

Pemantauan Posisi *Multi-Node* Menggunakan Aplikasi GUI pada LoRaWAN

Adam Fiqih Ristanto, Pauladie Susanto, Yosefine Triwidyastuti, Musayyanah*
Charisma Dimas Affandi, Fifin Ernawati

Teknik Komputer, Universitas Dinamika, Surabaya

*musayyanah@dinamika.ac.id

Abstrak— Perangkat *Internet of Things* (IoT) merupakan perangkat yang terkoneksi pada jaringan internet. Salah satu media transmisi yang *low power* berbasis IoT adalah LoRaWAN. LoRaWAN mampu menerima data dari banyak *user* pada jarak jauh dan terkoneksi langsung dengan database server melalui internet. Pada artikel ini, membahas mengenai monitoring posisi *multi-node* pada jaringan LoRaWAN. Terdapat tiga *node*, direpresentasikan sebagai *end node*. LoRaWAN sebagai satu *node server* atau disebut LoRa Gateway. Pada *node server* terpasang aplikasi berbasis *Graphic User Interface* (GUI) yang dapat memantau posisi *end node* secara *real time*. Pengujian pada artikel ini, meliputi pengujian transmisi *packet loss* dari *end node* ke *node server*, dan pengujian aplikasi untuk memenuhi *Human Computer Interaction* (HCI). Hasil transmisi tiga *node end* ke LoRa Gateway berturut-turut adalah 6,04%, 1,13%, dan 1,9%. Pengujian kelayakan aplikasi dari segi sudut pandang pengguna telah memenuhi standart HCI dengan nilai sangat setuju 61% dan setuju 50%.

Kata Kunci— *Graphic User Interface, Human Computer Interaction, IoT, LoRaWan, LoRa, Node, packet loss*

DOI: 10.22441/jte.2023.v14i2.002

I. PENDAHULUAN

Salah satu teknik memantau jarak jauh dengan memanfaatkan teknologi berbasis *Internet of Things* (IoT). IoT menghubungkan sensor, komputer, jaringan, *Radio Frequency Identification* (RFID), sistem tertanam, dan teknologi komunikasi [1]. IoT menawarkan teknologi dengan daya yang rendah, kecepatan transmisi tinggi pada jarak jauh, kompleksitas rendah, dan penerapan perangkatnya terjangkau. Saat ini, penerapan IoT dibarengi dengan *Low Power Wide Area Network* (LPWAN). LPWAN merupakan salah satu teknologi berdaya rendah seperti Sigfox, Zigbee, NB-IoT, dan LoRaWan. LoRaWan dapat terintegrasi dengan beberapa platform jaringan di dunia, penelitian terkait topik ini banyak dilakukan di lima tahun terakhir ini [2].

LoRaWAN menyediakan mekanisme komunikasi banyak perangkat dan gateway. LoRaWAN dianggap salah satu solusi dari permasalahan IoT, karena terdapat pengaturan *Spreading Factor* (SF) yang dapat memperkecil *jammung* dari proses komunikasi untuk nilai SF tinggi [3]. Penerapan IoT dengan LoRa telah dilakukan pada [4] dan [5] untuk monitoring data sensor secara jarak jauh. Kedua penelitian tersebut menggunakan perangkat LoRa sebagai satu perangkat pengirim dan satu penerima, tidak menerapkan multi komunikasi antar perangkat. Pada artikel ini membahas kinerja LoRaWan sebagai

penerima dari multi perangkat pengirim dan monitoring data posisi aplikasi GUI secara *real time*.

II. PENELITIAN TERKAIT

LoRaWAN merupakan teknologi LPWAN yang banyak diadopsi oleh kalangan industri dan akademik saat ini, karena dapat beroperasi pada *unlicensed* frekuensi. Pengaturan kecepatan, konsumsi daya dan *throughput* dapat disesuaikan dengan aplikasi IoT [6]. LoRaWan dianggap salah satu solusi dari berbagai permasalahan IoT. Banyak penelitian yang menggunakan LoRaWan seperti pada [7], untuk monitoring data sensor tekanan darah, kadar glukosa dan suhu tubuh di lingkungan urban atau perkotaan, dengan jangkauan jarak 33 km². Pada [8] mengimplementasikannya pada *smart city* untuk 20 perangkat pengirim dengan keberhasilan penerimaan 99%.

Graphical User Interfaces (GUI) membuat pengguna dapat berinteraksi dengan aplikasi perangkat lunak melalui elemen desain grafis seperti gambar, teks, dan *widget* [9]. Pemantauan data terasa mudah dengan perangkat lunak yang terintegrasi dengan perangkat keras, untuk menampilkan data secara *real time*. Perlu adanya penilaian terhadap aplikasi yang telah dibuat yang dikenal dengan *Human Computer Interaction* (HCI). HCI merupakan indikator penilaian kemudahan atau kepuasan penggunaan terhadap teknologi atau perangkat lunak. HCI bergantung pada *representative* pengguna atau penilaian pengguna [10]. HCI dapat dilakukan dengan menyebarkan angket pada pengguna dengan memberikan daftar pertanyaan terkait perangkat lunak tersebut.

Berdasarkan penelitian-penelitian tersebut LoRaWan sangat tepat untuk pemantauan data jarak jauh. Artikel ini mengintegrasikan perangkat keras dengan perangkat lunak dengan mentransmisikan data posisi secara *real time*. Perangkat keras terdiri dari Gateway dan tiga *node* pengirim. Perangkat lunak adalah aplikasi GUI untuk memvisualkan data posisi tersebut. Selain itu, artikel ini juga menganalisa skalabilitas performa jaringan Gateway, dengan tiga perangkat pengirim, memaparkan pembuatan aplikasi GUI dan menganalisa standar HCI dari aplikasi tersebut.

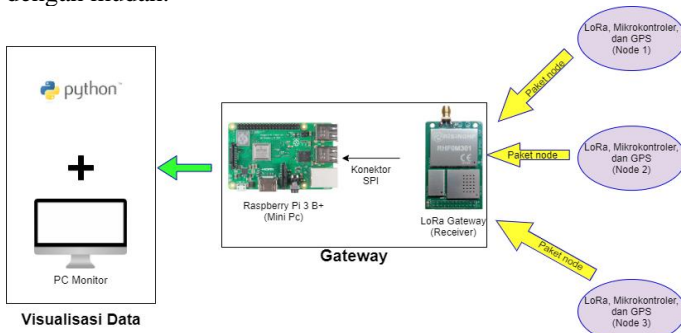
Kontribusi dari penelitian ini adalah membuat skenario dan algoritma multi komunikasi antara LoRaWAN dengan tiga *node* pengirim dengan mentransmisikan data posisi secara *real time*. Penelitian ini juga mengintegrasikan LoRaWAN dengan aplikasi GUI untuk menampilkan data posisi dalam bentuk diagram konstelasi dua dimensi.

Bagian Bab I dan II membahas terkait urgencitas penelitian dari artikel serta penelitian terdahulu, Bab III membahas

skenario sistem komunikasi, Bab IV mengenai analisis dari hasil pengujian. Bab V adalah kesimpulan yang kemudian diikuti dengan saran, ucapan terimakasih dan referensi.

III. METODOLOGI PENELITIAN

Gambar 1 merepresentasikan skenario komunikasi perangkat pengirim yang selanjutnya disebut node pengirim, dengan perangkat penerima yang selanjutnya disebut node *Gateway*. Berdasarkan skenario tersebut, terdapat tiga node pengirim, satu node *Gateway* dan satu perangkat lunak untuk visualisasi data. Node pengirim terdiri dari mikrokontroler yang terhubung langsung dengan LoRa dan sensor GPS. Node tersebut akan mengirimkan data berupa posisi dan data pilot sebagai identitas pembeda setiap node pengirim. Node *Gateway* merupakan salah satu media komunikasi LoRa yang terhubung dengan internet. Node ini mampu menerima data dari multi node pengirim. Selain itu, node *Gateway* juga terhubung dengan mini komputer *Raspberry Pie* untuk proses kompilasi dan *running* program. *Operating system* yang ada didalamnya juga terpasang aplikasi GUI dengan menggunakan *library* dari *python* yaitu Tkinter. Informasi yang ditampilkan pada GUI adalah berupa diagram konstelasi dua dimensi dari data posisi node pengirim, lokasi dari node pengirim dapat dipantau dengan mudah.

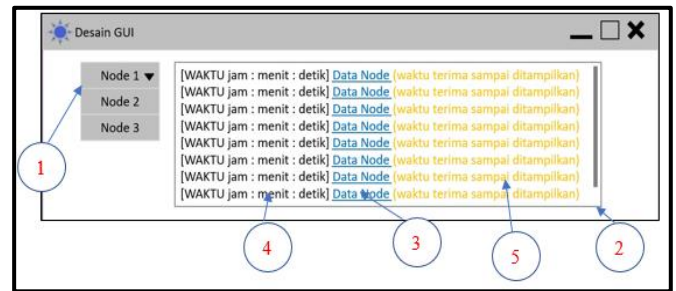


Gambar 1. Perancangan Sistem Komunikasi

A. Desain Tampilan GUI

Visualisasi tampilan bagian depan dari aplikasi GUI terlihat pada Gambar 2. Tampilan GUI tersebut memberikan informasi terkait identitas node pengirim, posisi node saat ini, dan waktu. Tampilan Visual GUI ini, muncul secara *real time* pada mini komputer. Adapun tampilan menu GUI terdiri dari:

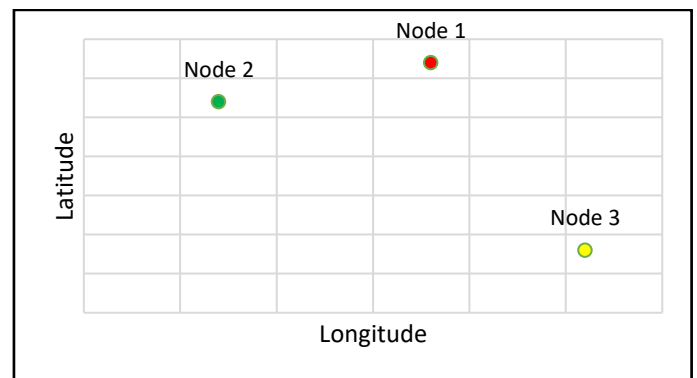
1. *Combo Box/ Box* Ganda yang digunakan untuk menampilkan pilihan node pengirim, yang akan ditampilkan informasinya.
2. *Multi Textbox* menampilkan seluruh informasi data dan mencatat waktu terima data, data posisi sensor GPS.
3. Pada tautan *Data Node* menampilkan informasi posisi node pengirim pada *google map*.
4. Waktu terima data dari node pengirim pada node *Gateway*.
5. Informasi waktu terima data dari node pengirim sampai ditampilkan pada aplikasi GUI.



Gambar 2. Tampilan Utama GUI

B. Posisi Node pada Tampilan Map Koordinat

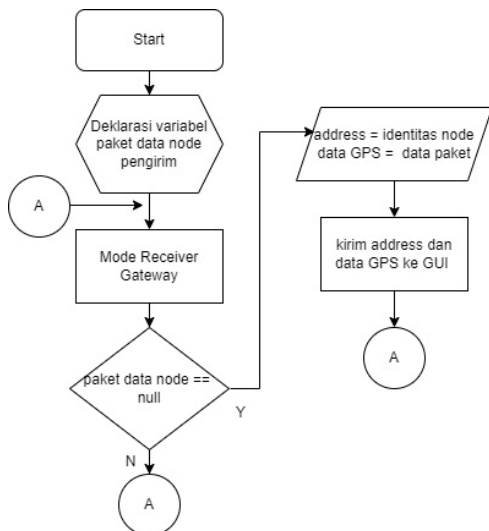
Data posisi node pengirim akan ditampilkan pada koordinat 2 dimensi *longitude* dan *latitude*, dengan cara melakukan *double* klik pada tautan *Data Node*. Tampilan posisi node pengirim dapat dilihat pada Gambar 3. Bagian vertikal adalah posisi *latitude*, sedangkan horizontal adalah *longitude*. Tampilan koordinat akan terpisah dengan tampilan utama dari visual GUI.



Gambar 3. Tampilan koordinat dari node pengirim

C. Algoritma Komunikasi Gateway

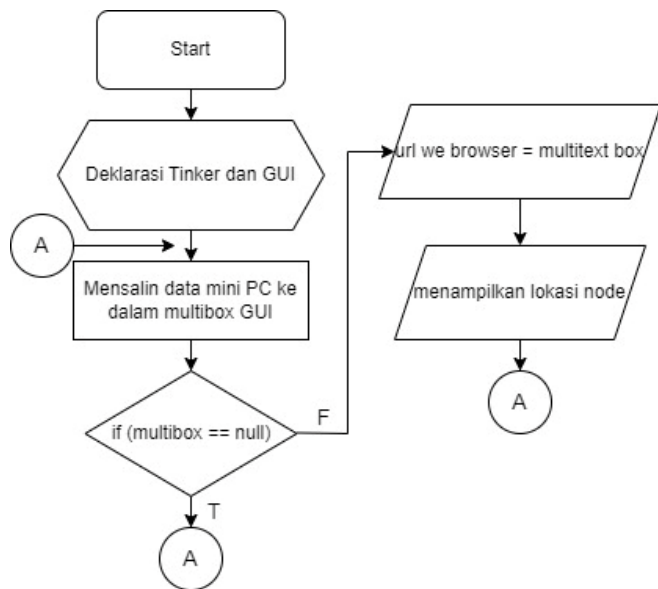
Komunikasi node *Gateway* merupakan komunikasi terpenting dari penelitian ini, dibuat sebuah algoritma *Gateway* dengan tujuan tidak adanya tabrakan data dari ketiga node pengirim tersebut. *Flowchart* algoritma ditunjukkan pada Gambar 4. Data yang diterima dari setiap node pengirim oleh *Gateway* terdiri dari identitas node dan data GPS. *Gateway* mengaktifkan *mode receiver* kemudian terdapat pengecekan data setiap node pengirim dengan memastikan data yang diterima tidak null atau tidak kosong. Jika data tidak null, maka data diterima. Kemudian data tersebut dipisah antara identitas node pengirim dan data GPS untuk ditampilkan di aplikasi GUI. Kemudian *Gateway* kembali mengaktifkan *mode receiver*nya.



Gambar 4. Algoritma node Gateway

D. Algoritma Tampilan GUI

Data yang sudah diterima dan diolah oleh node Gateway, akan diterima di bagian *multibox* GUI. Data dari Gateway disinkronkan dengan GUI dengan cara menyamakan tipe data data tersebut. Algoritma pada GUI ditunjukkan pada Gambar 5. Pada *multi box* GUI akan menampilkan data node dan data GPS dalam bentuk tautan. Ketika *User* mengklik tautan tersebut, maka data posisi dari node pengirim dapat dimunculkan pada *form* yang berbeda dari *multibox* utama dari GUI.



Gambar 5. Algoritma GUI

IV. HASIL DAN ANALISA

Hasil pengujian dan analisis dari penelitian ini terdiri dari paparan pembuatan aplikasi GUI, uji performa komunikasi node pengirim dengan node Gateway, dan penilaian HCI dari angket pengguna aplikasi GUI.

A. Pembuatan Aplikasi GUI

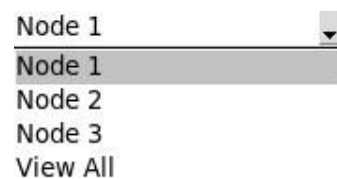
Langkah awal membuat GUI adalah inialisasi dan pembuatan dimensi awal pada GUI seperti pada Gambar 6. Dimensi tersebut terdiri dari *combo box*, *check button* dan *label* terlihat pada Gambar 7. *Combo box* terdapat tiga pilihan node pengirim dan *view all* untuk melihat semua data dari node pengirim seperti pada Gambar 8. Informasi dari node pengirim atau semua node pengirim terlihat pada Gambar 9 sampai Gambar 10. Tampilan data posisi ditunjukkan dengan visual dua dimensi informasi node pengirim yang ditunjukkan pada Gambar 11 dengan form baru dari tampilan utama GUI, sedangkan Gambar 12 menampilkan informasi seluruh node pengirim.



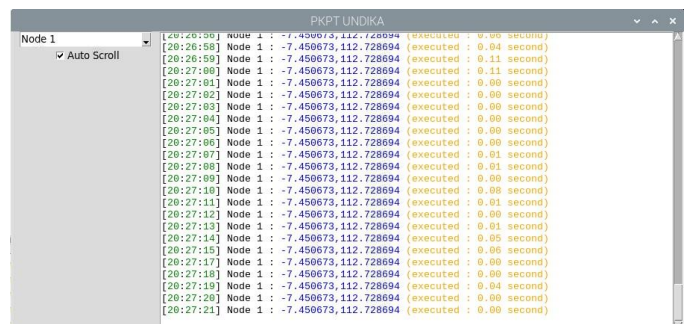
Gambar 6. Tampilan kosong dari GUI



Gambar 7. Tampilan aplikasi GUI



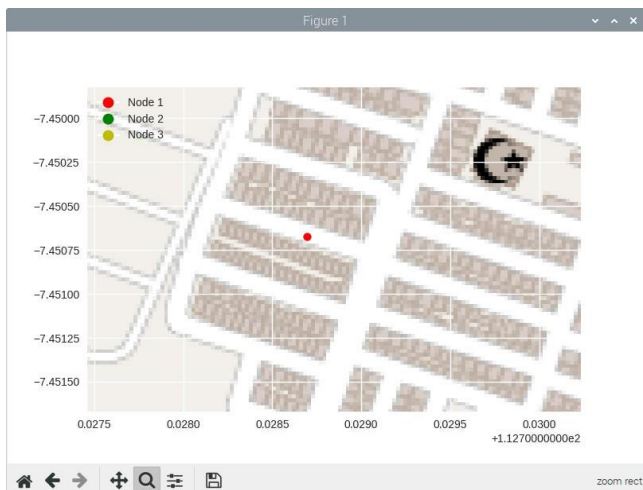
Gambar 8. Tampilan menu node pengirim



Gambar 9. Tampilan menu *combo box* node pengirim



Gambar 10. Tampilan menu *combo box* "View All"



Gambar 11. Tampilan form baru koordinat node pengirim



Gambar 12. Tampilan form baru "View All"

Setuju (SS), Setuju (S), Normal (N), Tidak Setuju (TS), dan sangat Tidak Setuju (STS). Hasil Angket terbanyak pada penilaian desain dan efisiensi dari aplikasi, yaitu 61% sangat setuju dan 50% setuju. Hal ini menunjukkan bahwa aplikasi GUI pemantau data node wisatawan memenuhi standar HCI. Hasil pengolahan angket disajikan pada Tabel 1.

Tabel 1. Hasil Angket Responden

No	Parameter	STS	TS	N	S	SS	Suara Terbanyak
1	Text And Font Usage	0,0	7,7	23,1	30,8	38,5	SS
		7,7	7,7	7,7	38,5	38,5	SS & S
2	Background And Foreground	0,0	0,0	30,8	46,2	23,1	S
		7,7	0,0	7,7%	38,5 %	46,2 %	SS
3	Button	0,0	0,0	23,1 %	23,1 %	53,9 %	SS
		0,0	15,4	0,0%	46,2 %	38,5 %	S
4	Menu Sturcture And Navigation	0,0	0,0	30,8 %	23,1 %	46,2 %	SS
		0,0	0,0	7,7%	15,4 %	76,9 %	SS
5	Learnability And User Friendly	0,0	15,4	23,1 %	38,5 %	23,1 %	S
		0,0	0,0	30,8 %	30,8 %	38,5 %	SS
6	Flexibility	0,0	0,0	23,1 %	46,2 %	30,8 %	S
		0,0	0,0	23,1 %	23,1 %	53,9 %	SS
7	Robustness	0,0	7,7	15,4 %	53,9 %	23,1 %	S
		7,7	7,7	15,4 %	30,8 %	38,5 %	SS
Presentase Jawaban Terbanyak							SS = 61%; S = 50%

Perhitungan pada Tabel 1, untuk presentasi angket sangat setuju (SS) didapat berdasarkan persamaan (1). Presentasi angket setuju (S) dihitung sesuai persamaan (2).

$$SS(\%) = \frac{\text{Jawaban terbanyak (1)Sangat Setuju}}{\text{Jumlah semua pertanyaan}} \times 100\% \quad (1)$$

$$= \frac{11}{18} \times 100\% = 61\%$$

$$S(\%) = \frac{\text{Jawaban terbanyak (2)Setuju}}{\text{Jumlah semua pertanyaan}} \times 100\% \quad (2)$$

$$= \frac{9}{18} \times 100\% = 50\%$$

C. Pengujian transmisi data pada LoRaWAN

Pengujian loss dari transmisi data antara node pengirim dengan node Gateway dilakukan pada jarak 100 meter dengan durasi 5 menit. Data yang dikirimkan oleh setiap node pengirim sebanyak 260 byte. Hasil pengujian yang ditunjukkan pada

Tabel 2. Pengujian ini juga merekam waktu proses node *Gateway* menampilkan data pada aplikasi GUI.

Tabel 2. Hasil Pengujian Loss

Node	<i>Packet Loss</i>	<i>T_proses</i>
N1	6,04%	0,045 detik
N2	1,13%	0,09 detik
N3	1,9%	0,04 detik
Rata-rata	3,02%	0,058 detik

Berdasarkan rata-rata *packet loss* pada Tabel 2, kualitas komunikasi LoRaWAN dengan ketiga node pengirim tergolong sangat baik. Rata-rata dari *T_proses* atau waktu proses *node Gateway* menampilkan data pada aplikasi GUI sangat cepat, dikarenakan kecepatan dari mini komputer pada node *Gateway*.

V. KESIMPULAN

Berdasarkan hasil pengujian dan analisis disimpulkan bahwa LoRaWAN mempunyai performa transmisi data yang sangat baik dengan *packet loss* 6,04% pada node pengirim 1 (N1), 1,13% pada node pengirim 2 (N2) dan 1,9% pada node pengirim 3 (N3). Visualisasi data posisi dari aplikasi GUI dengan rata-rata waktu komputasi yang sangat cepat yaitu 0,058 detik. Penilaian HCI dari GUI komunikasi *multi-node* mencapai 61% dengan angket sangat setuju dan 50% untuk angket setuju.

UCAPAN TERIMA KASIH

Penelitian ini merupakan salah satu sub topik dari topik besar penelitian hibah Kementerian Pendidikan dan Budaya

(Kemdikbud) dengan skema Penelitian Kerjasama Antar Perguruan Tinggi (PKPT) dengan nomor kontrak 004/ST-PPM/KPJ/VII/2021.

DAFTAR PUSTAKA

- [1] S. W. Nourillean, M. D. Hassib, and Y. A. Mohammed, "Internet of things based wireless sensor network: a review," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 27, no. 1, pp. 246–261, 2022, doi: 10.11591/ijeecs.v27.i1.pp246-261.
- [2] J. Haxhibeqiri, E. De Poorter, I. Moerman, and J. Hoebeke, "A survey of LoRaWAN for IoT: From technology to application," *Sensors (Switzerland)*, vol. 18, no. 11, 2018, doi: 10.3390/s18113995.
- [3] A. Lavric and V. Popa, "A LoRaWAN: Long Range Wide Area Networks study," *2017 11th Int. Conf. Electromechanical Power Syst. SIEMEN 2017 - Proc.*, vol. 2017-Janua, pp. 417–420, 2017, doi: 10.1109/SIEMEN.2017.8123360.
- [4] R. Angriawan and N. Anugraha, "Sistem Pelacak Lokasi Sapi Dengan Sistem Komunikasi Lora," pp. 33–39, 2016.
- [5] E. D. Widiyanto, "Rancang Bangun Sistem Pemantau Kualitas Udara Menggunakan Arduino Dan Lora Berbasis Jaringan Sensor Nirkabel," no. 1, pp. 6–14, 2020.
- [6] N. Matni, J. Moraes, H. Oliveira, and D. Rosário, "LoRaWAN Gateway Placement Model for Dynamic," 2020, doi: 10.3390/s20154336.
- [7] A. Mdhaffar, T. Chaari, K. Larbi, M. Jmaiel, and B. Freisleben, "IoT-based Health Monitoring via LoRaWAN," no. July, pp. 6–8, 2017, doi: 10.1109/EUROCON.2017.8011165.
- [8] S. J. Ossont, M. Apetroaie-cristea, and S. J. Cox, "LoRaWAN for Smart City IoT Deployments: A Long Term Evaluation," 2020, doi: 10.3390/s20030648.
- [9] J. Chen *et al.*, "Object detection for graphical user interface: old fashioned or deep learning or a combination?," in *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, Nov. 2020, pp. 1202–1214, doi: 10.1145/3368089.3409691.
- [10] S. Seinfeld, T. Feuchtner, A. Maselli, and J. Müller, "User Representations in Human-Computer Interaction," *Human-Computer Interact.*, vol. 36, no. 5–6, pp. 400–438, 2021, doi: 10.1080/07370024.2020.1724790.