Individual Report of CAN301 Group Project: Project OneClick

Lincheng Shi
Department of Computer Science and Software Engineering
Xi'An Jiaotong Liverpool University
Lincheng.Shi19@student.xjtlu.edu.cn

Abstract

To help users access some mobile functions more quickly and directly, our group has developed an Android app called OneClick. In this report, I will briefly introduce this app. Besides, I will evaluate the gap between the features of this app and that of expectation, as well as the synchronization between the features. The most important part is improvement. I put forward my own ideas on current features and UIs, including adding, expanding, and reworking some features and UIs.

1. Introduction

According to the survey, more than 675,000 apps have been uploaded to Google Play and each Android user needs to use 35 different applications on their mobile phone until 2014 [1]. More and more apps with similar icons make it difficult for some users who do not like managing mobile apps to find the apps they want to open. On top of it, different epidemic prevention and control policies require people to constantly switch between different apps. This can sometimes be very irritating. Although the "search apps" function of the system is very practical, some people, especially the elderly, may have problems in typing or pronunciation. We hope to find a faster and more convenient way to open different apps.

Based on the above content, we developed a hub app: OneClick, which can access some functions by plotting gestures. In addition to quickly opening other apps, we added the functions of fast dialing and opening website. It also allows users to quickly find and open OneClick itself at any time through the floating window.

2. Evaluation of Features

2.1 Progression of Features

At the beginning of our development, we planned many features. Most of the features of OneClick worked as expected. For example, we hoped that the app would keep alive when losing focus so that users can skip the step of loading page (Figure 1) when they need to directly access our app on the screen in some way. According to the official document of Android development published by google, we made our app send regular notifications to realize it.

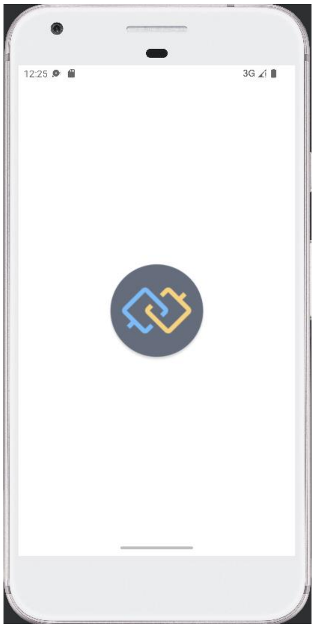


Figure 1. Loading page

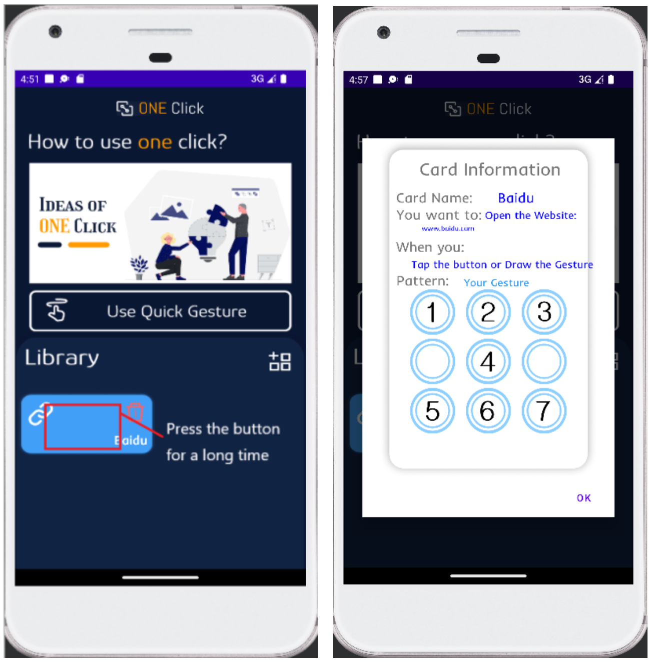
Another example is that we wanted to store the gesture users set so that our app can automatically compare it with the gesture which will be plotted in quick gesture later. We finally decided to number the 9 points into 1-9 respectively, and then store the digital representations of the gesture into List<Interger> in order. There are many other examples.

Although two features did not be realized as expected, we all adopted alternative schemes. The first one is that we originally hoped our app could pull information from all other applications, but this would cause some problems for some applications at the systematic level. We finally created a filter to remove these applications.

The second one is that we originally wanted to open our app quickly by pressing the up and down volume buttons at the same time, but this method is not feasible. Because it is difficult to press these two keys at the same time. Once these two keys cannot be pressed at the same time, the system will pop up the icon of volume display. Meanwhile, the combination of other keys has been occupied by other functions. Therefore, we proposed to pop out a canvas after the user unlocks the screen, and quickly open our app according to a specific gesture. However, this solution seems to have a little impact on the user experience and the canvas cannot appear when needed, so it was also abandoned. Finally, we decided to use the floating window to complete this feature. It can not only quickly open our app, but also adjust the location according to your own needs.

2.2 Synchronization of Features

The internal features in our app synergize well with each other. First of all, there will be no conflict between setting gestures and using quick gestures, and two gestures can be paired. In addition, three pages can be switched well and corresponding information can be recoded properly when users create a new action. Finally, as shown in Figure 2 and Figure 3, users can press a button for a long time to view all the information corresponding to the button (pair). The information includes the name of the button (pair), the corresponding function, whether there is a gesture and the corresponding gesture. This information is consistent with the information when the new action is created.



Card Name: Baidu
You want to: Open the Website: www.baidu.com
When you: Tap the button or Draw the Gesture
Pattern: Your Gesture

Figure 2, 3. Demonstration diagram

3. Improvement

Although the features and UIs of OneClick have fulfilled the three main goals proposed in our group report, there are still many features and UIs can be improved. If we have more time, we must do some of these changes.

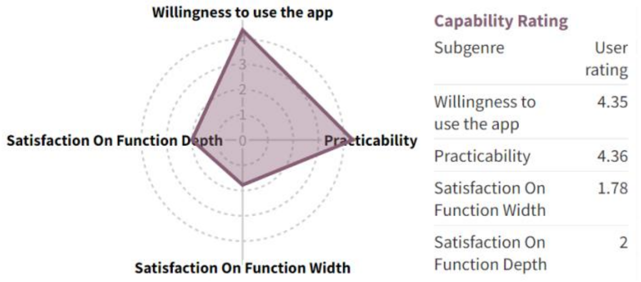
3.1 Features and UIs to Be Added

Nowadays, many people may have dozens of apps on their mobile desktops, but they don't like to classify and manage them. Therefore, the desktop may be overwhelmed by too many icons, making it difficult to play its original role [2]. Since users can quickly open other apps through OneClick, I think OneClick can be added a feature of hidden app icons. Users can choose to hide all app icons except applications at the systematic level with one click, or they can choose to hide app icons that have been added to the Oneclick as a shortcut. This will make the user's desktop look tidier and more comfortable.

Considering the context awareness and we have designed two versions of the UIs, I think we can add a feature of switching between night mode and day mode according to light sensor (Sensor.TYPE_LIGHT) or time. When the light intensity is low or the time is from 8pm to 6am, OneClick switched to the night mode. Otherwise, OneClick switches to day mode. Luckily, our two versions of UIs are black and white. Therefore, no too many changes are required in UIs.

3.2 Features and UIs to Be Expanded

According to the results of usability test from 14 participants (Figure 4), in addition to expanding the width of features, they also hope to expand the depth of features. I considered the following aspects.



Capability Rating

Subgenre	User rating
Willingness to use the app	4.35
Practicability	4.36
Satisfaction On Function Width	1.78
Satisfaction On Function Depth	2

Figure 4. Evaluation result of capability

With more and more apps appear on users' mobile phones, the time cost of finding the app they want to open will be higher and higher. As a result, I want to make OneClick be able to sort the pulled apps into alphabetical order. For example, the app name with the initial “a” is placed under “a” class. Similarly, the app name with the initial “b” is placed under “b” class. This can improve the efficiency of using OneClick for users.

Furthermore, in order to open some sub-functions of apps like health code in Alipay more conveniently, a feasible way is to allow OneClick to open the API of health code. Currently, OneClick can only open the website, because we use the regular formula to limit the format of the website when action is set as “Open website”. We need to reset regular formula or separate a new button "Open API".

3.3 Features and UIs to Be Reworked

Improving user experience is very important in developing apps. Therefore, I decided to rework the following aspects selectively.

First of all, the interface of “card information” must need to be reworked. Actually, this interface is a dialog box and it can be seen in Figure 3. The font size in it needs to be unified. In addition, the title "card information" and the guidance information like "Card Name:" and "You want to:" of this dialog box can be changed more specifically and clearly.

What’s more, I think OneClick allows buttons (pairs) to be sorted automatically by the number of times they are used instead of by their creation time. The buttons (pairs) that are used more often are at the top of the library. Besides, I think button (pair) colors can be unified rather than randomly assigned. For example, dialing can be set in purple, opening apps can be fixed in blue, and open websites can be fixed in yellow. These two points can improve the efficiency of using OneClick for users.

4. Conclusion

In conclusion, this report has illustrated the basic information of OneClick, evaluation of features and improvement of features and UIs. Although some improvements have been proposed, some are my personal views. If we have more time, we don't necessarily need to implement it all.

REFERENCES

[1] A. Tarute, S. Nikou, and R. Gatautis, “Mobile application driven consumer engagement,” Telematics and Informatics, vol. 34, no. 4, pp. 145–156, 2017. [Online] Available: https://www.sciencedirect.com/science/article/pii/S0736585316307006

[2] B. Jimmy, “Understanding the Importance of Desktop Icons Management Tool”. [Online]
Available: https://incrediblethings.com/tech/understanding-the-importance-of-desktop-icons-management-tool/