to user

email string //A valid and unique email address

lang string Default to "en" //Language code such as "en", must exist on server

timezone string Optional //Timezone code such as Australia/Perth, or 99 for default

}

TIME\_MGT\_ACCOUNT\_USER {

userid int prime key //user ID should be gernerated by system automatically

username string //Username policy is defined in Moodle security config

password string Optional //Plain text password consisting of any characters

createpassword int Optional //True if password should be created and mailed to user

email string //A valid and unique email address

lang string Default to "en" //Language code such as "en", must exist on server

state string Default to "enable"//The state of the user, such as enable, disable or terminate,

timezone string Optional //Timezone code such as Australia/Perth, or 99 for default

}

TIME\_MGT\_ACCOUNT\_USER\_LOG {

time float //datetime

operation string //KEY operation, login, change password, modify account, etc.

}

TIME\_MGT\_ACTIVITY {

activityid int prime key //id of the activity

activityName string //name of the activity

startDate string //starting time of the activity

endDate string //ending time of the activity

description string Default to "" //description

status string //eg visiable, invisiable, gone

}

## 5 Security Plan

### 5.1 Code Security

All external data should be check and filtered

* Ensure that data filtering cannot be bypassed.
* To ensure that no legitimate information does not affect the legal information.
* Identify data sources.

Check external data

* Data uploaded should be hashed using MD5

Filter data

* Create rules to filtering, for example data format, legal character, etc.

Filter functions to avoid rewriting existing functions

* Using PHP to help filter data.

### 5.2 Output

Encrypted output cookie, and ensure that no debug info output.

This problem can be solved by disabling displaying of error messages to screen

ini\_set(“display\_errors”, FALSE);

And enabling logging of errors

ini\_set(“log\_errors”, TRUE);

to a file

ini\_set(“error\_log”, “/var/log/php.log”);

### 5.3 Database Security

Secure the database configuration information:

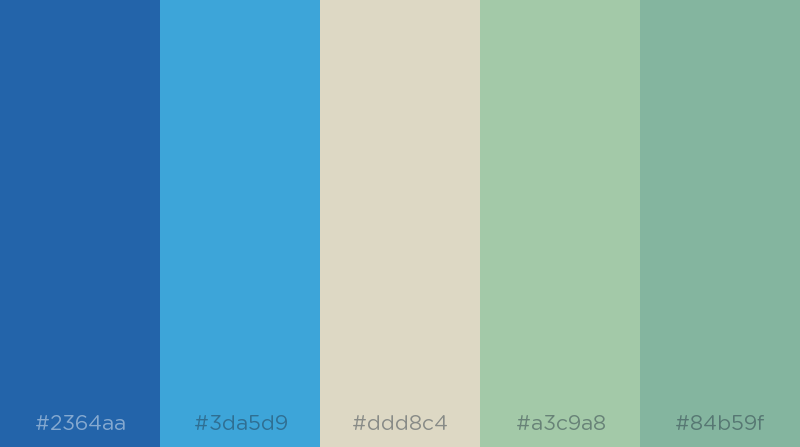
We choose MD5 description to encode the application.

MD5 encoding provides one-way encryption. It cannot be reversed or decoded. It is ideal for storing passwords in the database, which adds an extra layer of security over storing them in text format. For example, use MD5 encoding when storing passwords in the Oracle database.

So even if the database has been compromised, the hacker cannot retrieve the actual password (in text format) being used for the admin login.

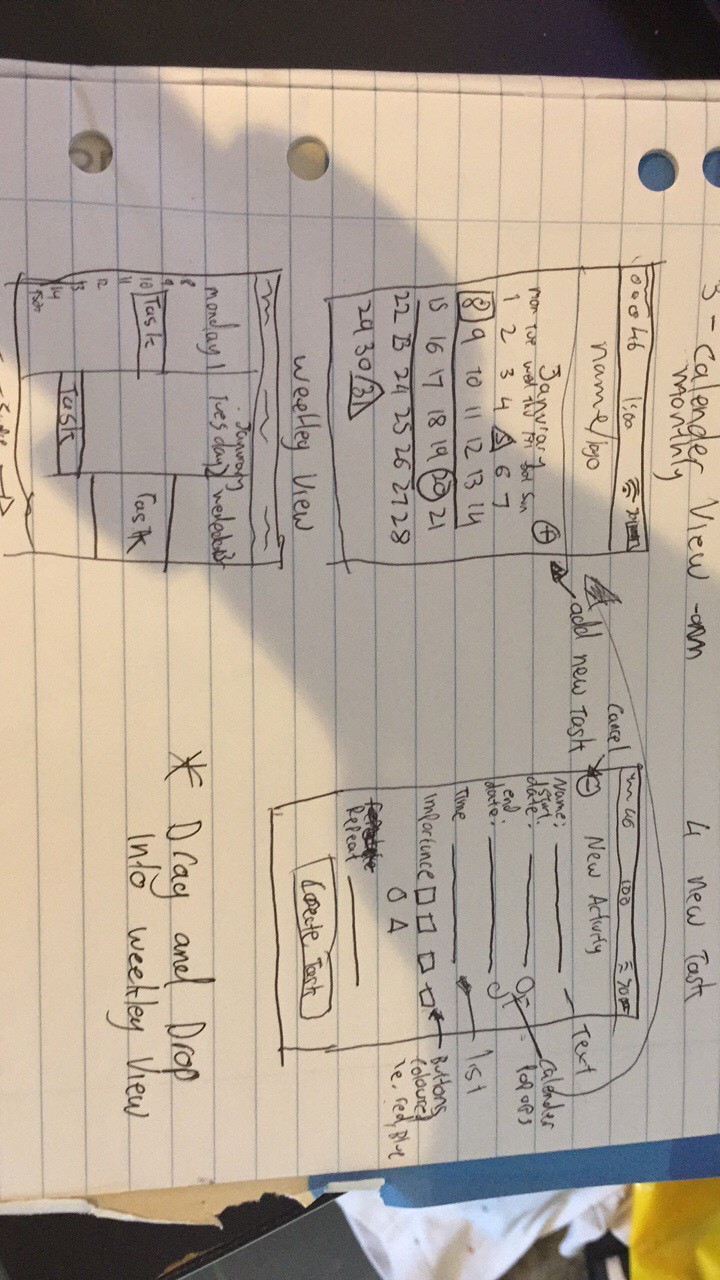
## 6 User interface defined

User interface is considered by some as the most important aspect in the process of building and constructing a successful mobile application. We identified this as the case also, and focused on creating an interface that promotes ease of use for the users and error free navigation throughout the application.

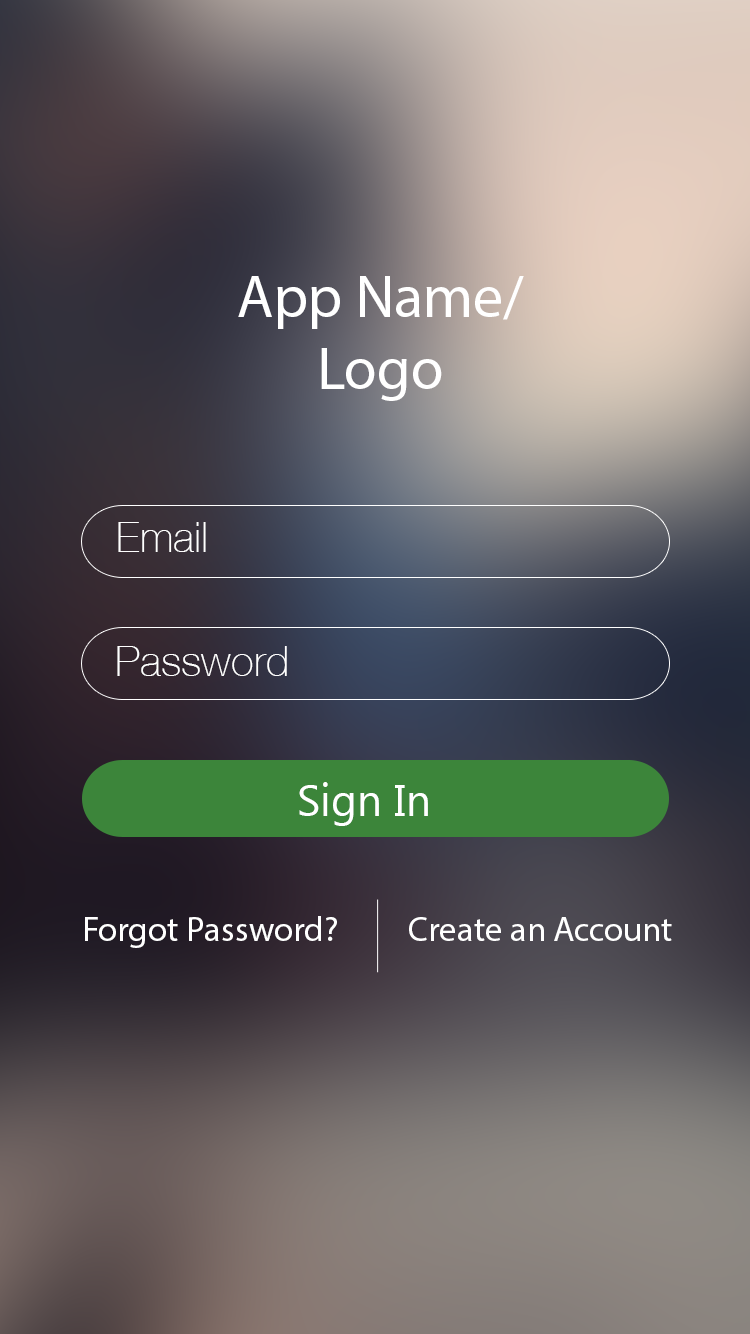
The first step in the interface design was identifying a colorful and welcoming style guide/ color scheme (pictured below) 

The bright blues and light greens bode well together in providing a consistent look throughout the application which assists the user in navigation and understanding.

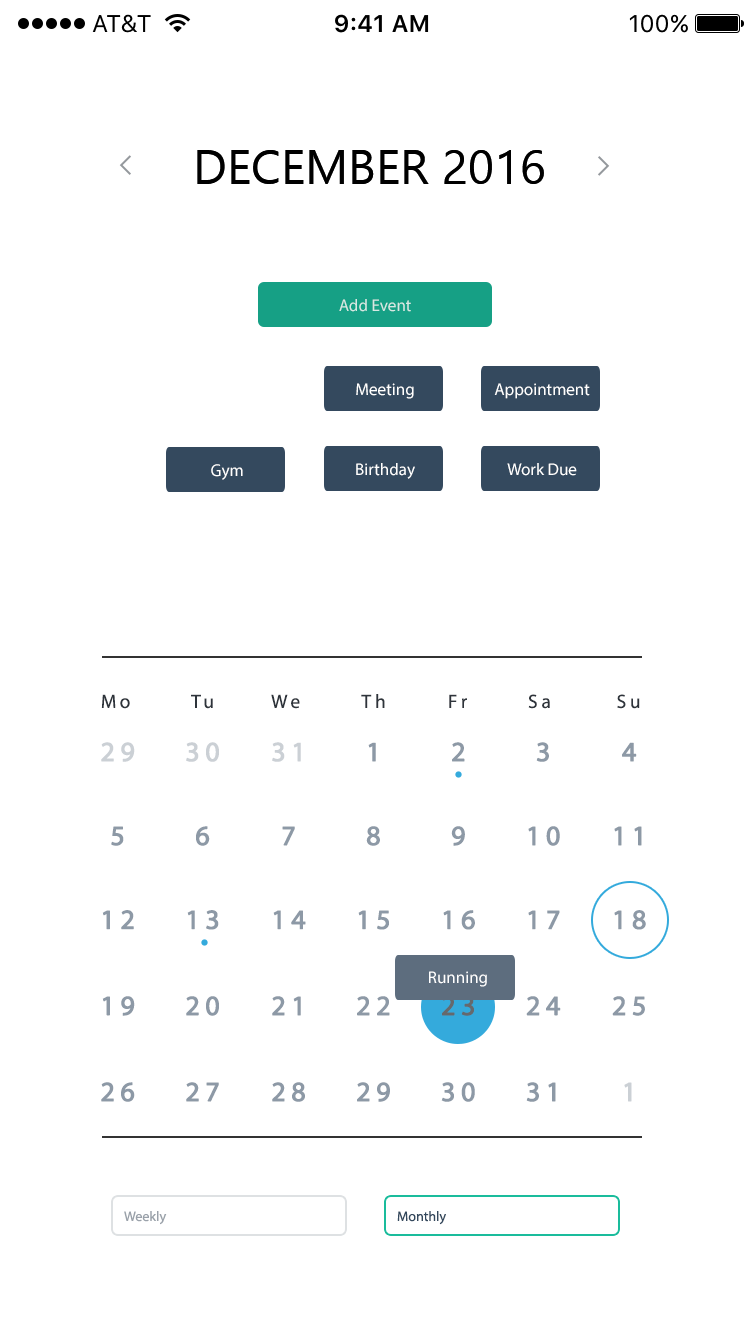
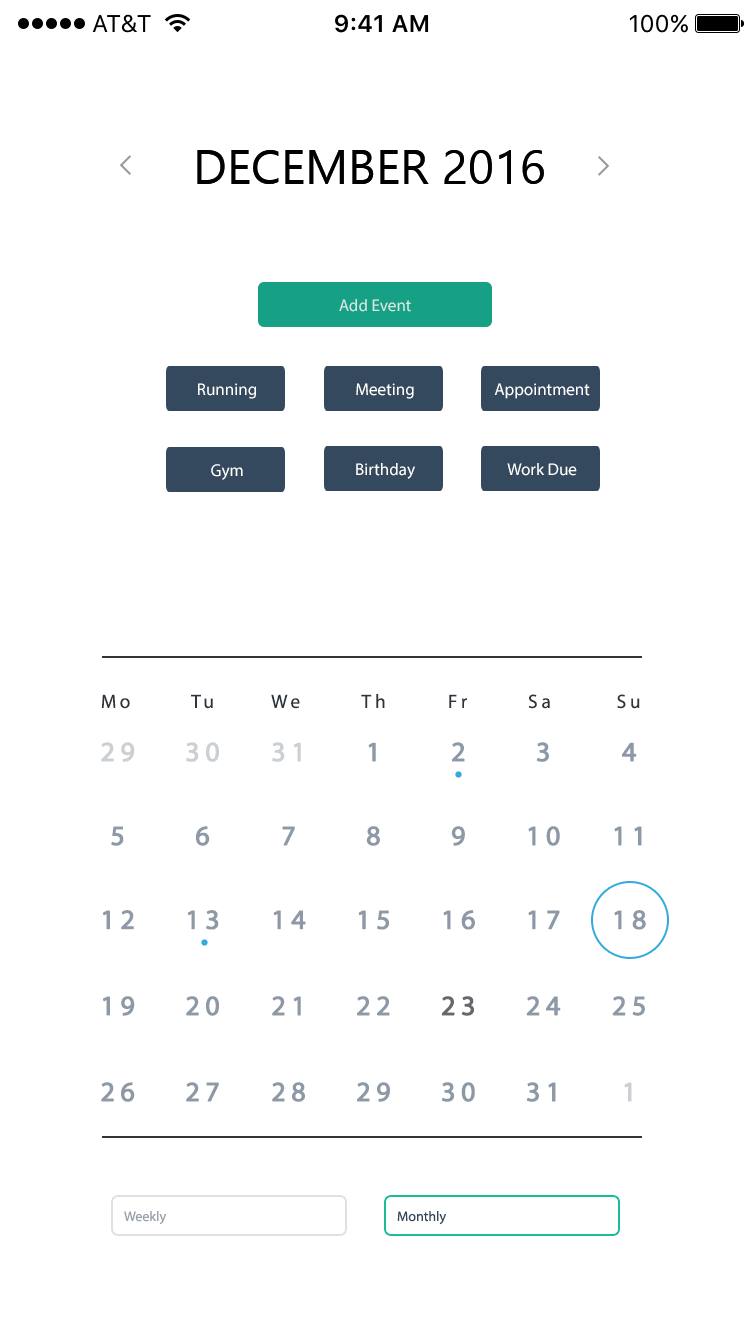
As mentioned prior, promoting ease of use was the underlying most important factor in our user interface goal, through brainstorming and playing around with sketches and concepts, we decided that the best way to incorporate interactivity into the application was through the drag and drop method. Given the application is centred on time management we thought the best way to approach this was through ‘dragging’ activities onto either calendar dates or onto a weekly timetable. As the designer this posed a conundrum on how I would design for this function whilst maintaining user ease of use, error free navigation and understanding.

After many different sketches and ideas – I decided the best way to do this was to drag onto a calendar date whilst observing a monthly view (this was a last minute change from dragging onto a weekly view).

Adopting this last minute change really assisted with what we were going for – as the weekly view interaction would’ve been way too confusing for novice users using the application.

Once the sketches were finished the Interface build began to take shape, firstly starting off with a log in screen – where users can enter their credentials in order to retrieve/save their tasks.

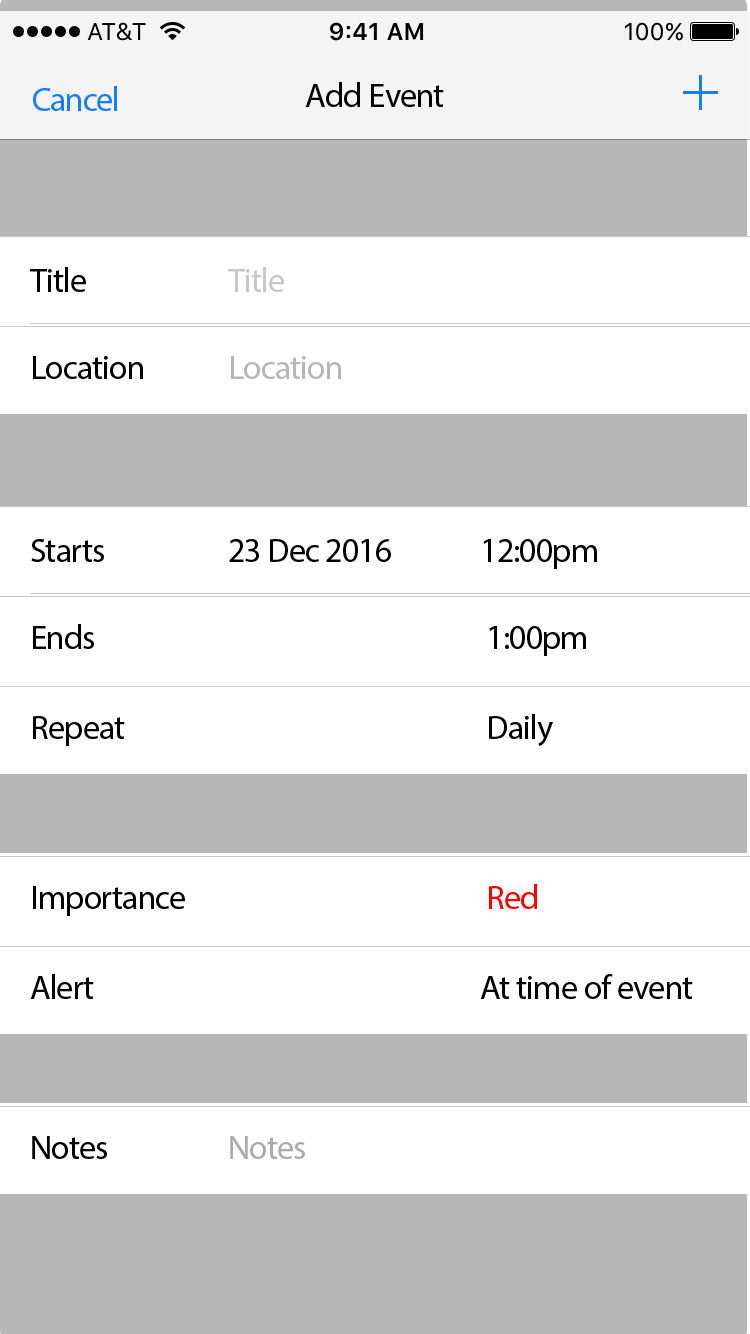
Once the log in screen had been completed we moved on to the main landing screen – the calendar view, with the drag-able interaction. This screen also leads to another important page the ‘Add Event’ page where users click the clearly labelled button and are taken to the screen where they can follow the prompts and add tasks and events to their schedule. All screens are pictured below



Calendar View - Monthly

Calendar View Monthly Dragging

Interactivity



Add Event Screen



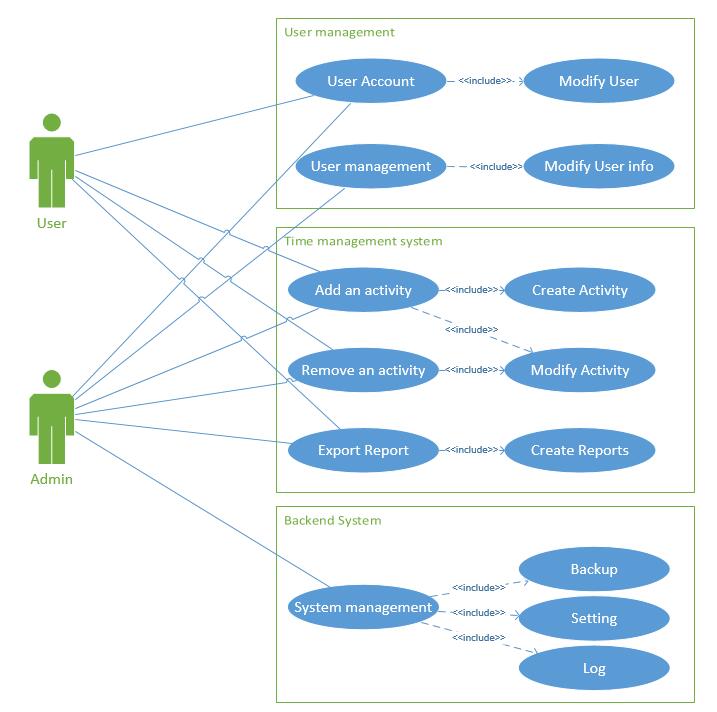
Weekly View

As you can see through the finalised screens on display – the original interface ideologies we had have been upheld. The application is consistent throughout with colours, iconography, shapes, text and positioning, the colour scheme is being utilised to full effect promoting that welcoming vibe and feeling. All these factors merge together to promote an easy to use application for any individual looking to manage their schedule and manage their time.

Another factor which promotes the ease of use are the clearly defined user inputs –found in the add event screen. Users are guided to the screen through clearly set out buttons and text, and once there are furthermore assisted with the labelling of the criteria that they are required to enter – such as, title of activity, date of activity, time of activity, alert time, importance and any additional notes they require.

All in all the user interface as well with the other components of the application development has come together quite nicely – and this bodes well for good final product for the user.

## 7 Data flow and Use case plan



## 8 Plans

### 8.1 Overview

This section will include how our project proceeds, and specify the deliverables that would be produced. There are 4 parts in this section: Implementation plan, Testing plan, Deployment plan and Maintenance plan.

### 8.2 Implementation plan

**1. System overview**

The system includes two parts: mobile and cloud. The mobile side is the main part. Mobile application is the entrance of cloud. All the activities user set will be uploaded into cloud. The cloud side can store user and activity information. Besides, it can also collect all the information to produce user reports.

**2. System describe**

As for mobile, before accessing this application, user should login first. If user does not have an account, he/she can register an account by select register button (User need to provide username, password, email, phone number). In addition, if user forgets their password, they can click the “forget password” button to find their password. The system will send a SMS with unique code to user’s phone and the application will verify this code. If the code match, the application will allow user to reset password.

After logging in, the calendar of current month will be presented on screen. TODAY and DAY which has set activities will be marked. Besides, user can view last and next month calendar by clicking “Last month” and “Next month”. Every day on the screen can be selected and the main screen will jump to OneDayView. Simultaneously, all the information about the day user selected will be downloaded from cloud and show on the OneDayView.

In OneDayView, there will be a timeline and activities will be attached on this timeline. Each activity has six properties: Date, Name, Start Time, End Time, Note, Finished (Bool). Different activities will show different background colors. The overdue activities are grey. The executing activities are red. The future activities are green. User can know the activity name, start time, end time, note. On the bottom of screen, there will be some icons and each represents one activity. What is more, user can custom new activity by clicking the “+” button. After setting, the new activity will be presented at the bottom as will. User can drag and drop these icons into the timeline to set new activities for current day. In addition, user can resize the duration of activities by using fingers. Once activity data were changed, all the new information will be uploaded and cover the old ones. Besides, activities can be removed by dragging into bin.

When user runs other programs, this application will still run at background. Once it reached the start time of some activity, this application will push notification and remind user to do this activity. When it reaches its end time, this application will notify user as well, and user can set it as Finished or Not Finished.

As for cloud, generally it has two functions. The first one is that it can store user and activity data, so the mobile application will not take too much storage of user’s phone. The other one is that it can collect big data and produce reports for users. It will provide visualized data to analyse user’s behaviour.

In summary, mobile side interact with users and generate personal data, while cloud is used for storing and analysing the data.

**3. Roles and responsibilities**

**3.1 Roles introduction**

There are four members in this team: Zhou Yunhao, Sun Xun, Chen Wanmin and Peter. Paul is the project client and Mohammed is the supervisor.

**3.2 Responsibilities detail**

Zhou Yunhao works as Project Manager in this team. He is responsible for communicating with Paul and controlling the scope of this project. He will define the problem, analyse the system and define the functionality and explain to other members. He must finish these work before 20 Nov 2016. Besides, he also needs to make plans for this project and write Plan documentation before 27 Nov 2016.

Sun Xun is the mobile developer. He is responsible for implementing all the functions on mobile. He should meet with Mohammed weekly and get suggestions. He must finish the prototype of this application and write Security Plan, Data Flow/ User Case/ Object Definition before 15 Jan 2017.

Chen Wanmin is the backend developer. His work is to design and develop user and activity database. Besides, he need to build the bridge between database and mobile. In addition, Database / Date model design appropriate and defined is edit by him. He must finish these work before 18 Dec 2016.

Peter is the UI designer. His work is to design the application interface and write User Interface / Style guide / Input processes defined. These work need to be done before 1 Jan 2017.

### 8.3 Testing Plan

**1. Purpose**

The testing plan will include that how to provide actual tests. The details of the tests should be recorded. Additionally, the testing content and a brief introduction of strategies will be given.

**2. Application testing**

For this testing plan, we decided to have two different ways to test our application.

* The group members should test the application and generate reports. All the records of testing should be written in tables.
* Volunteers will help us testing the application and the report generating function. They should fill out the tables and send it to the tester.

**3. Testing Strategy**

Scrum methodology will be used as the testing strategy. Testers will be divided into two parts. They are volunteers and group members. At the beginning, a template will be set up and it will be filled in content of test, validation, and detail by setting boundaries of testing and recording results. Our team will also share the experience and the drawbacks of current product, and put these drawbacks into next iteration requirements.

### 8.4 Deployment Plan

**Overview**

All of activities that develop the Time management application will be described in the deployment plan. There are four specific activities: launching, installing, releasing and updating.

**Launching**

After finishing the report of analysis and design, the Time management application will be launched on 2 Feb 2017. Our group member will have a meeting on 1 Feb 2017 to confirm all of requirements and distribute specific tasks to each group member.

**Installing**

After the launched activity, the IOS developer (Xun Sun) will build the website sever and database on 2 Feb 2017. Other group member will start to prepare the system of Time management application ready to automatically generate report at the same time.

**Releasing**

Then, the project will transfer to user interface, the testing about DIG website will be carried out and the group members will integrate all the codes. Finally, we will release the Time management app after basic system problem checked and reviewed.

**Updating**

After releasing of Time management app, this website will be submitted to the supervisor Mohammed. The updating activity will be executed by other person if the project has been accepted.

## 8.5 Maintenance plan

At the end of our project, the application will be submitted to the supervisor and unit chair.

The group members may not join the modification and maintenance of Time management app. Therefore, the time estimates and costs are not necessary for the development plan.

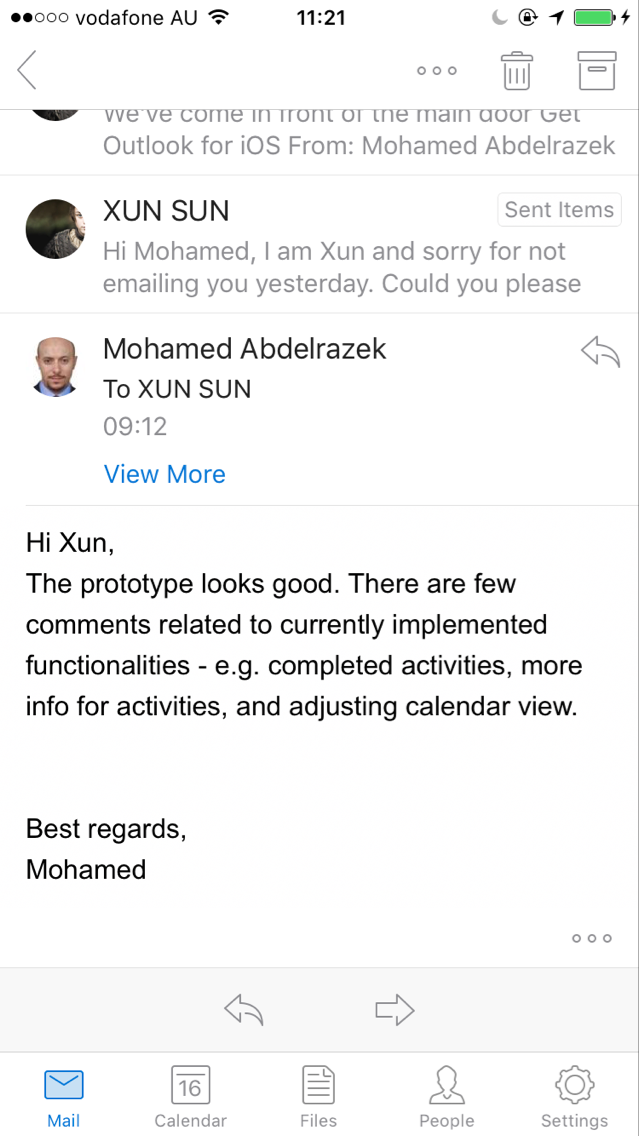
## 9 Functioning prototype

The essential functionality in this prototype includes login function, calendar view function, daily view function, “create activity” function, and automatically upload the activity data to cloud.

We have created a video for our functioning prototype and it is uploaded to YouTube.

Link: <https://youtu.be/Q_zcaDjpJPc>

The short statement from supervisor Mohamed:



The prototype looks good. There are few comments related to currently implemented functionalities - e.g. completed activities, more info for activities, and adjusting calendar view.

Best regards,

Mohamed

## References

Jerry C, What Is a Prototype: A Guide to Functional UX, UXPin Sp. z o.o., < <https://www.uxpin.com/studio/blog/what-is-a-prototype-a-guide-to-functional-ux/>>.

Microsoft, 2012, Plan PHP Application Security, May 4, 2012, < [https://technet.microsoft.com/en-us/library/hh994605(v=ws.11).aspx/](https://technet.microsoft.com/en-us/library/hh994605(v=ws.11).aspx/%20) >.