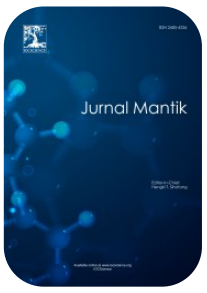

JURNAL MANTIK

INSTITUTE OF COMPUTER SCIENCE

P-ISSN : E-ISSN : 26854236 Subject Area : Economy, Science, Education

1.10495
Impact Factor

1411
Google Citations

Sinta 4
Current Acreditation

Google Scholar Garuda Website Editor URL

History Accreditation

2018 2019 2020 2021 2022 2023 2024

Garuda Google Scholar

The Effectiveness of Giving Red Spinning Juice on Increasing Hb Levels for Pregnant Anemia in the Work East Lahewa Health Center Year 2020

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2145-2150

2022 DOI: - Accred : Sinta 4

Optimization of Virtual Reality-Based Flood Simulation by Manipulating Particle System

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2117-2126

2022 DOI: - Accred : Sinta 4

Implementation of Elo Rating System and Player Clustering for Competitive Matchmaking in Trivia Education Game

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik)

2022 DOI: - Accred : Sinta 4

Database Forensics in Software as A Service Service using Stored Procedure

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2127-2135

2022 DOI: - Accred : Sinta 4

Automation Simulation of Plant Watering System using Moisture Sensor Based on Mobile Devices Urban Farming Method

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2136-2144

2022 DOI: - Accred : Sinta 4

Disease in Corn Leafe Using Gabor Wavelet and K-Means Clustering Algorithm

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2152-2156

2022 DOI: - Accred : Sinta 4

Optimization of Tree Algorithms by Resampling and Ensembling in Deffect Prediction Software:

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2177-2181

2022 DOI: - Accred : Sinta 4

Analysis of Service Quality and Product Quality on Customer Purchase Decisions in CV. Surya Indah Jaya Medan with Price as a Variable Intervention

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2156-2164

2022 DOI: - Accred : Sinta 4

Analysis of Factors Affecting the Work Discipline of Employees CV. Golden Seafresh Medan

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2165-2170

2022 DOI: - Accred : Sinta 4

The Effect of Pricing and Product Quality on Purchase Decisions on PT. Gunung Sentosa Sumatera Utama Industri

Institute of Computer Science (IOCS) Jurnal Mantik Vol. 5 No. 4 (2022): February: Manajemen, Teknologi Informatika dan Komunikasi (Mantik) 2171-2176

2022 DOI: - Accred : Sinta 4

View more ...

Jurnal Mantik
Volume 4, Number 1, Augustus 2020, pp. 1185-1191 E-ISSN 2685-4236
https://iocscience.org/ejournal/index.php/mantik

Crowd Modelling and Navigation in Unity 3D Game Engine

Moch Fachri[1], Avip Kurniawan[2], Ali Khumaidi[3]

1,2,3,4Informatics,Engineering Faculty,Universitas Krisnadwipayana, Jl. Raya Jatiwaringin, Jawa Barat 13077, Indonesia

E-mail: [1]moch.fachri@unkris.ac.id, [2]avip.kurniawan@unkris.ac.id, [3]alikhumaidi@unkris.ac.id

ARTICLE INFO

Article history:
Received: 12/01/2020
Revised: 22/09/2020
Accepted: 01/08/2020

Keywords:
Crowd Modelling, Unity3D, Prototype Method.

ABSTRACT

In the curret era of building construction and development, it is very difficult to foresee the evacuation point of a building if disaster occure, especially for huge and tall buildings. For that purpose we make crowd modeling on the Unity3D game engine. We try to build a concept using evacuation scenarios in the Krisnadwipayana University, Faculty of Engineering building. By utilizing crowd modeling using Unity3D game engine, we will put all crowd agents together at the virtual environment of the target building at maximum density. In this study we using the prototype method to build the virtual area/building, and mapping the movement of the crowds in order to calculate the movement to be carried out in a form of visual crowds movement. The results of research conducted show that the final prototype that was made can accept 3D building models using Sketchup. From the data that has been made through the prototype there are 4498 agents who exit through the main door and 2068 agents use the back door of the target building, with the time needed to evacuate for 2 minutes and 16 seconds. The conclusion of this research is that it is able to present a crowd modeling form with integrated navigation on the game engine, and by using Unity3D for crowd modeling, it can help various groups that require testing in an area.

1. Introduction

Crowd modeling is modeling the movement of a group of entities that move in an area [11]. With crowd modeling, crowd conditions in places such as buildings, squares, stadiums, for example, can be modeled in virtual form. This virtual modeling can be done using a game engine. The game engine or game engine itself is a software development device that is designed in making games. The game engine includes graphics engine, animation, physics engine, sound engine, and scripting [6]. One example of such game engine is Unity3D.

Unity3D with all the features that have been provided, can be utilized in developing applications for games[1], as well as other needs such as crowd simulation [5][8]. In the process of making applications at Unity3D, a team with various abilities is needed, including project managers, technical directors, art directors and various artists[6]. Unity3D has also been reviewed as having advantages over other game machines. As in[3] where when compared using AHP with Unreal Engine and CryEngine, Unity was ranked first with a score of 0.467. The main advantage of this game engine is that it allows the prototyping process to be fast enough[4]. This is because Unity3D provides an overall solution to software development needs that require utilization of two or three dimensional graphics of various formats, and multiplatform support for the results of applications created[9].

2. Method

This research uses the prototype method[2]. In this method a prototype software system is built, tested and then reworked as needed until results are acceptable. This result is the availability of a system with overall functionality. Our method is as shown in figure 1 which contains several process in order to produce best prototype result.

2.1. Needs analysis

This stage contains the collection of all the requirements needed to engineer the prototype to be made, as mentioned earlier, in the process of modeling this crowd required an area where this crowd will be modeled[7]. In this study we used the Krisnadwipayana University Faculty of Engineering building model. The 3D model

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

of the building is based on the schematic of the building shown in Figure 2.

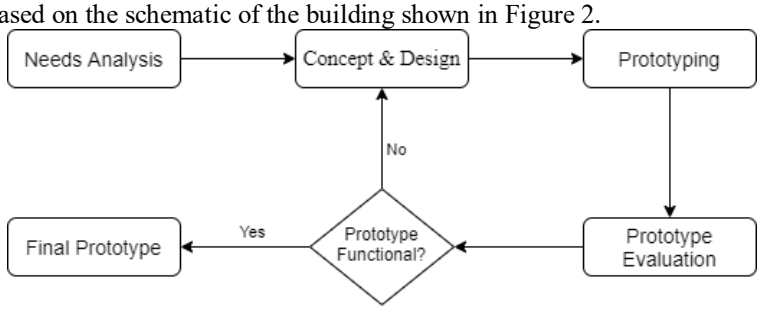

Fig 1. Prototype Method.

Other operational requirements have been provided in Unity3D game engines, such as Graphics and Physics engines[6], and navigation systems for crowd agents[8]. So the only thing made outside of Unity3D is the building model, where the building model is made using a third-party application. Application for building models using SketchUp, where the results of the 3D model of this building can be imported into the Unity3D game engine.

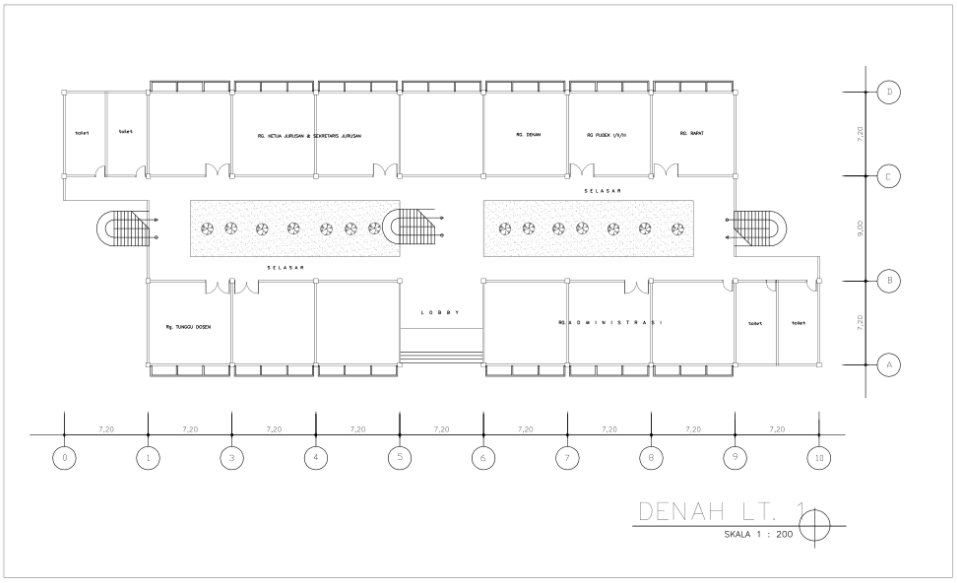

Fig 2. Schematic Trailer of the UNKRIS Faculty of Engineering Building which will be used in crowd modeling.

The navigation system for crowd agents in the Unity3D game engine is divided into two main processes, namely global and local navigation[10]. Figure 3 from the documentation of the Unity3D navigation system [10] illustrates the relationship between global and local navigation. Global navigation aims to determine the path / corridor that needs to be taken by a crowd agent to get to the destination point. Furthermore, on the local navigation agent will be directed to move according to the path that has been calculated from global navigation. Avoidance of obstacles that interfere with the running of crowd agents is also calculated in this local navigation.

2.2. Concept & Design

In this crowd modeling we conceptualize a scenario, namely the evacuation scenario of buildings in the UNKRIS Engineering Faculty building. The whole crowd agent will be placed with maximum density in each room except for the toilet. The crowd will move towards the evacuation point outside the building. When the application starts running, the first one is done is to model the movement of the crowd using the navigation system that is on Unity3D[8]. The results of this crowd movement model are not directly executed in the form of visual movements, but are stored first in memory until the calculation of the movements of all agents is completed.

After calculating the movements of all agents, the results of the crowd modeling can be presented in the form of an overall movement of the crowd agents heading out of the building for evacuation. Simultaneous crowd movement is possible because the calculation of crowd movement in Unity3D allows it to be pooled into memory. The results of this calculation are then executed when all agents have their model of movement for evacuation out of the building.

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

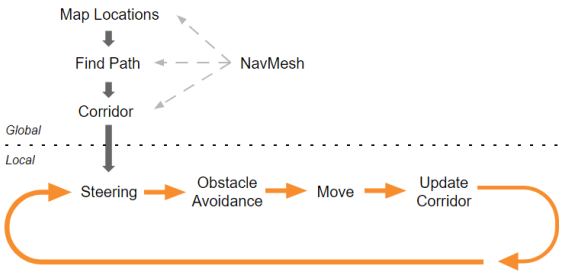

Fig 3.Navigation system of agents movement on Unity3D[10].

2.3. Prototyping

Based on the conceptual process that has been made, the prototype program will be designed with the steps shown in Figure 4. As shown in Figure 4, the building process and the crowd modeling application are made separately because they use different tools, namely building models with SketchUp, and making application with Unity3D. both results will be integrated, where 3D building models will be imported into Unity3D. After the building model is integrated, next stage is the protoype evaluation.

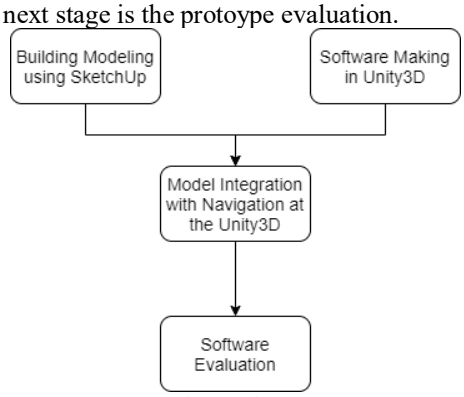

Fig 4. The process of making a prototype software for crowd simulation in Unity3D and SketchUp.

2.4. Prototype Evaluation

In this section, the prototype that has been created will be evaluated according to the functionality requirements of the application. The functionality needed in this case is the ability to map the area / building where the movement of the crowd. This protoype must also be able to calculate the movement to be carried out by a group of crowds to get to the specified point (in this case the evacuation point).

The results of crowd calculation results also need to be visualized in the form of movement of the crowd. In this crowd simulation for this evacuation, also required the existence of exit data used by the crowd agent. The ability to calculate the time needed for the entire crowd to clear the building is also very necessary. Exit usage data and evacuation time needed in evacuation can be used as a reference for the effectiveness of building design in evacuation conditions.

If the prototype produced meets the needs above, then the prototype has been formalized as the final prototype. If there is functionality that has not yet been fulfilled, a revamping process will begin, starting from the concept design to the protoype-making section. This process is repeated until the prototype needs are met.

3. Result and Discussion

The final prototype produced was able to accept 3D building models created with SketchUp. The results of the building model that have been imported in Unity3D are shown in figure 5. The building is a four-story Unkris Engineering Faculty building. The building has two doors for access in and out of the building, the main door (picture 6) and the back door (picture 7). The crowd in the evacuation scenario will try to get out of the Unkris Engineering Faculty building through the two doors.

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

Fig 5. 3D model of the UNKRIS Faculty of Engineering building at Unity3D.

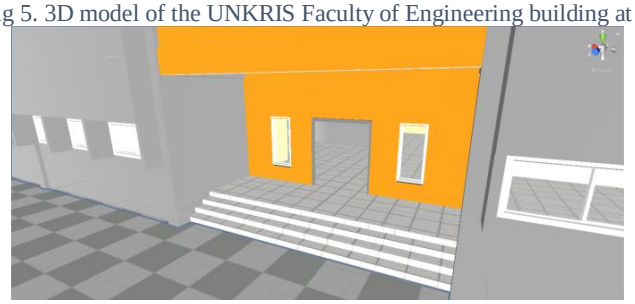

Fig 6. The main door on the 3D model of the UNKRIS Engineering Faculty building.

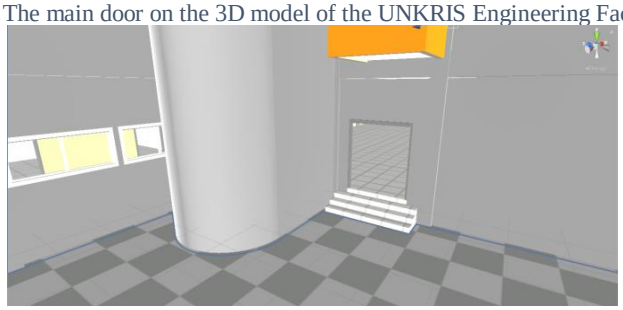

Fig 7. The back door on the 3D model of the Faculty of Engineering building.

When running the application, the user will be presented with a view of the building from above. The display is shown in Figure 8. From the display there is an option to see the condition of each floor in the UNKRIS Engineering Faculty building. The crowd will be raised with the option "start calculation", when the option is selected, the crowd agent will be placed until it fills the entire room in the UNKRIS Engineering Faculty building, except for the toilet room. Placement of crowd agents in the UNKRIS Faculty of Engineering building is 6566 agents. The placement of agents in this building model can be seen in Figure 9. where an agent is represented as a green dot on the building. Each agent is placed with an idle condition as a condition during initialization.

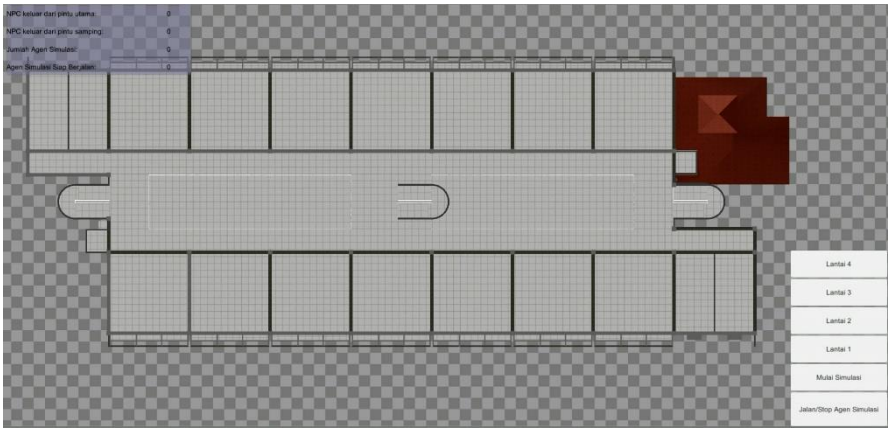

Fig 8. Application appearance when running.

After placing the crowd agent, the next is the calculation of the movement of each crowd agent. The result of this calculation is the model of crowd movement, which in this case is the modeling of crowd motion on evacuation in the UNKRIS Faculty of Engineering building. After the calculation is done, the movement model of the simulation can be displayed. Figure 10 shows a snapshot of the movement of a crowd agent to get out of the building. After all agents have exited the building as shown in Figure 11, data on the use of building

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

doors used during the course of the agent movement can be seen in the application interface. In the evacuation at the UNKRIS Engineering Faculty building, it was found that 4498 agents exited the building using the main door, and the remaining 2068 agents used the back door of the building. The time needed to vacate the building for 2 minutes 16 seconds. The number of agents that evacuated over time is shown in Figure 12.

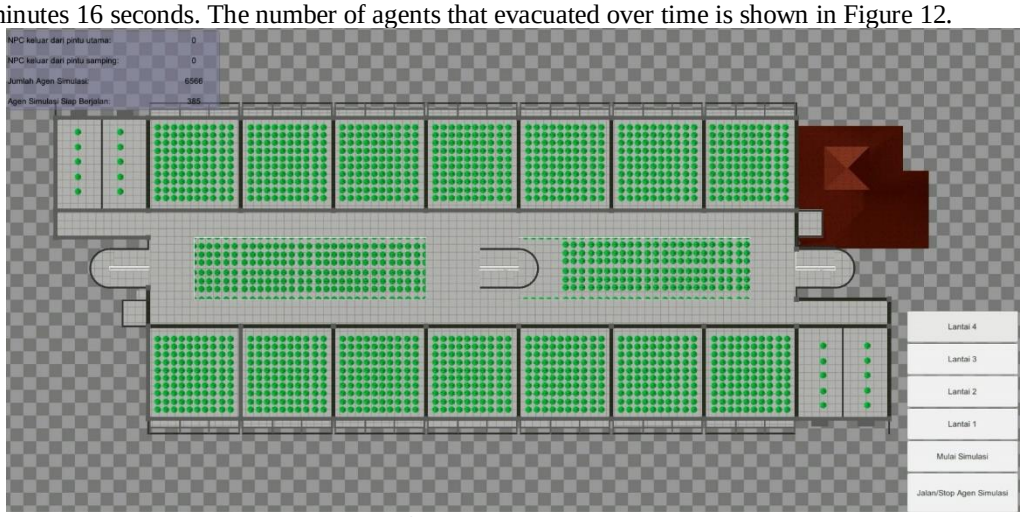

Fig 9. Placement of crowd agents in the calculation process.

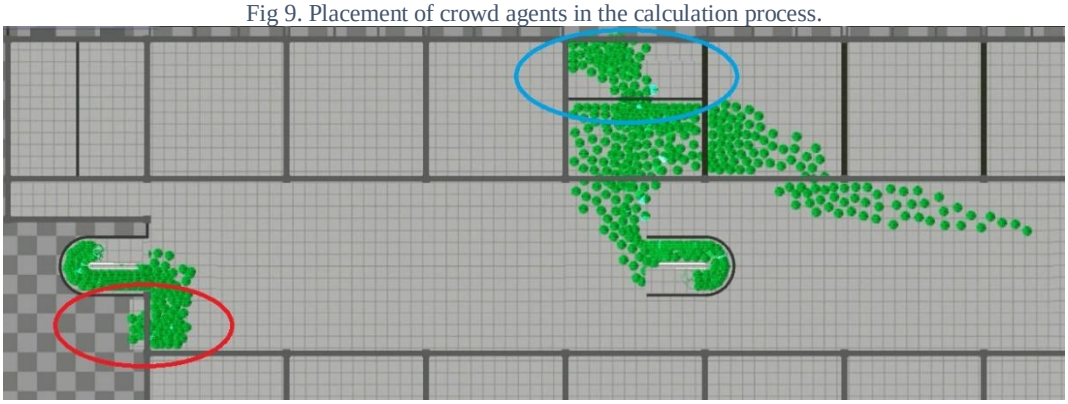

Fig 10. The movement of agents crowd headed out of the building.

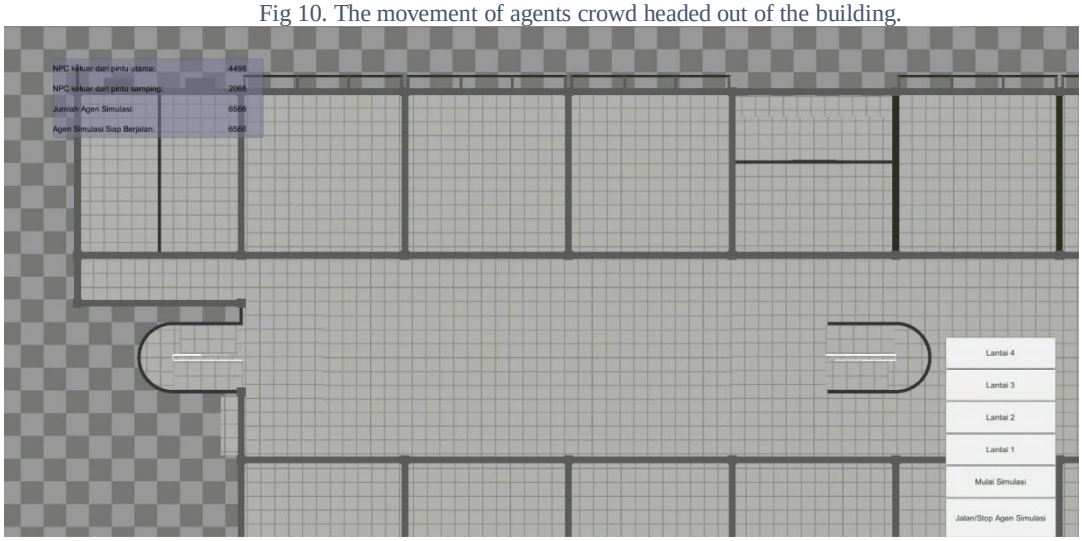

Fig 11. The whole crowd had left the building safely.

4. Conclusion

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

In this study we show the use of Unity3D in crowd modeling in a building. From the crowd modeling process with the prototyping method using Unity3D, it was concluded that the Unity3D game engine is able to present crowd modeling forms with a navigation system that has been integrated on the game engine and also the ability to receive 3D building models from third-party applications (in this case SketchUp). The Unity3D game engine as a game engine fulfills all crowd modeling needs. Successful needs are met which is able to map the area / building where the movement of the crowd. This application can also display crowd movements from the results of modeling calculations for each crowd agent. Last but not least, this application can display data on the use of exits used by crowd agents to reach the evacuation point outside the building, as well as being able to calculate the time needed to vacate the building. This is very helpful for various groups to conduct crowd testing in an area with evacuation scenarios with user-specified evacuation points.

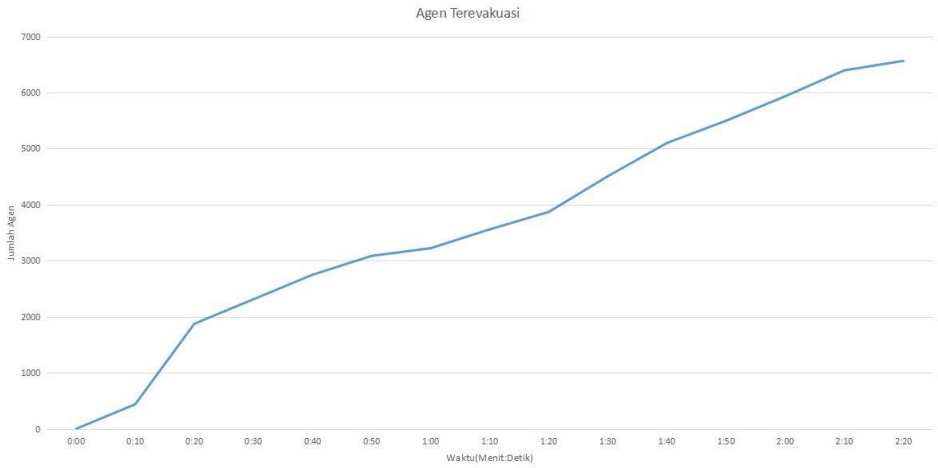

Fig 12. Graph of number of agents successfully evacuated with evacuation time.

5. References

[1] Andre J.L., Eva H., Madenda S., and Oktavia C. A, 2018. Pengembangan Game Virtual Reality Berbasis Android Menggunakan Unity Sebagai Media Penunjang Pengenalan Bahasa Inggris. J-INTECH, Vol 6(2) Desember 2018, pp.208-213.

[2] Banoriya D., Purohit R., Dwivedi R.K. (2015). Modern Trends in Rapid Prototyping for Biomedical Applications. Materials Today: Proceedings, 2(4-5): 3409-3418.

[3] Bourhim, E.M., and Cherkaoui, A., 2018. Selection of Optimal Game Engine by Using AHP Approach for Virtual Reality Fire Safety Training. In: 18th International Conference on Intelligent Systems Design and Applications (ISDA 2018). Vellore, India 6–8 December 2018. Springer: Switzerland.

[4] Brett, J., and Simons, A., 2017. Evaluating the use of the Unity engine for developing 2D mobile games in consideration of start-up/student developers. International Conference on Technologies for E-Learning and Digital Entertainment. pp. 271-278. 10.1007/978-3-319-65849-0_29.

[5] Juniastuti, S., Fachri, M., Nugroho, S.M.S., and Hariadi M., Crowd navigation using leader-follower algorithm based Reciprocal Velocity Obstacles, 2016 International Symposium on Electronics and Smart Devices (ISESD), Bandung, 2016, pp. 148-152. doi: 10.1109/ISESD.2016.7886709

[6] Labschütz, M., and Krösl, K., 2011. Content Creation for a 3D Game with Maya and Unity 3D. In: CESCG 2011: The 15th Central European Seminar on Computer Graphics. Vinič n, Slovakia 2-4 May 2011. Technische Universität Wien: Wien.

[7] Mahindru, D.V. & Mahendru, P. (2013). Review of Rapid Prototyping-Technology for the Future. Global Journal of Computer Science and Technology

[8] Nandy, A., 2018. Crowd Simulation as a Flocking Behavior and Windows. Intel Corporation. Retrieved from https://software.intel.com/en-us/articles/crowd-simulation-as-a-flocking-behavior-and-windows-mixed-reality-part-1 at 3 November 2019

[9] Petridis, P., Dunwell, I., Freitas, S.D., and Panzoli, D., 2010. An Engine Selection Methodology for High Fidelity Serious Games. In: 2nd International Conference on Games and Virtual Worlds for Serious Applications). Braga, Portugal 25-26 March 2010. IEEE Computer Society: Washington, D.C.

[10] Unity Technologies, 2019. Inner Workings of the Navigation System [Online] (Updated 14 January 2020). Tersedia di: https://docs.unity3d.com/Manual/nav-InnerWorkings.html. [Accessed 21 January 2020]

[11] Wang, J., Zhang, L., Shi, Q., Yang, P., dan Hu, X. 2015. Modeling and Simulating for Congestion Pedestrian Evacuation with Panic. Physica A: Statistical Mechanics and Its Applications, 428. Elsevier B.V: 396-409.