

# Individual Report

of Autism Children Story Telling Application

Liu JinKua (1098399)

[jinkual@kean.edu](mailto:jinkual@kean.edu)

## **Introduction**

Software engineering is the one of the core majors in computer science since a good software engineer should not only have a well command of the knowledge from the book but also consider the reality development environment with considerate planning. As the mobile technology has rapid development, more and more software and applications has established based on mobile platform, and the mobile software engineering has emerged. And disabled-aware engineering also become important as the internet of things (IoT) gradually go into disabled person's life and serve them a better-quality life experience, educations and life security. This article would mainly discuss these two engineering and their several unavoidable issues. Also combining the project and the knowledge author has learnt in this semester to have a comprehensive reflection.

### **1. Mobile Software engineering (MSE)**

Software engineering is key issues in the software development. Moreover, as the mobile technology developed and implement rapidly, the increasing of the capability, functionality, and user experience of the mobile lead to the mobile software engineering (MSE) as an unavoidable subject for both companies and schools. The same as the software engineering the mobile software engineering is a compressive subject that the engineer would have the ability to considerate several issues from the architectural models, database or information models, functional & non-functional requirement, human-computer interface including graphic user interface (GUI) and navigation models and the marketing issues. (Uskov, 2013) And this part, author would emphases several issues in MSE

#### **1.1 Marketing environment**

In our class project, the target users are the mainly focus on the autism kids, their parents and their teachers. As the COVID-19 outbreak, the software engineer should reconsider the issues of the application provide for the disabled people. According to the Bloomberg (2021), there's more than 90% of the website is accessible for the blind people and the user increasement has risen about 33% for the application designed for the disabled people during the pandemic. Since majority mobile application is still lacking focus on the disabled people, the market in this field is still profitable and in widely demand.

## **1.2 Platform and the Development tools**

The platform and the development tools are also important in the MSE since difference platform and development tools would influence performance such as the security, power consuming, complexity of testing, development speed. In our project we have implement the Unity and the Visual Studio Code. However, this may be the ideal platform. Even though applying Unity would decrease the time of the developing since the GUI and part of the function could be automatically coded. And the portability would also be enhanced since Unity support both Android and IOS and also minimize the native code (Wasserman, 2010). However, the complexity of testing and the power consuming may significantly increase, and the scalability and security would be decreased.

## **1.3 Computing Resources**

The computing performance is also a noticeable issue that the development of the cloud computing, edge-computing and in-memory computing comparing to the traditional native computing on the CPU would lead to the significant differences among applications. (Wasserman, 2010) For our project, the edge computing may be the ideal way to conduct the education application. Since the traditional native computing and the in-memory computing would require user to have high performance mobile phone and it is may not affordable for the target users' family or institution. As for the cloud computing, this kind of computing architecture would require centralized computing resources which means that the computing would be further to the target user. This would provide a manageable computing. However, the cloud computing may cause the delay in performance and cause some negative affect in the non-functional requirement. The edge computing architecture would require distribute computing resources which means that the computing would be closer to the target users. This would provide a reliable and effective computing performance.

## **1.4 Gamification**

Gamification is still in initial stage but appears strong potential in software engineering and mobile software engineering. Since proper gamification would enhance the user engagement, motivation and performance while making the application more appealing. (Pedreira,2015) According to Pedreira's research most of the gamification currently focused on the badges, points and ranking to enhance the user experience.

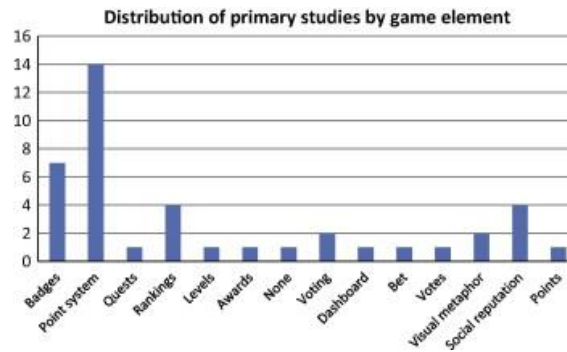


Figure 1: Distribution of primary studies by game element (Pedreira,2015)

However, gamification is a comprehensive topic since developer should understand the nature of the application, characteristic and profile of the target user, the environment of using the application and the appropriate gamification approaches. Moreover, since the gamification in software engineering or mobile software engineering is still at the preliminary stage and some researchers even predicted that “80% of the gamified applications will fail to meet their business goals due to a poor design”. (Gartner, 2008) Implementation of the gamification should be designed after considerate research. The gamification in education software has already exists using HuJiang KaiXin CiChang as an example which is an Japanese word learning application.



Figure 2&3: The level selection interface and sign in page of HuJiang KaiXin CiChang

This application has applied the point, level and ranking gamification elements to divide the user with different learnability and promote the user motivation and experience. Back to our project, the gamification may suitable and applicable to our application due to it is an educational software for the Autism kids. Our application could also implement the level element to identify the learnability of the kids seamlessly and enhance the motivation of the kids.

## **2. Disabled-aware Software Engineering**

Traditional software engineering would focus more on the target user without disabilities and the disabled users seem to be easily overlooked. So disabled-aware software engineering becomes vital importance in the software developing procedures for disabled persons to enhance the software more accessible and usable. Here, the author would briefly discuss two concepts of the disabled-aware software engineering for autism.

### **1.1 Concept of Simplification & Personalization**

#### **1.1.1 Concept of Simplification**

Due to the features of the autism spectrum disorder (ASD), it would be difficult for the autism person to perform the normal interactions, learning process and afford changes in surrounding environment and they would lose attention easily. So, using traditional interface to interact with the ASD person would be complex and risky. So, it leads to a concept of the simplify. However, this simplify doesn't mean to cut everything, in certain cases the information redundancy and graphic illustration is in need. The simplify more emphasis on logical simplify, predictability and information simplify. According to Milne's research (2017), the user interface should be predictable, and the interface should display all the features in an easily accessible way. All the tools could be interacted instantly without extra procedure would be appropriate. Moreover, the function of the application should also be simplified there should not have too many functions for the autism user. In our project, we have implemented this idea by cutting all the extra functions in the storytelling interface as long as the kids scan the pictures a sentence or a word with 3D model would appeared. However, this interface illustrates a problem that it is not predictable since we minimize all the UI in the storytelling interface. Adding some indicative pictures or marks would be better. However, what kind of indicative pictures to applied would also be a problem. According to Milne's research, too abstract graphic would be difficult for ASD user to understand and in education the development delay graphics would be suitable for child users. The researcher also elicited an important issue is about the redundancy of information, one content could be expressed in relative ways at the same time. Using relative pictures, audio record and text to present one content. However, this also should aware the information overload, developer should find the balance between the redundancy and simplification. Besides, Milne also emphasized that the

navigation of the application should be simple, consistent and logic. No more than three click, the application should navigate user to the correct page and no error should be occurred. Finally, the word used in the application should be concise. In our project, we have applied 50 development delay graphics and displayed all the level over one select menu. User could access the storytelling interface within 2 clicks. However, the navigation of the pictures in the book is still no well-organized. Future development is needed.

### **1.1.2 Personalization**

Personalization is an effective but challenging topic in the application for the autism. Since different autism would have different thinking approaches and their thinking approaches would be the most effective approaches for them to study. In Pavlov's research, researchers has mentioned that personalization could not be significantly changes in the application, it could be simply larger the space of the text, using different type of material. In Melin's research, researchers also mentioned that the interface should be sizable. In our application, we still have more space in developing the personalization. Future development is also in demand.

### **Conclusion**

Mobile software engineering and disabled-aware engineering is two essential parts in the software engineering. Mobile software engineering would consider more than traditional software engineering due to the lower computing resources, larger marketing environments and enhancing human-computer interface technology. And the disabled-aware engineering is also at an initial stage since the majority of the software-engineering still focus on the people without disable and the traditional development models would also overlook the issues of the disabled. (Nganji,2011) So, disabled-software engineering still needs further research and evaluation to explore new developments models such as user sensitive inclusive design (USID) in order to provide disabled user an accessible and usable software.

### **Future expectation**

As for the project of authors, there's still lots of space to improve from the personalization and gamification to promote target users' motivation and learning efficiency to the navigation of the user-interface and the balance of information redundancy.

## References:

- Antonina Dattol o,Flaminia L. Luccio. (2017). Accessible and Usable Websites and Mobile Applications for People with Autism Spectrum Disorders: a Comparative Study. EAI Endorsed Transactions on Ambient Systems. Reference from: <https://pdfs.semanticscholar.org/d5d0/9f0ab5e1422948ff575123a52da3df3d58ed.pdf>
- Currier, J. (2008). Gamification: game mechanics is the new marketing. OogaLabs Blog.
- LowenkronHadriana. (2021). The Apps That Are Redefining Accessibility. Bloomberg. Reference from: <https://www.bloomberg.com/news/articles/2021-07-29/the-apps-and-technology-helping-people-with-disabilities>
- Nganji, J. T., & Nggada, S. H. (2011). Disability-aware software engineering for improved system accessibility and usability. *International Journal of Software Engineering and Its Applications*, 5(3), 47-62.
- N. Pavlov, "User Interface for People with Autism Spectrum Disorders," *Journal of Software Engineering and Applications*, Vol. 7 No. 2, 2014, pp. 128-134. doi: 10.4236/jsea.2014.72014.
- Oscar Pedreira, Félix García, Nieves Brisaboa, Mario Piattini,. (2015). Gamification in software engineering – A systematic mapping. Reference from: <https://doi.org/10.1016/j.infsof.2014.08.007>.
- V. L. Uskov, "Mobile software engineering in mobile computing curriculum," 2013 3rd Interdisciplinary Engineering Design Education Conference, 2013, pp. 93-99, doi: 10.1109/IEDEC.2013.6526767.