

# Pengembangan Lingkungan Simulasi untuk Pengujian *Socially Assistive Robots* Menggunakan ROS 2 dan Gazebo

Muhammad Alfie Maulana Fikri  
Departemen Teknik Komputer  
Fakultas Teknologi Elektro  
dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia 60111  
maulana.17072@mhs.its.ac.id

Mauridhi Hery Purnomo  
Departemen Teknik Komputer  
Fakultas Teknologi Elektro  
dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia 60111  
hery@ee.its.ac.id

I Ketut Eddy Purnama  
Departemen Teknik Komputer  
Fakultas Teknologi Elektro  
dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia 60111  
ketut@te.its.ac.id

**Abstrak**—Pada penelitian ini kami mengajukan lingkungan simulasi yang dikembangkan menggunakan ROS 2 dan Gazebo untuk pengujian *socially assistive robots* (SARs). Di dalam lingkungan simulasi ini, model robot yang digunakan akan diujikan dengan model pengguna serta model-model objek lain secara virtual. Untuk mempermudah pemindahan program dari simulasi ke robot fisik, kontroler robot akan dikembangkan secara terpisah dari lingkungan simulasi yang mana ketika pengujian, keduanya akan saling terhubung menggunakan sistem komunikasi antar proses yang ada di ROS 2. Diharapkan, lingkungan simulasi yang dibuat dapat membantu pengujian SARs dengan meminimalisir resiko, mengurangi biaya, dan menghemat waktu jika dibandingkan dengan melakukan pengujian secara langsung menggunakan robot fisik.

**Kata kunci**—Simulasi, Assistive Robotics, ROS2, Gazebo.

## I. PENDAHULUAN

Selama beberapa tahun terakhir, robot telah mengalami perkembangan yang signifikan dari robot beroda untuk edukasi [1] hingga robot manipulator untuk skala industri [2]. Salah satu bentuk perkembangan lain dari robot tersebut adalah *socially assistive robots* (SARs). SARs merupakan jenis robot dalam bidang *socially assistive robotics* yang menggabungkan aspek yang ada pada *assistive robotics* dan *socially interactive robotics* sehingga menjadikan SARs sebagai robot yang mampu memberikan bantuan kepada pengguna dalam bentuk interaksi sosial [3].

Namun, karena sifat dari SARs yang melibatkan interaksi langsung dengan pengguna, maka pengujian dari robot akan menjadi sulit dan beresiko bagi pengguna yang ikut terlibat dalam pengujian tersebut [4]. Salah satu solusi untuk mengatasi masalah tersebut adalah dengan melakukan pengujian secara virtual melalui simulasi robot. Selain bisa meminimalisir resiko, penggunaan simulasi robot sebagai media pengujian robot juga bisa mengurangi biaya yang dibutuhkan dan menghemat waktu pengujian selama pengembangan robot tersebut [5].

Hingga saat ini sudah ada beberapa simulator yang bisa digunakan untuk menjalankan simulasi robot seperti Webots [6],

Gazebo [7], V-REP [8], OpenAI Gym [9], dan lain sebagainya. Namun, simulator-simulator tersebut hanyalah platform yang secara umum digunakan untuk membantu pengembangan robot melalui simulasi virtual. Sedangkan pengembangan dari lingkungan simulasi dan kontroler robot untuk simulasi tersebut harus dibuat sendiri oleh pengembang robot.

Untuk itu, pada penelitian ini kami mengajukan penelitian terkait pengembangan lingkungan simulasi untuk pengujian SARs menggunakan ROS 2 dan Gazebo. ROS 2 dan Gazebo sendiri dipilih karena tersedianya banyak library yang dapat membantu pengembangan maupun pengujian robot, terutama untuk simulasi. Selain itu, dengan adanya ROS 2, kontroler robot yang diuji melalui simulasi bisa dengan mudah dipindahkan ke robot fisik untuk diuji secara langsung pada pengguna [5].

## II. PENELITIAN TERKAIT

Beberapa penelitian sebelumnya telah berhasil dalam mengembangkan lingkungan simulasi untuk robot menggunakan ROS (Pendahu ROS 2) dan Gazebo. Seperti yang dilakukan Qian et al. [10] yang mengembangkan simulasi untuk robot manipulator, Zhang et al. [11] yang mengembangkan simulasi untuk robot quadrotor UAV, dan Takaya et al. [5] yang mengembangkan lingkungan simulasi untuk pengujian terhadap mobile robot. Namun, berbeda dengan penelitian yang telah dilakukan sebelumnya, penelitian yang akan kami lakukan memilih menggunakan ROS 2 agar kontroler robot yang dibuat untuk simulasi memiliki performa yang lebih baik serta dapat bekerja secara real-time [12].

Selain itu, penelitian lain juga telah dilakukan oleh Erickson et al. [4] yang mengembangkan Assistive Gym, sebuah framework simulasi untuk *assistive robotics* berbasis OpenAI Gym. Framework simulasi tersebut kemudian digunakan oleh Clegg et al. [13] untuk mengembangkan metode learning melalui simulasi pada kolaborasi antara robot dengan manusia dalam membantu pemakaian baju pada manusia. Namun, karena tidak menggunakan ROS, kontroler robot yang dibuat untuk

simulasi yang menggunakan framework tersebut perlu dibuat ulang ketika akan diujikan secara langsung pada pengguna menggunakan robot fisik. Walaupun begitu, penelitian yang dilakukan oleh Zamora et al. [14] menunjukkan bahwa simulasi yang ada pada OpenAI Gym juga bisa diintegrasikan pada ROS dan Gazebo, sehingga tidak menutup kemungkinan bahwa Assistive Gym juga bisa digunakan bersamaan dengan ROS 2 dan Gazebo.

### III. ROS 2 DAN GAZEBO

#### A. Robot Operating System 2 (ROS 2)

Robot Operating System (ROS) [15] merupakan kumpulan dari libraries, drivers, dan tools yang mempermudah pengembangan sistem pada robot. ROS memiliki command tool seperti Linux, sistem komunikasi antar proses, dan berbagai macam packages yang berhubungan dengan pengembangan sistem pada robot. Proses yang dieksekusi pada ROS disebut sebagai Node, komunikasi antar proses yang dimiliki menggunakan model *publish/subscribe*, dan data komunikasi yang dikirimkan disebut sebagai Topic. Suatu proses Publisher mampu mengirimkan satu maupun lebih Topic, kemudian proses-proses lain yang melakukan subscribe pada suatu Topic bisa memperoleh isi dari Topic tersebut. Selain itu ada juga Service yang memiliki fungsi seperti Topic, hanya saja dilakukan secara dua arah. Service ini bekerja menggunakan model *client/server* dimana Service Client akan mengirimkan data permintaan dalam bentuk Request dan kemudian Service Server akan mengirimkan data balasan dalam bentuk Response.

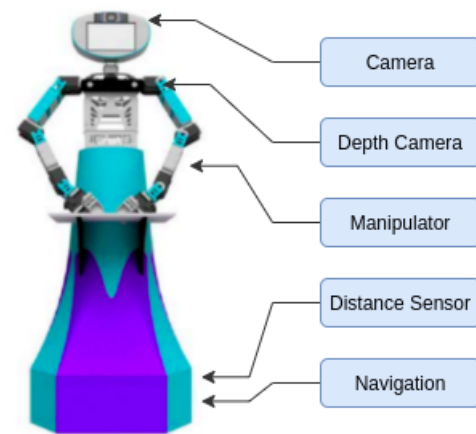
Generasi kedua dari Robot Operating System, ROS 2, merupakan kelanjutan dari ROS yang mengusung reliabilitas dan performa untuk penggunaan *real-time* sembari masih mendukung keunggulan yang dimiliki oleh ROS sebelumnya [12]. Untuk memenuhi kebutuhan reliabilitas dan performa untuk penggunaan *real-time* tersebut, ROS 2 menggunakan *Data Distribution Service* (DDS) [16] [17], standar industri untuk sistem komunikasi *real-time* dan *end-to-end middleware*, yang menggantikan sistem komunikasi antar proses yang dimiliki ROS sebelumnya.

#### B. Gazebo

Gazebo [7] merupakan bagian dari Player Project [18] yang memungkinkan sebuah simulasi robot dan aplikasi sensor bekerja di lingkungan simulasi indoor maupun outdoor tiga dimensi. Gazebo memiliki arsitektur *client/server* dan model *publish/subscribe* untuk sistem komunikasi antar prosesnya. Setiap objek simulasi di Gazebo dapat diasosiasikan dalam satu maupun lebih kontroler yang akan memproses perintah untuk mengatur dan menentukan keadaan dari suatu objek. Data yang dihasilkan oleh suatu kontroler akan dikirim ke *shared memory* menggunakan Gazebo interfaces (ifaces). Nantinya ifaces dari proses-proses lain dapat membaca data tersebut pada *shared memory*, sehingga memungkinkan komunikasi antar proses antara program yang mengontrol robot dan Gazebo, terlepas dari bahasa pemrograman yang digunakan.

## IV. DESAIN SISTEM

### A. Desain Robot yang Digunakan



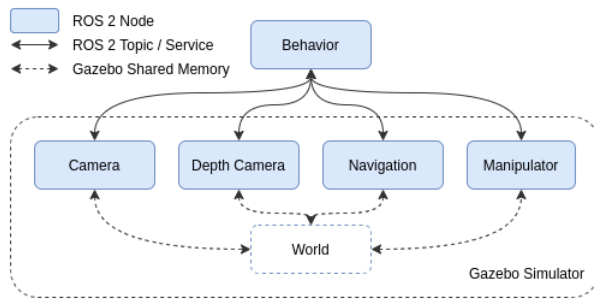
Gambar 1. Diagram desain robot Dienen.

Robot yang akan digunakan pada penelitian ini adalah robot Dienen yang merupakan kelanjutan dari robot IRIS [19][20] dengan penambahan desain dari robot ICHIRO [21] di bagian atas robot. Desain seperti ini secara umum dikenal sebagai desain *mobile humanoid robot* [22], yang merupakan desain gabungan antara robot mobile dan robot humanoid. Seperti yang terlihat pada Gambar 1, bagian bawah robot menyerupai robot mobile dengan penggerak *omnidirectional wheels* yang memungkinkan pergerakan robot secara *holonomic* ke segala arah[23], sedangkan bagian atas robot menyerupai robot humanoid yang terdiri atas badan, kepala, dan lengan. Dengan desain mobile humanoid robot ini, diharapkan pengguna bisa merasakan interaksi sosial yang lebih baik dengan robot karena memiliki bentuk mendekati manusia [24] sekaligus mempermudah navigasi dari robot ke berbagai tempat.

Robot Dienen dilengkapi dengan beberapa sensor seperti IMU (*inertial measurement unit*) untuk mengetahui orientasi dari robot, *rotary encoder* untuk melakukan perhitungan odometri dari robot, *distance sensor* untuk mendeteksi adanya objek lain di sekitar robot, sensor kamera di kepala untuk menangkap citra, dan sensor *depth camera* yang nantinya bisa digunakan untuk melakukan mapping dari ruangan. Selain itu robot ini juga dilengkapi dengan dua lengan seperti robot manipulator yang bisa diatur pada berbagai posisi dan orientasi [25]. Dengan adanya sensor dan aktuator ini diharapkan robot mampu melakukan tindakan *assistive* secara sosial sesuai dengan data yang didapatkan dari sensor yang ada.

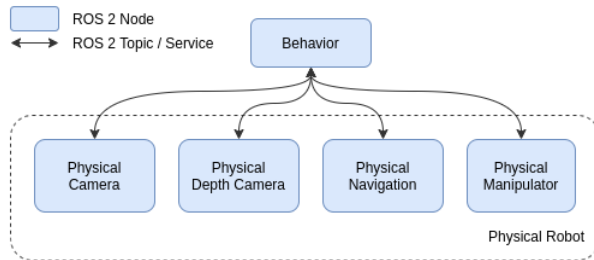
### B. Desain Kontroler Robot

Kontroler robot yang digunakan untuk simulasi ini akan dikembangkan menggunakan ROS 2. Kontroler tersebut akan dipisah menjadi beberapa bagian dalam bentuk *ROS 2 node* seperti yang terlihat pada Gambar 2. Setiap node yang ada akan terhubung satu sama lain menggunakan sistem komunikasi antar proses ROS 2 yang berupa *topics* dan *services*.



Gambar 2. Diagram sistem kontroler robot di simulasi.

Bagian utama dari kontroler robot tersebut adalah *node behavior* yang berisi program yang mengatur segala tindakan robot berdasarkan data yang didapat dari sensor yang ada di simulasi. Kemudian node Behavior tersebut akan terhubung dengan empat node lain yang merepresentasikan sensor dan aktuator yang ada pada robot. Keempat node tersebut akan terpasang di dalam cakupan simulator Gazebo sebagai *Gazebo plugins*, sehingga mampu digunakan untuk mengakses dan memanipulasi data yang ada di simulasi menggunakan sistem *shared memory* pada Gazebo [26].



Gambar 3. Diagram sistem kontroler robot untuk robot fisik.

Sistem ini dirancang secara terpisah agar *node Behavior* yang diujikan di lingkungan simulasi bisa langsung bekerja pada robot fisik. Seperti yang terlihat pada Gambar 3, transfer kontroler ke robot fisik dapat dilakukan dengan mengubah keseluruhan cakupan yang ada di simulator Gazebo, yang berupa empat node yang telah disebutkan sebelumnya, menjadi node yang memproses sensor dan aktuator yang ada pada robot fisik. Dengan ini pengujian yang dilakukan di simulasi bisa langsung diterapkan ketika diujikan pada robot fisik karena tidak perlunya pembuatan ulang kontroler yang menyesuaikan sistem yang ada pada robot.

## V. EKSPERIMEN

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum

urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

## VI. KESIMPULAN

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu,

sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetur at, consectetur sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

#### PUSTAKA

- [1] P. Gonçalves, P. Torres, C. Alves, F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, vol. 1, 2009.
- [2] M. Blatnický, J. Dižo, J. Gerlici, M. Sága, T. Lack, and E. Kuba, "Design of a robotic manipulator for handling products of automotive industry," *International Journal of Advanced Robotic Systems*, vol. 17, no. 1, 2020.
- [3] D. Feil-Seifer and M. J. Mataric, "Defining socially assistive robotics," in *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.*, 2005, pp. 465–468.
- [4] Z. Erickson, V. Gangaram, A. Kapusta, C. K. Liu, and C. C. Kemp, "Assistive gym: A physics simulation framework for assistive robotics," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 10 169–10 176.
- [5] K. Takaya, T. Asai, V. Kroumov, and F. Smarandache, "Simulation environment for mobile robots testing using ros and gazebo," in *2016 20th International Conference on System Theory, Control and Computing (ICSTCC)*, 2016, pp. 96–101.
- [6] O. Michel, "Cyberbotics ltd. webots™: Professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, p. 5, 2004.
- [7] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, 2004, pp. 2149–2154 vol.3.
- [8] E. Rohmer, S. P. N. Singh, and M. Freese, "V-rep: A versatile and scalable robot simulation framework," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 1321–1326.
- [9] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.
- [10] W. Qian, Z. Xia, J. Xiong, Y. Gan, Y. Guo, S. Weng, H. Deng, Y. Hu, and J. Zhang, "Manipulation task simulation using ros and gazebo," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, 2014, pp. 2594–2598.
- [11] M. Zhang, M. Lan, J. Lin, S. Wang, K. Liu, F. Lin, and B. M. Chen, "A high fidelity simulator for a quadrotor uav using ros and gazebo," in *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, 2015, pp. 002 846–002 851.
- [12] Y. Maruyama, S. Kato, and T. Azumi, "Exploring the performance of ros2," in *2016 International Conference on Embedded Software (EMSOFT)*, 2016, pp. 1–10.
- [13] A. Clegg, Z. Erickson, P. Grady, G. Turk, C. C. Kemp, and C. K. Liu, "Learning to collaborate from simulation for robot-assisted dressing," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2746–2753, 2020.
- [14] I. Zamora, N. Lopez, V. Vilches, and A. Cordero, "Extending the openai gym for robotics: a toolkit for reinforcement learning using ros and gazebo," *arXiv preprint arXiv:1608.05742*, 2016.
- [15] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," *ICRA Workshop on Open Source Software*, vol. 3, 2009.
- [16] G. Pardo-Castellote, "Omg data-distribution service: architectural overview," in *23rd International Conference on Distributed Computing Systems Workshops, 2003. Proceedings.*, 2003, pp. 200–206.
- [17] J. M. Schlesselman, G. Pardo-Castellote, and B. Farabaugh, "Omg data-distribution service (dds): architectural update," in *IEEE MILCOM 2004. Military Communications Conference, 2004.*, vol. 2, 2004, pp. 961–967 Vol. 2.
- [18] B. Gerkey, R. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," *Proceedings of the International Conference on Advanced Robotics*, 2003.
- [19] R. Dikairono, Setiawardhana, D. Purwanto, and T. A. Sardjono, "Cnn-based self localization using visual modelling of a gyrocompass line mark and omni-vision image for a wheeled soccer robot application," *International Journal of Intelligent Engineering and Systems*, 2020.
- [20] Muhtadin, R. M. Zanuair, I. K. E. Purnama, and M. H. Purnomo, "Autonomous navigation and obstacle avoidance for service robot," in *2019 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, 2019, pp. 1–8.
- [21] M. Muhtadin, M. Arrazi, S. Ali, T. Pratama, D. Wicaksono, A. Putra, I. Ari, A. Maulana, O. Bramastyo, S. Asshakina, M. Attamimi, M. Arifin, M. Hery Purnomo, and D. Purwanto, "Ichiro robots winning robocup 2018 humanoid teensize soccer competitions," in *RoboCup 2018: Robot World Cup XXII*, 2019, pp. 425–435.
- [22] Z. Mohamed and G. Capi, "Development of a new mobile humanoid robot for assisting elderly people," *Procedia Engineering*, vol. 41, pp. 345 – 351, 2012.
- [23] H. Oliveira, A. Sousa, A. Moreira, and P. Costa, "Dynamical models for omni-directional robots with 3 and 4 wheels," *ICINCO 2008 - Proceedings of the 5th International Conference on Informatics in Control, Automation and Robotics*, vol. 1, 2008.
- [24] S. Rossi, M. Staffa, and A. Tamburro, "Socially assistive robot for providing recommendations: Comparing a humanoid robot with a mobile application," *International Journal of Social Robotics*, 2018.
- [25] J. Iqbal, M. Ul Islam, and H. Khan, "Modeling and analysis of a 6 dof robotic arm manipulator," *Canadian Journal on Electrical and Electronics Engineering*, vol. 3, pp. 300–306, 2012.
- [26] "Gazebo plugins," 2021. [Online]. Available: [http://gazebo-sim.org/tutorials?tut=ros\\_gzplugins](http://gazebo-sim.org/tutorials?tut=ros_gzplugins)