

A Functionality Testing Method Based on Exploratory Testing for Android Mobile Apps

Jiujiu Yu1*, Jishan Zhang1, Ning Wu1, Yingying Mei1, Deqing Zhang1, Lili Zhu1, Canglu Zhu1 and Shaomin Zhu2

1College of Computer Engineering, Anhui Sanlian University, Hefei, Anhui, 230601, China
2School of Software Engineering, Tongji University, Shanghai, 201804, China
*Corresponding author’s e-mail: jsjgc@slu.edu.cn; yjjyjL@163.com

Abstract. The research is based on the characteristics of mobile Applications (Apps), taking an Android mobile App of a small examination system as an application case, this study is devoted to exploring a feasible functionality testing method based on exploratory testing for Android mobile Apps, which includes fundamental functionality logic verification, Graphical User Interface (GUI) automated testing, and special testing. Furthermore, further work is put forward to apply this functionality testing method to IOS mobile Apps, adding new test items for mobile Apps and improving the method of depth-first search for control traversal in GUI automated testing.

1. Introduction

With the rapid development of mobile terminal, mobile applications are gradually infiltrating all aspects of people's lives and work [1]. A variety of mobile applications in Internet are changing social life in the information period. All kinds of mobile applications and service are mainly for individual consumers. From the perspective of software engineering, it is required to be released and tested quickly, and user feedback is obtained in time. Mobile Apps (Applications) are significantly different from conventional software in UI mode, operating habits, network environment, user specifications, etc., if there are problems with the quality of the APPs, the impact may be greater than that of conventional software [2]. When testing for mobile APPs, except for unit testing for the code, functionality testing for the system, compatibility testing, interactivity testing, user experience testing, power consumption testing, network data testing, network connection testing, performance testing and stability testing are included, which are demonstrated in [3]. For mobile APPs, comprehensiveness of the testing is reflected in functionality testing, special testing and performance testing, but test efficiency is reflected in automated testing and auxiliary testing [4].

There are many differences between mobile APPs and conventional Internet software based on PC, such as the diversification of device screen sizes and resolutions, the diversification of system versions, and the difficulty in data inputting. Most of the mobile APPs rely on the service which is provided by the background interface to achieve functions, and the interface display effect of the APPs is implemented on terminal mobile device. Therefore, functionality testing of mobile APPs is needed to be tested for application interfaces [5]. Moreover, mobile APPs have the remarkable characteristic of individuality for user experience, and the individual experience of mobile APPs has its own clear aims for providing users with specific service [2].

Published under licence by IOP Publishing Ltd
As a free-style and heuristic testing idea, more hidden defects of usability and functionality for implicit requirements can be detected by software exploratory testing, and it is available for rapid development environment currently [6]. Android platform is a major mobile terminal platform for Apps which is widely used. Therefore, this paper devotes to the researching on a feasible functionality testing method based on exploratory testing for Android mobile Apps. The structure of the paper is organized as the following: Section 2 describes the concept of related technologies of mobile Apps testing. Section 3 describes how to implement on functionality testing based on exploratory testing. An application case of functionality testing of a mobile App on Android platform is given in Section 4. Section 5 concludes the paper and puts forward the further work.

2. Concept of related technologies

2.1. Software exploratory testing

Software exploratory testing is a kind of testing idea or testing thinking style based on experience that is not limited by any idiographic testing technique [7]. Accurately, exploratory testing prefers to a test idea of exploratory, which can be freely applied to any phase of testing activities, and it is not restricted by a certain testing method [6]. Table 1 demonstrates the three categories of exploratory testing method. The test value, test content and test steps of each method are described in [8-10].

| Categories of exploratory testing method | Name for each exploratory testing method |
|----------------------------------------|-----------------------------------------|
| Individual feature                     | arrogant American tour, super model tour, clubbing tour |
|                                        | bad neighbour tour, couch potato tour, rained-out tour |
|                                        | supporting actor tour, antisocial tour, saboteur tour |
|                                        | obsessive-compulsive tour, test one get one free tour, ...... |
| Multiple feature                       | landmark tour, FedEx tour, prior version tour |
|                                        | garbage collector’s tour, museum tour, back alley tour |
|                                        | longer businessman tour, ...... |
| Exploration based on system interaction| business district, slum, historical district |
|                                        | hotel zone, tourist area, entertainment area |

2.2. Model of test automation pyramid

Figure 1 demonstrates the model of test automation pyramid. Exploratory testing should be reflected in manual tests. In Figure 1, the lower two layers of App tests are code level testing which involve testing of each class, each method, relationship of between classes, and the App Interface (API) tests. Table 2 demonstrates the categories of API tests, and automated testing is implemented on App GUI tests. Various elements of button, menu, dialog box, etc. on UI are identified through testing tools to complete App GUI tests.

| Categories of API tests | Names for interface protocol |
|------------------------|-----------------------------|
| Web interface          | Based on HTTP               |
Based on web service: SOAP/REST/JSON-RPC/XML-RPC/……

Non-web interface | Some of the customized communication interface

### 3. Functionality testing based on exploratory testing

#### 3.1. Fundamental functionality logic verification

For a mobile App, no matter what kind of terminal mobile device it runs on, fundamental logic functions of installation, permission, registration and login, uninstallation, running are indispensable to be tested and verified. Table 3 demonstrates major test content of each fundamental function.

| Fundamental functionality logic verification | Major test content |
|---------------------------------------------|--------------------|
| Installation                                | Whether the App can be installed on the mobile device correctly or not? /Installation path can be customized? /Without the user's permission, the App can be started automatically? /During the software installation process, whether the handling of unexpected situations (crash, restart, power failure) conformance the requirements or not? /…… |
| Permission                                  | Is the App allowed to obtain user data? /If there is risk of privacy disclosure in the App? /User authorization level of the App can be checked? /…… |
| Registration and login                      | Is the App can be login in when the username and password are wrong? / Is there user login cache? / Security of the password is ensured? / User password can be retrieved if it is missed? / Is it correct to login in the App with a disabled account? /…… |
| Uninstallation                              | Whether there is a warning message or not when a user deletes the installation folder to uninstall directly? /How to deal with unexpected situations during app uninstallation? /Does uninstallation support cancel function? /…… |
| Running                                     | Testing of running status after the App is installed. / Whether the switching between pages of the App is smooth, whether the logic is correct, etc. / If an App can be switched between foreground and background correctly? / An App is started after user unlocking the lock screen and test whether it will be crashed? /…… |

#### 3.2. GUI automated testing

In the model of test automation pyramid, the top layer is GUI tests that test targets are User Interface (UI) controls which are implemented by automated testing. Control traversal method for GUI automated testing is the major method recently. A generation method of control traversal under the guidance of standard path is proposed in [11], which is referenced in this paper. In this method, a test script was recorded as a standard path, the control in UI was automatically acquired and the control relationship graph based on depth-first search was generated for traversing the control relationship graph to generate test cases [11]. Figure 2 demonstrates the automated testing framework of this method, communication from platforms of Android and PC by ADB (Android Debug Bridge) and Socket.
3.3. Special testing

Special testing refers to testing for some special items of mobile Apps which are different from conventional software on PC platform. Mobile Apps are running based on mobile platforms (such as Android), special testing should be implemented for special usage scenario by manual testing that based on exploratory testing. On the top layer of the model of test automation pyramid, the implementation of exploratory testing can find more defects in functionality and ease of use [12]. Table 4 demonstrates major special test items of Android Apps.

Table 4. Major special test items of Android Apps.

| Special test item       | Test description                                                                 | Major tool (file)                |
|------------------------|----------------------------------------------------------------------------------|----------------------------------|
| Electricity consumption| How an App uses the battery, includes the detail of wake lock.                   | GSam Battery Monitor Pro         |
| Flow capacity          | Connection method of wireless network. Monitor of network flow capacity. /…….     | Tcp_snd/Tcp_rcv                  |
| Crossover events       | Do multiple Apps running at the same time affect normal functions?               | Android SDK                      |

4. Application

In this section, functionality testing for an Android mobile App of a small examination system is given as an application case. Limited by paper length, functionality testing for "save test record" model is given. Table 5 demonstrates test method of this model.

Table 5. Test method of "save test record" model.

| Description of "save test record" model | Note: In this model, test records and test records of all students (user information, test time, number of wrong questions, etc.) can be saved automatically [13]. |
|----------------------------------------|-------------------------------------------------------------------------------|
| Functionality logic verification       | Permission                                                                   | Pass ✓                           |
|                                       | Running                                                                       | Fail □                          |
|                                       | Data update                                                                   | Pass ✓                           |
|                                       | Functionality interaction                                                    | Fail □                          |
|                                       |                                                                              |                                  |
|                                       | Functionality interaction                                                    | Pass ✓                           |
|                                       |                                                                              | Fail □                          |
|                                       |                                                                              |                                  |
|                                       |                                                                              |                                  |
### GUI automated testing

| Control          | Test Description                                                                 | Pass | Fail |
|------------------|----------------------------------------------------------------------------------|------|------|
| "save records"  | Button test under the guidance of standard path [11]. Acquire the input event during the operation, and Android service class is used for UI controls. | ☐√   | ☐    |
| "cancel" Button |                                                                                  | ☐√   | ☐    |
| "submit records" Button |                                                                                | ☐√   | ☐    |

#### Special testing

| Test Description                                                                 | Test Details                                                                 | Pass | Fail |
|----------------------------------------------------------------------------------|-----------------------------------------------------------------------------|------|------|
| Electricity consumption                                                         | (exploratory testing method: obsessive-compulsive tour/couch potato tour/supporting actor tour/super model tour) | ☐√   | ☐    |
| Flow capacity                                                                    | (exploratory testing method: antisocial tour/FedEx tour/back alley tour/garbage collector’s tour) | ☐√   | ☐    |
| Crossover events                                                                 | (exploratory testing method: business district/entertainment area)           | ☐√   | ☐    |
| CPU/sensor usage/App wake locks/wake time/kernel wake locks/...                   |                                                                             | ☐√   | ☐    |
| Preferred network/automatic download/flow statistics/...                         |                                                                             | ☐√   | ☐    |
| Network switch/browse network/phone calls/send messages/...                      |                                                                             | ☐√   | ☐    |

### 5. Further work

The automation of mobile application testing is a major trend in mobile application testing, and more and more companies have such requirements [14]. How to apply the feasible functionality testing method based on exploratory testing in this paper to IOS mobile Apps is further work. Furthermore, existing research work needs to be further improved. New test items on the basis of functionality test should be added for mobile Apps, such as performance test, reliability test, security test, memory leak test, and improving the method of depth-first search for control traversal in GUI automated testing.

### Acknowledgments

The work was supported by "Project of Natural Science of Anhui Province University" under Grant No. KJ2020A0809, "Project of Quality Engineering of Anhui Province University" under Grant No. 2019jxtd122, "Excellent Young Talent Support Project of Anhui Province University" under Grant No. gxyq2019138 and "Project of Quality Engineering of Anhui Sanlian University" under Grant No. 20zlgc076.

As the corresponding author of this paper, I would like to express my heartfelt gratitude to Prof. Shaomin Zhu of Tongji University and the whole authors of the references in this paper.

### References

[1] Du, C.Y., Wu, J.H., Song, W. (2017) Application Software Performance Test Based on Mobile Terminal. Electronic Technology & Software Engineering, 03: 67-69.

[2] Song, Y.M. (2016) the Research and Design of Test Method for Mobile Internet Applications. Thesis of Master's Degree of Beijing University of Posts and Telecommunications, Beijing.

[3] Zhu, S.M. (2016) Software Testing (Second Edition). POSTS & TELECOM Press, Beijing.
[4] Yueminw. (2018) Summary of Mobile Test Technology. https://testerhome.com/topics/11512?order_by=like&.
[5] Zhu, S.M. (2019) Whole Process of Software Testing (Third Edition). POSTS & TELECOM Press, Beijing.
[6] Yu, J.J., Zhang, J.S., Chen, Y., etc. (2020) A Design Method of Reusable Test Case Based on Exploratory Testing for E-Commerce Website. In: International Conference on Information Science, Parallel and Distributed Systems. Xi’an. pp. 196-199.
[7] Yu, J.J., Zhang, Y.S. (2014) A Survey on Software Exploratory Testing. Journal of Research and Exploration in Laboratory, 33: 93–102.
[8] Shi, L., Gao, X. (2012) Practice Road of Exploratory Testing. Publishing House of Electronics Industry, Beijing.
[9] Hendrickson E. (2014) Explore It! Reduce Risk and Increase Confidence with Exploratory Testing. China Machine Press, Beijing.
[10] Yu, J.J. (2018) Research on Exploratory Testing Methods of Special Learning Website. Journal of Tonghua Normal University, 39: 55–59.
[11] Zhang, M., Cheng, B.L., Zha, W.Z., etc. (2018) Method of App Function Testing for Andriod Mobile Phone. Computer Engineering and Design, 39: 684-689.
[12] Yang, Y., Liu, Z., Cai, J.T. (2020) Research and Practice of Software Testing Based on Exploratory Testing. Computer Applications and Software, 37: 29-33.
[13] Yu, J.J., Zhang, J.S., Sheng, Y.L., etc. Design and Implementation of a Small Examination System Based on Scrum. unpublished.
[14] Feng, G., Li, N.G. (2017) Model-driven Testing for Mobile Applications. Computer Science, 44: 233-239.