Designing A Student Attendance Application Using Mit App Inventor

Achry Rafly Fadlillah, Ferdinandus Cilverius Ola Beda
Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang,
Kota Tangerang Selatan, Indonesia
E-mail : achryrafly@gmail.com , mrnandoid@gmail.com

Abstract Education is the main foundation in community development and human resource development. In the context of higher education, the quality of the learning process and lecture management greatly affects student success. One aspect that has a significant impact on the learning process is student attendance management. Traditionally, recording student attendance is done manually by using a physical attendance list that must be filled in by students and signed by lecturers. Although it has been used for a long time, this method has a number of limitations that need to be overcome. In the rapidly evolving digital era, we have access to technology that can simplify and improve student attendance management. The use of information technology can provide a more efficient, accurate, and practical solution in recording student attendance. The app was developed using MIT App Inventor, a platform that makes it easy for users to create Android apps without in-depth programming knowledge.

Keywords : Design, Attendance, App, Android, MIT App.

Abstrak Pendidikan merupakan landasan utama dalam pengembangan masyarakat dan pengembangan sumber daya manusia. Dalam konteks pendidikan tinggi, kualitas proses pembelajaran dan pengelolaan perkuliahan sangat mempengaruhi keberhasilan mahasiswa. Salah satu aspek yang mempunyai dampak signifikan terhadap proses pembelajaran adalah manajemen kehadiran siswa. Secara tradisional, pencatatan kehadiran mahasiswa dilakukan secara manual dengan menggunakan daftar hadir fisik yang harus diisi oleh mahasiswa dan ditandatangani oleh dosen. Meski sudah digunakan sejak lama, cara ini memiliki sejumlah keterbatasan yang perlu diatasi. Di era digital yang berkembang pesat, kita memiliki akses terhadap teknologi yang dapat mempermudah dan meningkatkan pengelolaan kehadiran siswa. Pemanfaatan teknologi informasi dapat memberikan solusi yang lebih efisien, akurat, dan praktis dalam pencatatan kehadiran siswa. Aplikasi ini dikembangkan menggunakan MIT App Inventor, sebuah platform yang memudahkan pengguna membuat aplikasi Android tanpa pengetahuan pemrograman yang mendalam.

Kata Kunci : Desain, Absensi, Aplikasi, Android, Aplikasi MIT.

PRELIMINARY

Attendance is a record of attendance, part of reporting the activities of an institution, or a component of the institution itself that contains attendance data that is organized and arranged in such a way that it is easy to find and use if needed at any time by interested parties [1]. Attendance is one of the important factors in universities because attendance can support lecture activities. The attendance system that is generally applied so far is located in the collection of student attendance data carried out in class, where the current attendance system still uses paper media (paper based) for recording attendance.

MIT App Inventor is an android editor or opensource web application maintained by the Massachusetts Institute of Technology (MIT). MIT is a tool to create and develop Android-based applications in the form of visual programming, which allows everyone even children to create Android-based applications on smartphones. App Inventor uses a graphical display or interface, which allows users to drag and drop to change logic in the visual form of objects. So

it can be run on a smartphone device. MIT App Inventor is the simplest to use android application editor, making it easy for new users to understand how to design and create android-based applications in a short time or less than 30 minutes. With App Inventor users are able to create android applications ranging from simple to the most complex. As for some applications that can be made, namely, map applications, music player applications, applications to help traffic, simple cashier applications, robot control systems, and many others.

Based on the above problems, we take a title "DESIGNING A STUDENT ATTENDANCE APPLICATION USING MIT APP INVENTOR" as an example of an web to make it easier for students to take attendance without having to take attendance manually. But, there are many previous studies that have developed student attendance applications, including Muhammad Al Satrio with the title Presence Application Using Android-Based QR Code at Bina Darma University. This research discusses the development of an android-based application using a QR Code so that it can make it easier for students to take attendance.

LITERATURE REVIEW

In writing on the topic of this designing a student attendance application, the researcher uses various kinds of literature review references related to the topic. However, the extent to which this research develops requires a review of studies of previous studies that raise the topic of decision support systems. Conduct a study review to learn about the methods[2]–[5], data, and models that have been used in previous research. Based on the review of journals, it can be concluded that the researcher chose Luther to get a decision that was taken effectively and systematically. Starting from collecting data to decide on a value of criteria and alternatives which finally get the results of an effective decision and serve as a guide for decision making through a calculation of a method that has been chosen and the researcher creates or develops the method to a place of research that has not previously been used. Exists or already exists, but is developed from the beginning.

MATERIALS AND METHODS
Understanding MIT App Inventor

MIT App Inventor is a drag-and-drop visual programming tool for designing and building fully functional mobile apps for Android. App Inventor promotes a new era of personal mobile computing in which people are empowered to design, create, and use personally meaningful mobile technology solutions for their daily lives, in endlessly unique situations. App Inventor's intuitive programming metaphor and incremental development

capabilities allow the developer to focus on the logic for programming an app rather than the syntax of the coding language, fostering digital literacy for all. Since it was moved from Google to MIT, a number of improvements have been added, and research projects are underway[6].

App Inventor allows new users to program computers to create software applications for the Android operating system. App Inventor uses a graphical interface, similar to the user interface on Scratch and StarLogo TNG, that allows users to drag-and-drop visual objects to create applications that can run on Android devices. In creating App Inventor, Google has conducted research related to educational computing and completed Google's online development environment[7]–[9].

Data Collection Methods

Data collection is one of the most important stages in research. The correct data collection technique will produce data that has high credibility, and vice versa[10]:

Primary Data Obtained Through :

1.　Literature review is a piece of academic writing demonstrating knowledge and understanding of the academic literature on a specific topic placed in context. A literature review also includes a critical evaluation of the material.

2. Observation is one of the data collection techniques that not only measures the attitudes of respondents (interviews and questionnaires) but can also be used to record various phenomena that occur (situations, conditions). This technique is used when research is aimed at studying human behavior, work processes, symptoms of natural symptoms and is carried out on respondents who are not too large[11].

Research Method

According to Luther, the development of multimedia systems is based on six stages, namely concept, design, material collecting, assembly, testing, and distribution [12]The stages of the multimedia development methodology do not need to be sequential. The six stages can exchange positions but still start from the concept stage first and end with the distribution stage. Of Luther's six stages, starting from Concept and ending with the Distribution stage. While the Material Collecting stage can be done in parallel with the Assembly stage[13]. The stages of Luther's version are described in Figure 1 below:
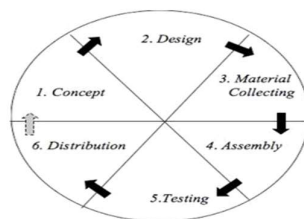
Figure 1. Luther's Development Model

Some stages of development of Luther's model include:

A. The concept stage is the stage to determine the purpose and who is the user of the program. It also determines the type of application (presentation, interactive, etc.) and the purpose of the application (entertainment, training, learning, etc.). The basic rules for design are also determined at this stage, such as application size, target, and others.

B. Design is the stage of making specifications regarding program architecture, style, interface and material requirements for the program. Specifications are made as detailed as possible so that in the next stage, namely material collecting and assembly, new decisions are no longer needed. This stage usually uses storyboards to describe each scene, listing all multimedia objects and links to other scenes. A flowchart to describe the flow from one scene to another. This stage suggests working on specifications in as much detail as possible because it will affect the next stage.

C. Material Collecting is the stage where the collection of materials in accordance with the needs of the multimedia product being worked on such as images, text, and audio. This stage can be done in parallel with the assembly stage. In some cases, the Material Collecting stage and the Assembly stage will be done linearly not in parallel.

D. Assembly is the stage where all multimedia objects or materials are created. Application development is based on the design stage, such as story boards and navigation structures.

E. Testing is done after completing the assembly stage by running the application and seeing if there are any errors or not. This stage is also referred to as the alpha test stage where testing is carried out by the maker, after which a betha test is carried out involving end users. The function of this stage is to see the results of making the application whether it is as expected or not.

F. The stage where the application is stored in a storage medium to be distributed to end users or clients. At this stage if the storage media is not enough to accommodate the application, then compression of the application is done. At this stage an evaluation will also be carried out as input.

RESULTS AND DISCUSSION

Use Case Diagrams

A usecase Diagram is a behavior diagram in the Unified Modeling Language(UML). The use case diagram describe the functional requirements of the software. Use case diagram can be used to understand how the system should work[14]-[15]. Based on the framework

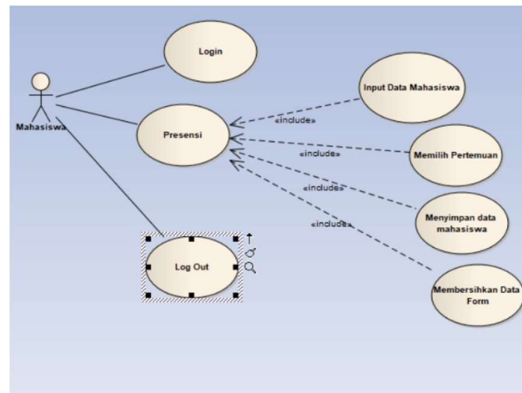above, a use case design is obtained to determine the application to be made as shown in the following figure:

Figure 2. Use Case Diagram

System Display Results

a. Main Page

The following is the main page which consists of an input form for nim, name, class, a combo box to select a meeting, a save button to save data and a reset button to clear the data on the form.

Figure 3. Main Page

CONCLUSIONS

Conclusion

MIT App Inventor is a platform that makes it easy for users to create simple applications without the need to learn many programming languages. By using a wide variety of layouts and components available, we can design Android applications to our liking. In addition, App Inventor also allows users to drag-and-drop visual objects to create applications that can be run on Android devices, without having to have expertise in the field of programming.

REFERENCE

Journal Article

[1]   Purwanto, *Evaluasi hasil belajar*. Pustaka Pelajar, 2009. [Online]. Available: https://books.google.co.id/books?id=C6i_ZwEACAAJ

[2]   J. Parhusip, D. Ronaldo, and B. Valentino, "Rancang Bangun Aplikasi Ensiklopedia Elektronik Tanaman Obat Berbasis Android Menggunakan Metode Luther," *Jurnal Teknologi Informasi: Jurnal Keilmuan dan Aplikasi Bidang Teknik Informatika*, vol. 14, no. 2, pp. 129–141, 2020.

[3]   A. Maulana, V. Rosalina, and E. Safaah, "Implementasi teknologi virtual tour perpustakaan menggunakan metode pengembangan multimedia development life cycle (Mdlc)," *JSiI (Jurnal Sist. Informasi)*, vol. 7, no. 1, pp. 1–6, 2020.

[4]   A. P. D. Prayogha and M. R. Pratama, "Implementasi Metode Luther Untuk Pengembangan Media Pengenalan Tata Surya Berbasis Virtual Reality," *BIOS: Jurnal Teknologi Informasi dan Rekayasa Komputer*, vol. 1, no. 1, pp. 1–14, 2020.

[5]   N. N. K. Sari, "Rancang Bangun Media Pengenalan Huruf Hijaiyah Untuk Anak Usia Dini Berbasis Android," *Jurnal Teknologi Informasi: Jurnal Keilmuan Dan Aplikasi Bidang Teknik Informatika*, vol. 14, no. 2, pp. 161–170, 2020.

[6]   S. C. Pokress and J. J. D. Veiga, "MIT App Inventor: Enabling personal mobile computing," *arXiv preprint arXiv:1310.2830*, 2013.

[7]   D. Wolber, H. Abelson, and M. Friedman, "Democratizing computing with app inventor," *GetMobile: Mobile Computing and Communications*, vol. 18, no. 4, pp. 53–58, 2015.

[8]   S. B. Mir and G. F. Llueca, "Introduction to programming using mobile phones and MIT app inventor," *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, vol. 15, no. 3, pp. 192–201, 2020.

[9]   K. Perdikuri, "Students' Experiences from the use of MIT App Inventor in classroom," in *Proceedings of the 18th Panhellenic conference on informatics*, 2014, pp. 1–6.

[10]   M. Rahardjo, "Metode pengumpulan data penelitian kualitatif," 2011.

[11]   M. Ciesielska, K. W. Boström, and M. Öhlander, "Observation methods," *Qualitative Methodologies in Organization Studies: Volume II: Methods and Possibilities*, pp. 33–52, 2018.

[12]   P. A. Nanda, "Simulasi Visualisasi Teknik Gerakan Yoga Dengan Metode Pengembangan Multimedia Luther-Sutopo Berbasis Mobile," *JURIKOM (Jurnal Riset Komputer)*, vol. 7, no. 2, p. 207, Apr. 2020, doi: 10.30865/jurikom.v7i2.1944.

[13]   A. P. D. Prayogha and M. R. Pratama, "Implementasi Metode Luther Untuk Pengembangan Media Pengenalan Tata Surya Berbasis Virtual Reality," *BIOS: Jurnal Teknologi Informasi dan Rekayasa Komputer*, vol. 1, no. 1, pp. 1–14, 2020.

[14]   R. Fauzan, D. Siahaan, S. Rochimah, and E. Triandini, "Use case diagram similarity measurement: A new approach," in *2019 12th International Conference on Information & Communication Technology and System (ICTS)*, IEEE, 2019, pp. 3–7.

[15]   A. Y. Aleryani, "Comparative study between data flow diagram and use case diagram," *International Journal of Scientific and Research Publications*, vol. 6, no. 3, pp. 124–126, 2016