



FIGHTER PLANE ONLINE GAME BASED ON DESIGN SCIENCE RESEARCH METHODOLOGY USING SOCKET.IO AND NODEJS

I PUTU AGUS EKA PRATAMA*, I WAYAN GUS ARISNA

Dept. of Information Technology, Faculty of Engineering, Udayana University, Bukit Jimbaran, Badung, Bali, Indonesia

**Corresponding author: eka.pratama@unud.ac.id*

(Received: 19 January 2022; Accepted: 27 February 2022; Published online: 01 July 2022)

ABSTRACT: Computer networks are growing rapidly from time to time, making the internet, websites, web programming languages, and socket programming increasingly used in the provision of information technology-based applications and services. One of them is the availability of web-based online game multiplayer applications involving several players. This paper presents the research results in the form of designing and implementing an online game based on socket programming fighter aircraft on a computer network with a client-server paradigm. This research uses DSRM (Design Science Research Methodology) with seven sequential steps. Game applications are designed using Use Case Diagrams and implemented using JavaScript, Socket.io, and NodeJS. Application testing uses Black Box Testing and User Acceptance Testing (UAT). The results of Black Box Testing show that all features and menus in the game application can run well. In contrast, the results of the UAT with a total of 100 respondents indicate that 60% of respondents said the game was very easy to use, 30% said it was easy to use, and 10% said it was difficult to use.

KEYWORDS: *black-box testing, DSRM, fighter plane online game, NodeJS, UAT.*

1. INTRODUCTION

The development of computer networks is rapidly increasing, driving the emergence of various web-based online applications and services on the internet today. One of the online applications and services is online games. The existence of web technology, especially from web 2.0 technology to web 3.0, enables web-based applications and online services to be created more interactively and enables the web to become a communication medium that can collect data and information from anywhere and provide interaction between users. Web-based online games are a form of Web 2.0 and web 3.0 based applications and services attracting the attention of many internet users today. Online games are said to be interesting if these online games can be played by many users at once (multiplayer). To support multiplayer in a computer network, theoretically, an online game application must implement socket and socket programming.

The socket is a two-way communication channel between two or more computers in a computer network, based on the Internet Protocol (IP) at the Network Layer and client-server paradigm [1]. The client-server paradigm is a paradigm in computer networks; at least one computer or device is connected as a service provider (server) and at least one computer or connected device that acts as a service user (client). Each socket is generally equipped with port numbers. Port number is an integer number used to distinguish services running on the

same computer. The port itself can be likened to a track on a network. The socket itself consists of the local and remote port, local and remote Internet Protocol (IP), and network protocol.

With a socket, an application on a computer can open a connection to contact other applications on other computers connected in a computer network [2]. Applications and computers connected can accept connections through the socket. So, in this case, the socket acts as a bridge between computer network-based applications utilizing the TCP/UDP protocol at the Transport Layer and the Internet Protocol at the Network Layer. The TCP/UDP layer in the operating system allows application developers to develop socket programming based applications. Socket programming is one type of programming on a computer network that utilizes the presence and role of the socket and TCP/UDP protocols for the communication paths of at least two different applications or computers that are connected [3]. With TCP and UDP protocols, two types of sockets can be selected for use in Socket Programming, namely: 1.) Connection-oriented and reliable TCP Socket where data is sent in data packets with sequential numbers, 2.) UDP Socket is connectionless and unreliable, where data is sent in datagrams. In computer networks, applications are developed based on Socket Programming, one of which facilitates the transfer of files between systems [4]. Thus, this paper aims to designing and implementing an online game based on socket programming fighter aircraft on a computer network with a client-server paradigm. The illustration of the socket at client-server paradigm on the computer network is shown in Fig. 1.

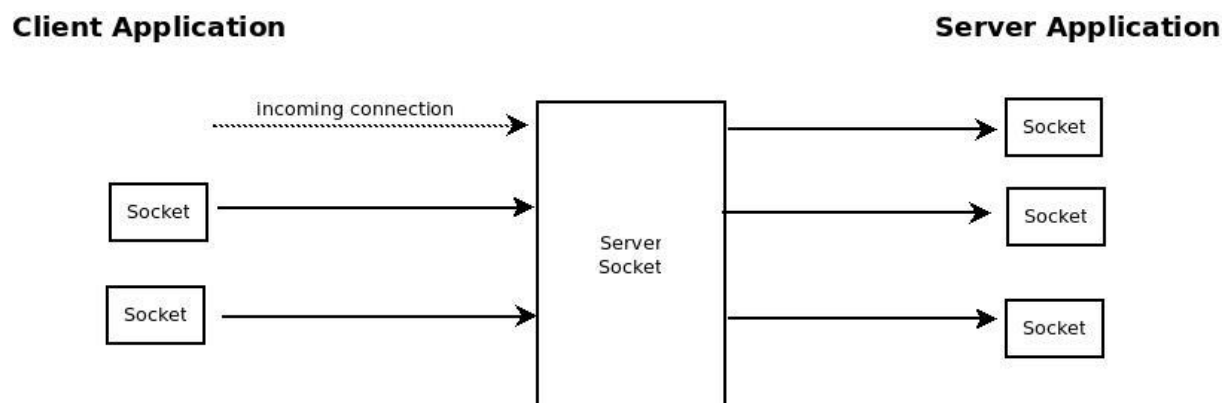


Fig. 1. Illustration of the socket at client-server paradigm on the computer network

2. MATERIALS AND METHOD

2.1. Hardware and Software

The hardware used in this research is Toshiba Intel i3 @ 1,8GHz, 4GB of RAM). The software used is Ubuntu Linux, Apache, PHP, MySQL, NodeJS, Socket.io, HTML5, V8 JavaScript engine, Firefox browser, and Chrome browser.

2.2. Design Science Research Methodology (DSRM)

The research method used in this research is Design Science Research Methodology (DSRM) which includes 7 sequential steps, namely: the study of literature, determination of research objectives, identification of problems, design solutions, making prototype applications (prototyping), demo, documentation and publications [5]. DSRM is a research methodology that is widely used in research related to software development and applied applications, such

as developing village fund applications [6] and developing data collection systems for migrants based on Hybrid Cloud [7]. The seven sequential steps at Design Science Research Methodology (DSRM) are shown in Fig. 2:

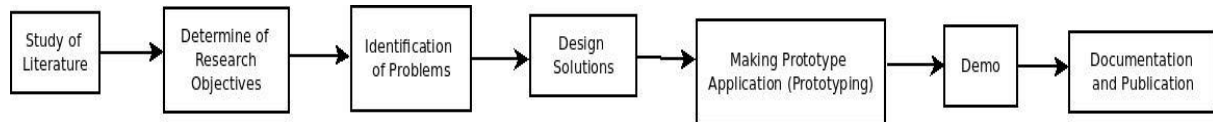


Fig. 2. Seven research steps at DSRM

For the literature study stage, several references related to the field of research were collected and several state-of-the-art pieces from previous related studies. These references are taken on the internet in journals, proceedings, websites, and books/e-books. In determining research objectives, research objectives are determined to design and implement socket programming in the form of web-based fighter plane online game applications, to be able to prove the value of socket and socket programming in two-way communication (client-server) on a computer network in real-time. Problem identification is stated in the form of research questions, which include: 1.) Using the socket programming concept, how to design and implement fighter plane online games based on Design Science research Methodology (DSRM)? 2.) How to test the fighter plane online games that have been designed and implemented?

Solution design is carried out in application design using Use Case Diagrams that consist of an actor and three use cases. The prototyping process of web-based fighter plane online game applications is based on the design, using the JavaScript programming language, Socket.io library, and NodeJS, which act as web servers. Demo of the completed application prototype, done on the developer and user side, accompanied by Black Box Testing and User Acceptance Testing (UAT) methods. Documentation includes recording steps and test results and analyzing and discussing test results, concluding, and writing in the form of published papers. The publication is done through selected scientific journals (in this research paper published in the Kinetic Journal).

The development of technology in the web field also influenced the programming languages and libraries used in developing applications in this study. Three technologies are selected on the latest interrelated website platforms to support the current web 3.0 technology along with the socket programming: JavaScript as the main programming language in application development; and Socket.io as a library to support real-time communication; and NodeJS as a web server [8]–[10]. JavaScript has been widely recognized as a powerful programming language for web-based application development in the web 2.0 and 3.0 eras in the form of game applications, 2D games, animations, and simulations [11], [12]. NodeJS combined with HTML5, is widely used in developing web 2.0 and web 3.0 platform applications [13].

2.2. State of the Art

Several previous types of research underlie this research underlie the development of several game applications and their impacts and studies. DeRosier and Thomas developed the Hall of Heroes as an online game application aimed at young users to improve their social skills, utilizing the Social Skills Training (SST) method [14]. Wu and Wang present research in the field of game application development by focusing on the study of several scientific kinds of literature relating to the method of based learning using the Game Development



Framework (GDF) on children's education, so that the level of effectiveness of education can be known, to increase knowledge, skills, attitudes, and behaviour of children [15]. Almeida et al. Developed ALTRIRAS, a game application intended for autistic children to learn the emotional basics in feeling sad, happy, or angry [16]. Alom et al. [17] conveyed the design and implementation of collaborative multiplayer games using the latest and dynamic technologies such as Construct2 and HTML5, which can accelerate data flow and enhance user experience. Aleem et al. [18] used the Systematic Literature Review (SLR) methodology to assess mobile applications research topics. Still, in the realm of research in the development of game applications to help the learning process, Oberdorfer and Latoschik [19] conveyed that Desktop Mechanics Game based on Desktop 3D and Virtual Reality (VR) to improve the effectiveness and efficiency of learning outcomes achieved and the effectiveness of audio-visual knowledge in the learning process.

In addition, there are also several previous types of research on socket programming based multiplayer games in computer networks along with the technology used. Amir et al. [20] announced his research in architecture and the environment development of multiplayer online games with the concept of socket programming in a computer network, using a peer-to-peer paradigm. Widharma [21] developed socket programming and artificial intelligence-based multiplayer game applications using Non-Player Character (NPC). Ryan et al. [22] developed an Android-based multiplayer online RPG game application on smartphone devices for children, given the fact that smartphones are currently widely used by children to get entertainment through various games provided in them. Riwinoto and Muspita [23] conducted research in a comparison test between single-player game applications and multiplayer game applications through questionnaire methods to users, where both types of applications were developed based on socket programming and Mobile Backend a Service (MBaaS) Cloud.

Meanwhile, Lai and Beaumont researched developing client server-based online game applications and socket programming on Ubiquitous Computing [24]. Husniah et al. conducted a multiplayer application development research using the MDA Framework to introduce Indonesian traditional weapons to the younger generation [25]. So, from these previous researches, a research gap was obtained, namely the absence of research in the form of developing a web-based fighter plane online game application based on DSRM with the concept of socket programming, which was implemented using JavaScript, Socket.io, and NodeJS. This research gap is then chosen to be worked on in this research. The novelty is obtained from the technical side of implementation and the answers to the research questions raised at problem identification.

3. RESULTS AND DISCUSSION

3.1. Application Design (Use Case Diagram)

The first thing to do is create a design from the application using the Use Case Diagram that contains an actor and three use cases. This web-based fighter plane online game application allows players to enter a room where other players can play together (multiple players). The three features provided application: players can move and control their fighter plane objects, shoot bullets at other players' fighter objects (as an enemy), and see the player's score that will increase if the bullet hits the enemy plane, as shown in Fig. 3:

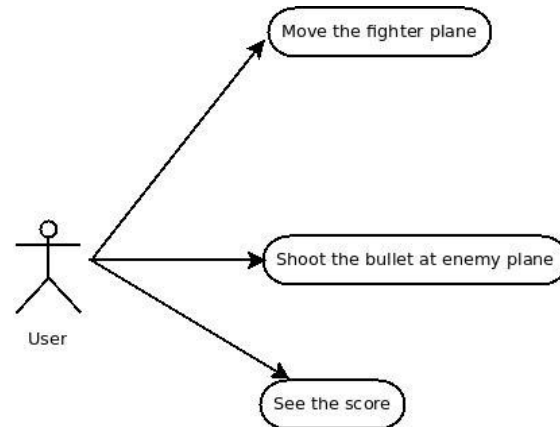


Fig. 3. Use Case Diagram

3.2. System Architecture Design

In developing the online game application fighter plane based on the web, system architecture design is also carried out. The system architecture design developed refers to the client-server paradigms in a computer network. Therefore, there are two main components in the architecture of online game-based fighter plane applications on the web, namely the application running on the server and the application running on the client. The system architecture design is shown in Fig. 4:

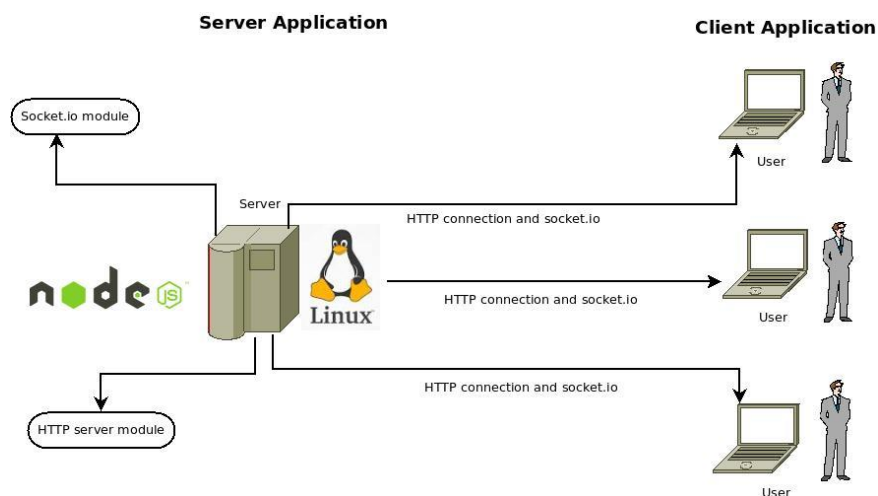


Fig. 4. System architecture design

From Fig. 4. above, the application that running on the client, using a web browser that supports HTML5 (in this case HTML5 is an application canvas), while on the server-side application, there is a Linux operating system and NodeJS, which has two modules, namely HTTP Server and Socket.IO. NodeJS is tasked with providing services for all users (clients) and serving all their requests through the two modules available. The HTTP Server module is tasked with handling the process of retrieving HTML pages in a web browser. In contrast, the Socket.IO module is tasked with handling Socket.IO connections over a computer network with the concept of socket programming and the client-server paradigm.



3.3. Implementation

After the application design and system architecture design have been carried out, proceed with the implementation process. Both HTML5 (as a canvas) and JavaScript were used as a client application and NodeJS as a server application. The application structure consists of sub-directories of server, client, node modules, and the app.js and package.json files. All subdirectories and files are placed in the game-multiplayer directory. The server must be activated with the node app.js command at Terminal to run the application first. Fig. 5 shows when running the server application:

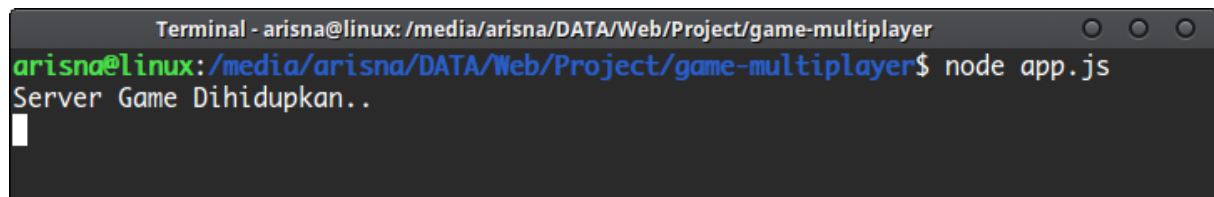


Fig. 5. Running the server application

After the server is activated, all players can use the game. Players can access the address 127.0.0.1:2000 on a web browser. In this research, two different web browsers are used, namely Google Chrome and Mozilla Firefox, representing two different players with different IDs. Each web browser that accesses the server address will be recorded by the server with a different ID and a fighting plane display. The ID for each player is random, which is stored and recognized by the server. Fig. 6 shows the player accessing the server address from two different web browsers (ID 975 and ID 802) and when the player moves the fighter plane and shoots the enemy.



Fig. 6. The player accesses the server address (ID 975 and ID 802) and shoots the enemy



From Fig. 6 above, when all players are logged into the system, they can play multiplayer games in real-time. Players can move the fighter plane (left, right) and shoot enemies, and other players can return fire. If the bullet hits another player's fighter plane, the player's score will increase, and the server will record it. Fig. 6 shows the player accessing the server address from two different web browsers (ID 975 and ID 802) and when the player moves the fighter plane and shoots the enemy.

3.4. Black Box Testing

Black Box Testing is used to test on the developer side of all the features and menus available in the Fighter Plane Online Game Based on Design Science Research Methodology Using Socket.io and NodeJS. There are six test steps, the sequence of steps carried out, and the results. Table 1 shows the testing, testing steps, and results of Black Box Testing:

Table 1: Black Box Testing, Testing Step, and Result

Black Box Testing	Testing Step	Result
Activating the server	Open Terminal → type the command node app.js	Success
Login as player	Access the server address at URL 127.0.0.1:2000 → see the display of player ID → see the fighter plane used → see the other players	Success
The server assigns random IDs to the players and records each player's ID	The server generates an ID for each player --> server display ID to the player on-screen --> server also display join status for each player	Success
Players can move the fighter plane	Players move the fighter plane as needed to the left and right to avoid the enemy or get close to the enemy → player use the left and right button → fighter plane can move left and right according to instructions	Success
Players can shoot enemies and get scores if the shot hits the enemy	The player shoots the enemy on the target --> the player can see the increase in score points when the enemy is shot	Success
The server can record the accumulated points of each player and display them to the players	The server records the movement of each player --> the server accumulates points for each successful shot --> the server displays the score to the player	Success

3.5. User Acceptance Testing (UAT)

In this research, User Acceptance Testing (UAT) was conducted through a questionnaire to 100 respondents with questions about the ease of implementing the Online Game of Fighter Aircraft Using Socket.io and NodeJS-Based Socket Programming Concepts. Three answer

choices were given to respondents, namely very easy, easy, and difficult. Fig. 7 shows a pie chart of the UAT results:

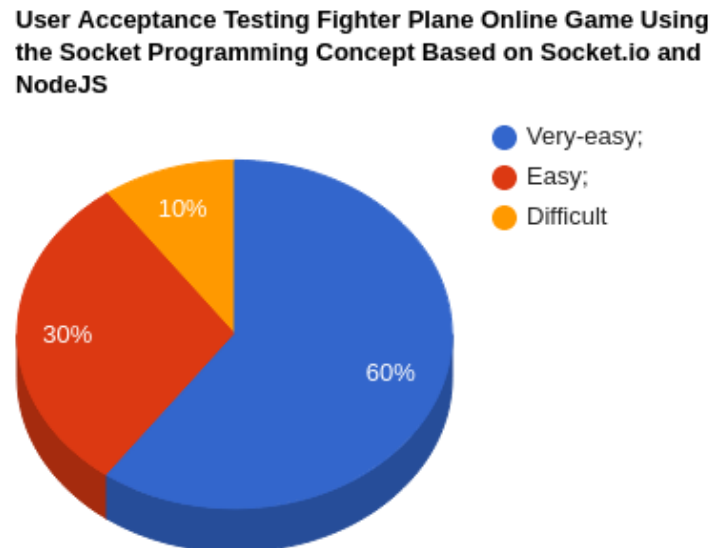


Fig. 7. Statement of response from 60 users

From Fig. 7. above, the test results show that most users (60 respondents or 60%) find it very easy to use this game application, the remaining 30 respondents (30%) find it easy to use, and the rest 10 respondents (10%) find it difficult to use this application.

4. CONCLUSION

In this research, testing the Fighter Plane Online Game Based on Design Science Research Methodology Using Socket.io and NodeJS was carried out on the developer side using Black Box Testing and, on the user, side using User Acceptance Testing (UAT). Based on the Black Box Testing carried out, it can be concluded that technically all the features and menus in the application can run well following the design and development objectives. Based on User Acceptance Testing (UAT) conducted on 100 respondents online, 60% of respondents stated that the application was very easy to understand and use, 30% said it was easy to use. The remaining 10% found it difficult to use it. So it can be concluded that in general, application development has been successfully carried out where all features and menus run well, and the majority of users are able to use it very easily.

ACKNOWLEDGEMENT

Thanks to the Indonesian Linux community, the Indonesian Open Source community, the Indonesian NodeJS community, and Udayana University during this research.

REFERENCES

- [1] R. L. R. Maata, R. Cordova, B. Sudramurthy, and A. Halibas, "Design and implementation of client-server based application using socket programming in a distributed computing environment," in *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 2017, pp. 1–4.
- [2] I. P. A. E. Pratama, *Handbook Jaringan Komputer Teori dan Praktik Berbasikan Open Source*. Bandung: Penerbit Informatika, 2014.



- [3] D. Darmawan and P. Pharmayeni, "Development of client-server application by using udp socket programming for remotely monitoring CNC machine environment in fixture process," *J. Rekayasa Elektr.*, vol. 12, no. 2, pp. 48–53, 2016.
- [4] P. Arora and A. Dumka, "Review of Sockets for Transfer of Files Between Systems," in *Intelligent Communication, Control and Devices*, Springer, 2018, pp. 1455–1465.
- [5] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *J. Manag. Inf. Syst.*, vol. 24, no. 3, pp. 45–77, 2007.
- [6] N. Dilson and N. Noviard, "Metode DSRM Dalam Pengembangan Aplikasi Penggunaan Alokasi Dana Desa," *J. RESTI*, vol. 1, no. 3, pp. 217–225, 2017.
- [7] I. Pratama, "Uat sistem pendataan penduduk pendatang di kabupaten ganyar berbasis hybrid cloud," 2018.
- [8] R. Nixon, *Learning PHP, MySQL & JavaScript*. "O'Reilly Media, Inc.," 2021.
- [9] R. Rai, *Socket. IO Real-time Web Application Development*. Packt Publishing Ltd, 2013.
- [10] S. Tilkov and S. Vinoski, "Node. js: Using JavaScript to build high-performance network programs," *IEEE Internet Comput.*, vol. 14, no. 6, pp. 80–83, 2010.
- [11] A. Dobre and D. Ramtal, *Physics for JavaScript Games, Animation, and Simulations: With HTML5 Canvas*. Apress, 2014.
- [12] G. Stuart, G. Stuart, and Corrigan, *Introducing JavaScript Game Development*. Springer, 2017.
- [13] D. Nagle, *HTML5 Game Engines: App Development and Distribution*. CRC Press, 2014.
- [14] M. E. DeRosier and J. M. Thomas, "Hall of heroes: A digital game for social skills training with young adolescents," *Int. J. Comput. Games Technol.*, vol. 2019, 2019.
- [15] B. Wu and A. I. Wang, "A guideline for game development-based learning: a literature review," *Int. J. Comput. Games Technol.*, vol. 2012, 2012.
- [16] L. M. Almeida *et al.*, "ALTRIRAS: A computer game for training children with autism spectrum disorder in the recognition of basic emotions," *Int. J. Comput. Games Technol.*, vol. 2019, 2019.
- [17] B. M. M. Alom, C. Scoular, and N. Awwal, "Multiplayer game design: performance enhancement with employment of novel technology," *Int. J. Comput. Appl.*, vol. 145, no. 1, pp. 27–32, 2016.
- [18] S. Aleem, L. F. Capretz, and F. Ahmed, "Game development software engineering process life cycle: a systematic review," *J. Softw. Eng. Res. Dev.*, vol. 4, no. 1, pp. 1–30, 2016.
- [19] S. Oberdörfer and M. E. Latoschik, "Knowledge encoding in game mechanics: Transfer-oriented knowledge learning in desktop-3d and vr," *Int. J. Comput. Games Technol.*, vol. 2019, 2019.
- [20] A. Yahyavi and B. Kemme, "Peer-to-peer architectures for massively multiplayer online games: A survey," *ACM Comput. Surv.*, vol. 46, no. 1, pp. 1–51, 2013.
- [21] I. B. M. O. Widharma, "Game FPS dengan menggunakan multiplayer game," *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 1, no. 01, 2016.
- [22] R. Hosea, G. S. Budhi, and L. W. Santoso, "Pembuatan game rpg multiplayer online berbasis android," *J. Infra*, vol. 5, no. 1, pp. 355–361, 2017.
- [23] R. Riwinoto and A. Muspita, "Penerapan Multiplayer Pada Aplikasi Permainan Android (Studi Kasus Aplikasi Permainan" Bisa Jadi")," *Pros. Semnastek*, 2017.
- [24] A. S. Y. Lai and A. J. Beaumont, "Mobile bluetooth-based multi-player game development in ubiquitous computing," *J. Comput. Inf. Syst.*, vol. 6, no. 14, pp. 4617–4625, 2010.
- [25] L. Husniah, F. Fannani, A. S. Kholimi, and A. E. Kristanto, "Game Development to Introduce Indonesian Traditional Weapons using MDA Framework," *KINETIK*, vol. 4, no. 1, pp. 27–36, 2019.