

## **Survey on Developing Android MicroApp in CoT**

**Nishaanth .S, Kamalrajsha .M.R , Ramya G. Franklin\***

Computer Science and Engineering, SathyabamaUniversity , Chennai-600119,

Tamil Nadu, India

\*Faculty of Computing, Computer Science and Engineering, Sathyabama University, chennai - 600119, Tamil Nadu, India

### **Abstract**

**Cloud of Things offers an IoT design for device manufacturers and service providers, which can be up and run within hours. The uses for CoT includes: secure plug and play communication firmware, an IoT device management and control back-end, and an automatically-generated mobile service applications. With Cloud of Things, IoT solution does not need to be developed. To develop various types of services, from Web service and native Smartphone application to domestic services, i.e., services to manage sensors and devices to control temperature, lighting, security system, etc.**

**Keywords- Android Application, CoT, IoT, MicroApp, Mobiles Services.**

### **Introduction:**

Android is an Operating System for mobiles and tablets developed by Google. Android has been the best selling OS. Java Programming Language is used to develop Android Applications since it has full access to Android APIs. Android platform has a number of third-Party Applications which can be downloaded, installed and can also be removed by the android users from the android play store or any other third-party app store. Our Motto is to make the End users develop their own Android application according to their requirements. We develop a Micro-App which makes to users to develop their applications.

### **Survey on Cloud of Things:**

Cloud of Things (CoT) is a concept of Integrating the Internet of Things (IoT) in a Cloud Environment. Jiehan Zhou and Chen Yu has presented a paper about integrating the Internet of Things concept in Cloud Computing environment with the use of a Common Architecture. In

1999, Kevin Ashton predicted that computers in future will be depending upon the data created by the computers rather than the data created by the users. Nicholas Negroponte (head of the Media Lab at MIT) also exhibited that Internet of Things can also be embedded so that the things can get more intelligent and work with the mindset of reducing cost, wastage, and memory.

Both Internet of Things and Cloud Computing are both emerging technologies and have their own functionalities. Things are been linked to their virtual representation on the Internet and can be accessible via the Internet. Some of the important services that are offered by the Cloud Computing operators are SaaS, PaaS, IaaS these are some of the important and most used services that are offered by the Cloud Service Providers. In Cloud Computing, most of the resources are already existed in the internet servers, and can be used by the client side machines like laptops, personal computers. The Cloud-based programming environment has a common benefit that the local host admin and users does not need to spend time with large client and server machine's installations, setups, or software updates. By using the tools for cloud , the user can program from anywhere that has an Internet connection.

#### **Survey on Model Driven Development Pattern:**

The benefit of Cloud-based integrated programming environment is that the local administrators and users do not have to spend much time with large client and server machine installations, or software updates. The basic concept of the Model Driven Architecture is the MVC approach. The Model View Controller (MVC) is a pattern that divides an application into three components: the model, the view, and the controller. Model Driven Architecture (MDA) is used to develop Software as a Service (SaaS). Hongming Cai , Yizhi Gu, and Boyi Xu has said that, purpose of building cloud based mobile applications in a configurable and adaptive manner, Model-Driven Development Patterns based on semantic reasoning mechanism are provided towards CoT application development. The Software development is facilitated by MDA to providing various abstraction levels of the software development process. For the Generation of Mobile Application development the Model driven Pattern is been used for Developing the Application according to the user requirements. The Goal of the MDA is to “program” on higher level of abstraction. This means that you have to specify less and generate more.

#### **Survey on User End Development:**

Using End User Development (EUD) approaches, the developers can develop the system in a way that even the users with no knowledge on programming can develop their own application. Michael Spahn , Christian Dorner, Volker Wulf says that the User End Programming can be very useful for the users who does not have any idea about programming language. They can create their own software according to their needs, in which they does not need any basic knowledge about the programming language for developing the application. End User Development as a term has evolved over time and complements a lot of older research fields denoted by other terms, while having a stronger focus on the development of new artefacts. Hongming Cai , Yizhi Gu, and Boyi Xu also says that when a Model Driven Application has to be build there should be 3 basic concepts (1) Model Driven Architecture , (2) End-User-Development, (3) Software as a Service SaaS. These are the main concepts that are needed for the Developer to develop an Application that will be useful for the User to develop his/her own application.

### **Survey on MicroApp:**

MicroApp Generator lets end-users compose pre-existing applications/services available on the smartphone, the local network and the Web. MicroApp generator is the main concept of generating the mobile applications by the users. This helps the user to develop his own applications based on his needs. Rita Francese, Michele Risi and GenoveffaTortora, Senior member of IEEE, has presented his paper for developing Mobile User End Application. In this paper a MicroApp is been generated by the use of MicroApp Generator.The MicroApp Generator tool improves the effectiveness of the MicroAppin terms of time and editing errors. MicroApp Generator enables the generation of service Applications (i.e., MicroApps) through selecting the necessary services by the user, the services are already present in the device.The MicroAppdeveloper focuses on how the application works, since the environment automatically generates a graphical user interface for the app. In our proposed system the MicroApp Generator will be in a Dynamic Environment. Not just the generation of MicroApp is dynamic but also the generated MicroApp will be in a Dynamic form, i.e., Even after the MicroApp generation the services will be Dynamic so that the Application itself behaves Dynamic. This makes the MicroApp more flexible for the user. At the time of generation if the selected service is not available, then an alternative mobile service which performs almost the same operation will be

added or mapped while generating the user required MicroApp. The MicroApp can be able to represent by the user since the user is the one who generated the MicroApp. So, this behaves as the User End Development application.

### **Survey on Developing Applications Directly from Mobile Devices:**

By the paper exhibited by A. De Lucia, R. Francese, M. Risi, and G. Tortora, a mobile application can be able to generate directly from the mobile phones by the users. End-User-Development and MicroApp concept are useful for developing applications directly from mobile devices. These are the concepts which explain about the generation of mobile application directly from the mobile phone itself. The user has to generate the mobile application by the help of a MicroApp generator in which the user does not need to have any basic knowledge about programming language. We are using a Qrcode as a path for downloading the generated MicroApp. After generation of the android application a keystore file has to be used as a process of user authentication to identify the authorized user. The android generated application will be in .apk extension. S. Cuccurullo, R. Francese, M. Risi, and G. Tortora, also explains in their paper that generation of MicroApp and the uses of the user who generates the application. This paper is a basic concept of generating an application and by the paper said by A. De Lucia, the generated mobile application is then downloaded by the user mobile device itself. In our Proposed System, we use a Qrcode as a path for downloading the mobile application this has been generated.

### **Conclusion:**

This helps the Mobile End User to create their own MicroApp with the help of the available services in the device. The use of generating an application dynamically helps the user to generate the Application in a dynamic environment. Therefore, the user with a routine job could make use of the repeatedly using services.

### **References:**

[1] J. Wong and J. I. Hong, "Making Mashups with Marmite: Towards End-user Programming for the Web," in SIGCHI Conf. CHI. ACM, 2007, pp.1435–1444.

- [2] S. R. Ponnekanti, B. Lee, A. Fox, O. Fox, T. Winograd, and P. Hanrahan, "ICrafter: A Service Framework for Ubiquitous Computing Environments," in *UbiComp*. Springer-Verlag, 2001, pp. 56–75.
- [3] P. Korpiö, E.-J. Malm, T. Rantakokko, V. Kyllönen, J. Kela, J. Mantyjarvi, J. Hakkilä, and I. Kansala, "Customizing User Interaction in Smart Phones," *IEEE*, vol. 5, no. 3, pp. 82–90, 2006.
- [4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comp. Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [5] J. Brnsted, K. Hansen, and M. Ingstrup, "Service Composition Issues in Pervasive Computing," *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 62–70, 2010.
- [6] S. Cuccurullo, R. Francese, M. Risi, and G. Tortora, "A Visual Approach supporting the Development of MicroApps on Mobile Phones," in *International Conference on DMS*, 2011, pp. 171–176.
- [7] A. De Lucia, R. Francese, M. Risi, and G. Tortora, "Generating Applications directly on the Mobile Device: An empirical evaluation," *AVI*, 2012, pp. 640–647.
- [8] E. Christensen, F. Curbera, G. Meredith, S. Weerawarana et al., "Web Services Description Language 1.1," 2001.
- [9] V. R. Basili and D. M. Weiss, "A Methodology for Collecting Valid Software Engineering Data," *IEEE* vol. 10, no. 6, pp. 728–738, 1984.
- [10] R. A. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. USA: Addison-Wesley Longman Publishing Co., Inc., 1999.
- [11] T. Gu, H.K. Pung, D.Q. Zhang, "A service-oriented middleware for building context-aware services", *Journal, Network and Computer Applications*, 2005, pp.1-18
- [12] Zhang, H., Liu, J., Zheng, L., Wang, J.: "Modeling of Web Service Development Process Based on MDA and Procedure Blueprint". 2012 IEEE 11th ICIS, pp. 422-427
- [13] Kang, W., Liang, Y.: "A Security Ontology with MDA for Software Development". 2013 International Conference, pp. 68-74 (2013)
- [14] Spahn, M., Dorner, C., Wulf, V.: "End User Development: Approaches towards a Flexible Software Design". 16th ECIS 2008.
- [15] Hongming Cai, Yizhi Gu, Athanasios V. Vasilakos, Boyi Xu, Jun Zhou "Model-Driven Development Patterns for Mobile Services in Cloud of Things", *IEEE*, Feb 2016.

