

**Pick Pic**

**Software Requirement Specification**

**2017 - 1 Human ICT Software Engineering | 2017.06.02**

|  |  |
| --- | --- |
| 20155652 | KangSoYeon |
| 20154019 | LeeYeWon |
| 20151722 | MyungSeKyo |
| 20152164 | LeeWonJun |
| 20151535 | ParkGunHoo |

Table of Contents

[1.0 Product description 4](#_Toc484191612)

[1.1 Introduce 4](#_Toc484191613)

[1.2 Goal 4](#_Toc484191614)

[1.3 Function 4](#_Toc484191615)

[1.3.1 Normal Gallery 4](#_Toc484191616)

[1.3.2 Automatically tag\* image 4](#_Toc484191617)

[1.3.3 Synchronizer 5](#_Toc484191618)

[1.3.4 Image View 5](#_Toc484191619)

[1.3.5 Show category 5](#_Toc484191620)

[1.3.6 Word/Selective search to find images with tags\* 5](#_Toc484191621)

[\*kinds of tag 6](#_Toc484191622)

[2.0 Pick Pic Icon 6](#_Toc484191623)

[3.0 UI Diagram 7](#_Toc484191624)

[4.0 Use cases 7](#_Toc484191625)

[4.1 Main 8](#_Toc484191626)

[4.1.1 Synchronizing 8](#_Toc484191627)

[4.1.2 Directory tab 8](#_Toc484191628)

[4.1.3 Time tab 9](#_Toc484191629)

[4.1.4 Tag tab 9](#_Toc484191630)

[4.1.5 Menu 9](#_Toc484191631)

[4.1.6 How to use 10](#_Toc484191632)

[4.2 Search 10](#_Toc484191633)

[4.2.1 Search 10](#_Toc484191634)

[4.2.2 Repeatedly search in searched result 10](#_Toc484191635)

[4.3 ImageViewer 11](#_Toc484191636)

[4.3.1 Image View Screen 11](#_Toc484191637)

[4.3.2 Image Tag Screen 11](#_Toc484191638)

[4.3.2.1 Add Tag Screen 11](#_Toc484191639)

[4.3.2.2 Delete Tag Screen 12](#_Toc484191640)

[4.3.3 Image Setting Screen 12](#_Toc484191641)

[4.3.3.1 Share Image 12](#_Toc484191642)

[4.3.3.2 Rotate Image 13](#_Toc484191643)

[5.0 Process description 13](#_Toc484191644)

[5.1 Software toolset-client 13](#_Toc484191645)

[5.1.1 Android Studio 13](#_Toc484191646)

[5.1.2 ION 13](#_Toc484191647)

[5.2 Software toolset - server 13](#_Toc484191648)

[5.2.1 tensorflow(inception v3) 13](#_Toc484191649)

[5.2.2 dominant color 13](#_Toc484191650)

[5.2.3 color-namer 14](#_Toc484191651)

[5.2.4 express 14](#_Toc484191652)

[5.1.5 python-shell 14](#_Toc484191653)

[5.1.6 textract 14](#_Toc484191654)

[5.3 Group organization(roles) 14](#_Toc484191655)

[5.4 Risk Summary 14](#_Toc484191656)

[5.3.1 Synchronizing problem 14](#_Toc484191657)

[5.3.2 Data management 15](#_Toc484191658)

[5.3.3 Text recognition and object recognition accuracy problem: 15](#_Toc484191659)

[5.3.4 Server performance limitation issues 15](#_Toc484191660)

[5.5 Schedule / Timeline 15](#_Toc484191661)

[6.0 Meeting reports 16](#_Toc484191662)

[6.1 Talk about 16](#_Toc484191663)

[6.2 Feedback 16](#_Toc484191664)

# Product description

## 1.1 Introduce

Today, smart phone memory is largely increased and there are many images on your gallery.

And also with the development of SNS, there are lots of chance to sharing images.

But you may have the experience that you were fail to find the image you want because there were too many images.

This application is for the people who have difficulty in finding or managing there images.

## 1.2 Goal

The goal of this application is by automatically tagging all images in your gallery, make easy to find the images you want. Additionally tag will help you to manage the images in your gallery.

It makes you finding the precious memory in your gallery and you will never going to lose your memory.

## 1.3 Function

### 1.3.1 Normal Gallery

Pick Pic provide a basic gallery function.

First, it shows image thumbnail in gridview.

Second, when you click thumbnail of one of image, it shows the image full screen.

### 1.3.2 Automatically tag\* image

Pick Pic sends the image to the Pick Pic server, which creates a tag for the image and sends the tag to the client.

The Pick Pic server’s tag is the object, domain color, and text for the image.

The additional tag is the date, directory for the image

### 1.3.3 Synchronizer

Pick Pic synchronizes tagDB based on Local Images DB.

Delete the deleted image

Add the added image, request tags to Pick Pic server and add the additional tag.

### 1.3.4 Image View

When you click one image, it provides several functions

You can zoom in/out image and rotate image.

C:\Users\Arsene holmes\AppData\Local\Microsoft\Windows\INetCacheContent.Word\그림1.pngWhen you touch button above, you can see the tags belongs to the image. The button changes image view and tag list.

Apart from the way that artificial intelligence automatically adds tags. You can see these tags at tag list. You can add and delete your own tags to your images. You can zoom in / out your image.

You can also share your image on SNS by click share button in menu.

### 1.3.5 Show category

Pick pic shows three categories of image classification.

1. By directory
2. By time
3. By tags\*

### 1.3.6 Word/Selective search to find images with tags\*

You can find the images you want by searching the tags.

It provides repeatedly search in searched result.

As a result you can search the image with several tags.

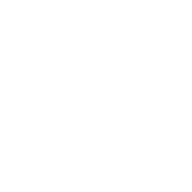
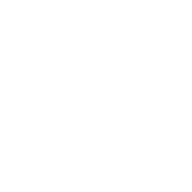
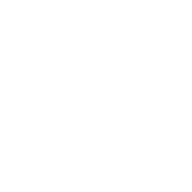
It also provides automatic completion when you search the image.

The searched result will be shown in gridview.

### \*kinds of tag

1. color
2. object
3. date
4. directory
5. text
6. And so on

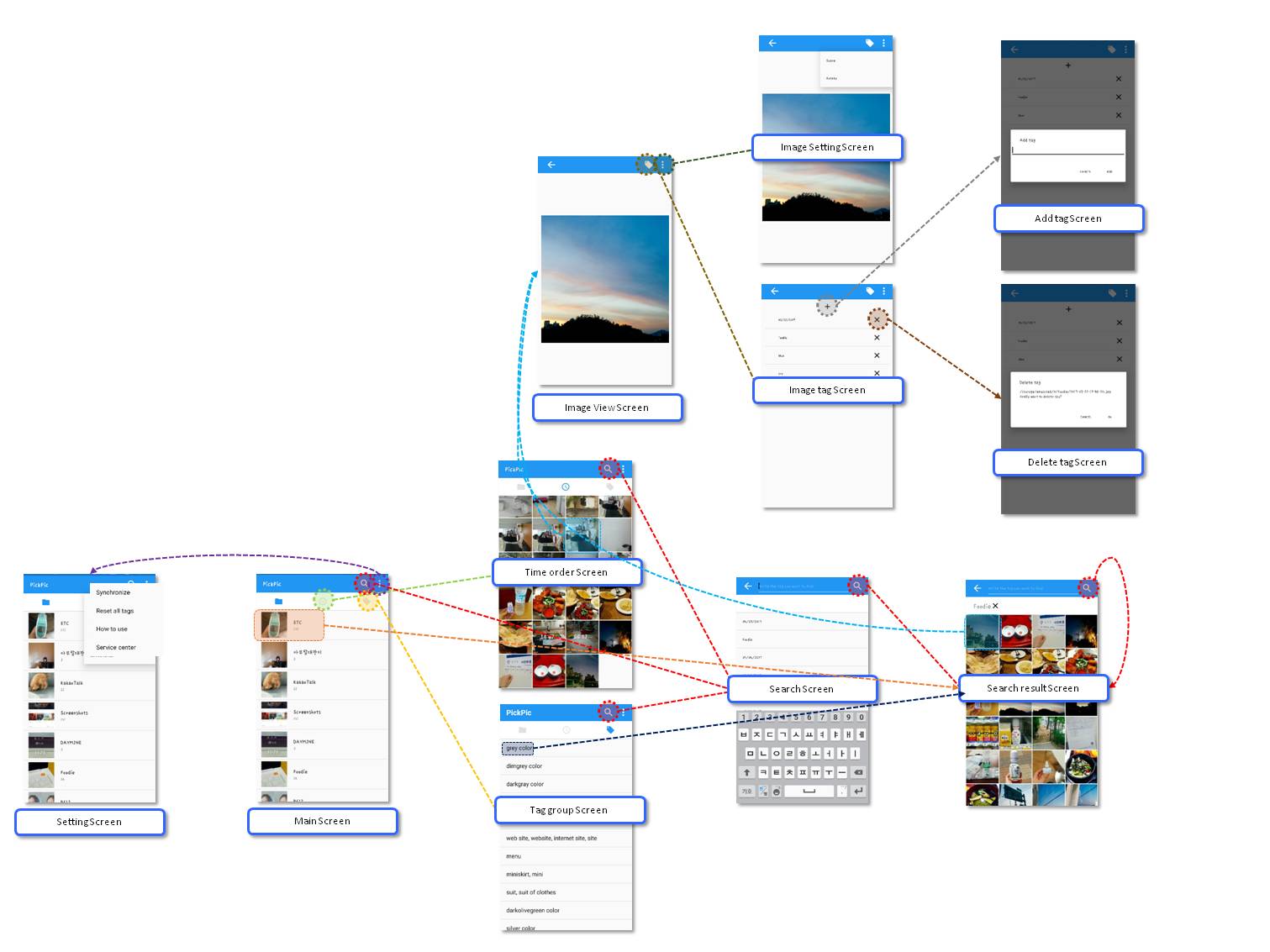
# Pick Pic Icon



**Figure 1 icon a Figure 2 icon b Figure 3 icon c**

The first icon, figure 1, was replaced with figure 2 because the machine arm could not be properly shaped and the background color was not trendy. So, we made the machine arm more angular and chose a trendy gradient background color. However, the background color of figure 2 reminds me of Instagram, so I replaced it with another gradient color. As a result, we made figure 3 and selected it as an icon.

# UI Diagram

****

# Use cases

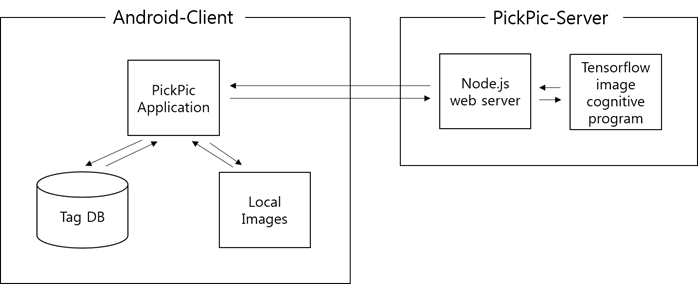
****

Figure System Environment

The program is largely composed of a server and a client. First, the client has a Pick Pic application, a TagDB implemented in SQLite, and local images accessible by the mediastore in the application. On the server, the web server implemented as node.js receives the image from the client, creates a tag using the tensorflow image recognition program, and returns it.

## 4.1 Main

### 4.1.1 Synchronizing

|  |  |
| --- | --- |
| Use Case name | Synchronizing |
| User requirement | Synchronizing between Tag DB and Local images |
| Trigger | When the user launches the application |
| Brief Description | When the user launches the application, synchronization between the local images and the DB is automatically performed. |
| Initial step-by-step description | 1. Import a list of images stored in the tag database  2. Import image list from local repository  3. Delete the photos in the tag database that are not in the local repository but in the tag database.  4. An image in the local repository that is not in the tag database is sent to the server to generate the tag and store it in the tag database. |

### 4.1.2 Directory tab

|  |  |
| --- | --- |
| Use Case name | Directory tab |
| User requirement | I want to see my images ordered by directory! |
| Trigger | When User click or swap to Directory tab. |
| Brief Description | User can see images by directory name. |
| Initial step-by-step description | 1. Click the directory tab button or swap to directory tab  2. Show directory on user’s device.  3. Show directories in the order of recently added images. |

### 4.1.3 Time tab

|  |  |
| --- | --- |
| Use Case name | Time tab |
| User requirement | I want to see my images ordered by time! |
| Trigger | When User click or swap to Time tab. |
| Brief Description | User can see images ordered by time. |
| Initial step-by-step description | 1. Click the time tab button or swap to time tab  2. Show all images on user’s device.  3. Show images in the order of recently added images. |

### 4.1.4 Tag tab

|  |  |
| --- | --- |
| Use Case name | Tag tab |
| User requirement | I want to see all tags. |
| Trigger | When User click or swap to tag tab. |
| Brief Description | User can see all tags in order. |
| Initial step-by-step description | 1. Click the tag tab button or swap to tag tab  2. Show all tags in user’s images.  3. Show tags in order. |

### 4.1.5 Menu

|  |  |
| --- | --- |
| Use Case name | Menu |
| User requirement | I want to set my application and give an opinion to developer.  I forgot how to use the application. |
| Trigger | When User click menu button. |
| Brief Description | User can see menus called ‘Synchronize’, ‘Delete all tags’, ‘How to use’, ‘Service center’. |
| Initial step-by-step description | 1. Click the menu button.  2. Show 4 menus called ‘Synchronize’, ‘Delete all tags’, ‘How to use’, ‘Service center’.  3. Execute synchronize when ‘Synchronize’ is clicked.  4. Delete all tags in users images when ‘Delete all tags’ is clicked.  5. Show tutorials how to use the application when ‘How to use’ is clicked.  6. Show developers’ information and sites that can provide feedback. |

### 4.1.6 How to use

|  |  |
| --- | --- |
| Use Case name | How to use |
| User requirement | I want to know how to use this application. |
| Trigger | When user click ‘How to use’ on the menu button. |
| Brief Description | User can see how to use this application through manual images. |
| Initial step-by-step description | 1. Click the ‘how to use’ button on menu button.  2. Show manual images how to use this application when ‘How to use’ is clicked.  3. Tutorial images are shown and it terminate when user click back button. |

## 4.2 Search

### 4.2.1 Search

|  |  |
| --- | --- |
| Use Case name | Search |
| User requirement | I want to search the image I want! |
| Trigger | When User click Search button |
| Brief Description | User can search the Image by tag |
| Initial step-by-step description | 1. Click the search button  2. Show the tags that system recommend.  3. Show automatically complete tag list to help user search the tag.  4. Show searched image gridview. |

### 4.2.2 Repeatedly search in searched result

|  |  |
| --- | --- |
| Use Case name | Search repeatedly in searched result |
| User requirement | I want to search the image by several tags! |
| Trigger | When User click Search button again after search once(or several time) |
| Brief Description | User can search repeatedly in searched result |
| Initial step-by-step description | 1. There are already searched image result and there are already searched tags above.  2. If User click search button again, there Search Screen is shown and user can search again in same way.  3. Tags that user searched is shown above and the image that have the tags common is shown of the result of the search. |

## 4.3 ImageViewer

### 4.3.1 Image View Screen

|  |  |
| --- | --- |
| Use Case name | Image View Screen |
| User requirement | I want to see a bigger image. |
| Trigger | User clicked a thumbnail image at girdview. |
| Brief Description | Change to full-screen image view screen. |
| Initial step-by-step description | 1. User clicked a thumbnail image at gridview.  2. Show full-screen image to user.  3. User can zoom in / out picture. |

### 4.3.2 Image Tag Screen

|  |  |
| --- | --- |
| Use Case name | Image Tag Screen |
| User requirement | I want to see the list of tags about this image. |
| Trigger | C:\Users\Arsene holmes\AppData\Local\Microsoft\Windows\INetCacheContent.Word\그림1.pngUser clicked button. |
| Brief Description | Change to tag-list screen |
| Initial step-by-step description | 1. User clicked a button.  2. Show the list of tags to user.  3. User can add / delete tag action here. |

### 4.3.2.1 Add Tag Screen

|  |  |
| --- | --- |
| Use Case name | Add Tag Screen |
| User requirement | I want to add new tags. |
| Trigger | User clicked + button. |
| Brief Description | Add tag to DB and show it to user |
| Initial step-by-step description | 1. User clicked + button.  2. A guide window is popup.  3. User enter the new tag.  4. User can click ADD button or CANCEL button.  5. When user clicked ADD button, new tag is added on the list.  6. When user clicked CANCEL button, there is no change. |

### 4.3.2.2 Delete Tag Screen

|  |  |
| --- | --- |
| Use Case name | Delete Tag Screen |
| User requirement | I don’t like this tag. I want to delete it. |
| Trigger | User clicked x button next to the tag that user want to delete. |
| Brief Description | Delete Tag at DB and show it to user. |
| Initial step-by-step description | 1. User clicked x button that is next to the tag that user want to delete.  2. An alert window is popup asking if you really want to delete it.  3. User can click OK button or CANCEL button.  4. When user clicked OK button, the tag is disappeared at the list.  5. When user clicked CANCEL button, there is no change. |

### 4.3.3 Image Setting Screen

|  |  |
| --- | --- |
| Use Case name | Image Setting Screen |
| User requirement | Are there any other options I can do in gallery? |
| Trigger | User clicked menu button above. |
| Brief Description | Show functions that user can use. |
| Initial step-by-step description | 1. User clicked menu button.  2. User can see Share, Rotate, Delete button.  3. User choose the action. |

### 4.3.3.1 Share Image

|  |  |
| --- | --- |
| Use Case name | Share image |
| User requirement | I want to post this image on SNS. |
| Trigger | User clicked ‘share’ button at menu. |
| Brief Description | Share image to other application. |
| Initial step-by-step description | 1. User clicked Share button.  2. Choose the application(Kakao talk, Facebook etc).  3. Share image. |

### 4.3.3.2 Rotate Image

|  |  |
| --- | --- |
| Use Case name | Rotate image |
| User requirement | The image is rotated. I want to rotate it again. |
| Trigger | User clicked ‘rotate’ button at menu. |
| Brief Description | Rotate image and show it to user. |
| Initial step-by-step description | 1. User clicked Rotate button.  2. Image rotates 90 degrees clockwise.  3. User can see image is rotated. |

# Process description

## 5.1 Software toolset-client

### 5.1.1 Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA . On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps

- <https://developer.android.com/studio>

### 5.1.2 ION

Android Asynchronous Networking

- <https://github.com/koush/ion>

## 5.2 Software toolset - server

### 5.2.1 tensorflow(inception v3)

Inception v3 is trained for the ImageNet Large Visual Recognition Challenge using the data from 2012, and it can differentiate between 1,000 different classes, like Dalmatian or dishwasher.

- <https://goo.gl/U8QG8m>

### 5.2.2 dominant color

Get the dominant color from an image

- <https://www.npmjs.com/package/dominant-color>

### 5.2.3 color-namer

Color Namer is an npm package for use in Node.js or the browser that calculates color distance using the Delta-E color difference technique. Given a color in Hexadecimal RGB, RGBA, HSL, or HSV format, it converts the color to the Lab\* color space, then calculates the color's Euclidean distance from a set of colors with known names.

- <https://www.npmjs.com/package/color-namer>

### 5.2.4 express

Fast, unopinionated, minimalist web framework for node.

- <https://www.npmjs.com/package/express>

### 5.1.5 python-shell

A simple way to run Python scripts from Node.js with basic but efficient inter-process communication and better error handling.

- <https://www.npmjs.com/package/python-shell>

### 5.1.6 textract

A text extraction node module.

- <https://www.npmjs.com/package/textract>

## 5.3 Group organization(roles)

**MyungSeoGyo** Middle / Android Backend

**LeeWonJun** Backend / DB, Server

**LeeYeWon** Frontend / GalleryActivity

**KangSoYeon** Frontend / SearchActivity

**ParkGunHoo** Frontend / MainActivity

## 5.4 Risk Summary

### 5.3.1 Synchronizing problem

This application require synchronizing in case of first install this and many new photos added. So little time should be taken

### 5.3.2 Data management

In synchronizing, some Data fee will occur.

So we recommend you to use Wi-Fi in that process

### 5.3.3 Text recognition and object recognition accuracy problem:

Due to time and cost constraints, sufficient data acquisition and learning are difficult. So, accuracy may be somewhat reduced. However, this can be solved by learning and getting the data.

### 5.3.4 Server performance limitation issues

The performance of the currently running server computer is poor, so image analysis that can be processed at once is limited. This can be solved by having a better server.

## 5.5 Schedule / Timeline

**April 30** Make a prototype of the application.

1. Save the tags returned from the Server
2. Coding base backend function

**May 30** Complete detail functions

1. Sorting images – by tags, by dates, by directory
2. Make application’s own gallery
3. Share function
4. Add tag directly by user
5. Searching function
6. Make tutorial for user

**June 2** Complete final application

# 6.0 Meeting reports

We had customers meeting at March 29th. We talked about the background and purpose to our customers. After that, we explained the functions we are planning to make to them. They gave us some feedbacks.

## 6.1 Talk about

* Gallery with search function
* Show images by directory, time, tag
* Search images by tag
* Show results of search by thumbnails
* Open gallery when image is clicked
* Zoom in / out, Delete function is available

## 6.2 Feedback

* Add share function to other SNS like Kakao Talk
* Add tag directly by users
* Show detail tutorial to users

1. How to use basic function of ‘PickPic’
2. How to search picture more accurately