

PERANCANGAN ANTARMUKA APLIKASI PREDIKSI PM2.5

Given Putra¹⁾ Tony²⁾

¹⁾Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Tarumanagara
Jl. Letjen S. Parman No. 1, Grogol Petamburan, Jakarta Barat 11440
Email: given.535220143@stu.untar.ac.id

²⁾Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Tarumanagara
Jl. Letjen S. Parman No. 1, Grogol Petamburan, Jakarta Barat 11440
Email: tony@fti.untar.ac.id

ABSTRAK

Polusi udara merupakan krisis kesehatan global, dengan PM2.5 sebagai polutan paling berbahaya yang berkontribusi pada 8,1 juta kematian pada tahun 2021. Di Indonesia, yang merupakan salah satu negara dengan polusi PM2.5 tertinggi, masih terdapat kesenjangan antara ketersediaan data kualitas udara dan manfaatnya bagi masyarakat. Alat pemantau yang ada seringkali hanya menyajikan data saat itu, bukan prediksi yang dibutuhkan untuk tindakan preventif. Penelitian ini bertujuan untuk merancang antarmuka aplikasi prediksi PM2.5 yang dapat diterima dan mudah digunakan oleh masyarakat umum. Metode perancangan yang digunakan adalah software development life cycle dengan model waterfall. Proses desain awal dilakukan menggunakan Figma, sedangkan implementasi antarmuka dikembangkan menggunakan framework Streamlit dengan bahasa pemrograman Python. Hasil realisasi mengubah desain awal yang terdiri dari empat halaman (dashboard, information, evaluation, about) menjadi tiga halaman fungsional: prediksi, validasi, dan tentang. Pengujian dilakukan menggunakan metode black box testing dan menghasilkan luaran sesuai ekspektasi yang diinginkan. Antarmuka yang dihasilkan ini dirancang untuk menyajikan data prediksi yang kompleks dalam format yang mudah dipahami, guna membantu meningkatkan kesadaran publik akan bahaya PM2.5.

Keywords

User interface, PM2.5., forecasting, air pollution

1. Pendahuluan

Polusi udara merupakan salah satu krisis kesehatan dan lingkungan yang tengah dihadapi masyarakat global. Namun, karena polusi udara cenderung bersifat tak terlihat, masih banyak yang meremehkan dampaknya. Padahal, di tahun 2021, polusi udara berperan sebagai faktor risiko utama kedua dengan jumlah kematian terbanyak di dunia, setelah tekanan darah tinggi. Di tahun tersebut, angka kematian global yang disebabkan oleh polusi udara mencapai 8,1 juta orang. Di antara berbagai jenis polusi udara yang beredar, *particulate matter 2.5* (PM2.5) menduduki peringkat pertama sebagai yang

paling berbahaya. Hal ini terlihat karena 96% kematian yang disebutkan sebelumnya terjadi akibat PM2.5 [1].

Permasalahan polusi udara semakin relevan ketika dibawa ke ranah nasional. Indonesia berada pada urutan pertama sebagai negara paling berpolusi di Asia Tenggara dan menduduki peringkat ke-15 secara global pada tahun 2024. Rata-rata PM2.5 tahunan di Indonesia mencapai 35,5 mikrogram per meter kubik ($\mu\text{g}/\text{m}^3$), tujuh kali lebih tinggi dari standar yang telah ditetapkan *World Health Organization* (WHO), yaitu $5 \mu\text{g}/\text{m}^3$. Perkotaan besar di Indonesia secara konsisten menempatkan diri sebagai kota dengan tingkat PM2.5 tertinggi di dunia. Salah satu contohnya adalah Kota Tangerang Selatan, yang mencapai tingkat PM2.5 hingga $56 \mu\text{g}/\text{m}^3$, sebelas kali lebih tinggi dari panduan WHO [2][3].

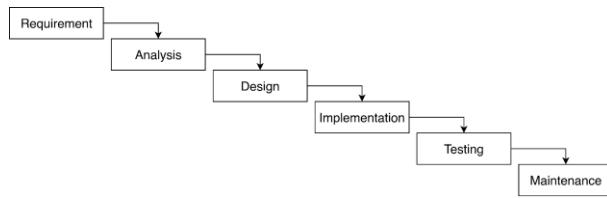
Melihat urgensi tersebut, diperlukan peningkatan kesadaran akan ancaman yang bisa ditimbulkan dari PM2.5. Masih terdapat kesenjangan antara ketersediaan informasi kualitas udara dengan manfaatnya bagi masyarakat. Beberapa perkotaan besar telah menyediakan alat pemantau kualitas udara, namun alat terkait hanya berfungsi sebagai alat pengukur saat itu, bukan sebagai alat yang menyajikan informasi yang akan datang. Untuk mengatasi keterlambatan informasi tersebut, diperlukan sebuah sistem yang dapat memprediksi tingkat PM2.5. Ketersediaan prediksi yang akurat memungkinkan masyarakat untuk mengambil tindakan yang tepat sasaran.

Namun, sebuah sistem untuk melakukan prediksi saja tidak cukup. Agar dapat digunakan dengan baik oleh masyarakat, sistem prediksi tersebut harus dibalut dengan antarmuka yang menarik dan mudah digunakan oleh kalangan luas. Maka dari itu, tujuan dari artikel ini adalah untuk merancang antarmuka aplikasi prediksi PM2.5 yang dapat diterima dan digunakan oleh masyarakat umum, sebagai upaya untuk memerangi permasalahan polusi udara dan PM2.5 yang sedang dialami. Beberapa artikel serupa sudah pernah disusun sebelumnya, salah satunya oleh Halim *et al.* [4] yang merancang sistem prediksi zat pencemar udara bersama dengan antarmukanya. Artikel lain yang menjadi sumber inspirasi disusun oleh Ratchagit [5], yang menyusun model prediksi PM2.5 yang kokoh dan berpotensi untuk diterapkan sebagai sebuah aplikasi prediksi dan

Handhayani [6] yang meneliti mengenai hubungan antara fitur meteorologi dengan PM2.5 di Kota Jakarta.

2. Metode Penelitian

Perancangan antarmuka aplikasi dilakukan menggunakan metode *software development life cycle* dengan model *waterfall*. Model *waterfall* merupakan proses pengembangan perangkat lunak yang bekerja secara berurutan, di mana setiap tahap dalam siklus hidup harus diselesaikan secara keseluruhan sebelum beralih ke tahap berikutnya [7]. Tahapan model *waterfall* yang diterapkan untuk mengembangkan antarmuka dapat dilihat pada **Gambar 1**. Desain awal dari antarmuka dibuat menggunakan perangkat lunak Figma, sedangkan untuk pengembangan antarmuka dilakukan menggunakan *framework* Streamlit dengan bahasa pemrograman Python pada perangkat lunak Visual Studio Code.

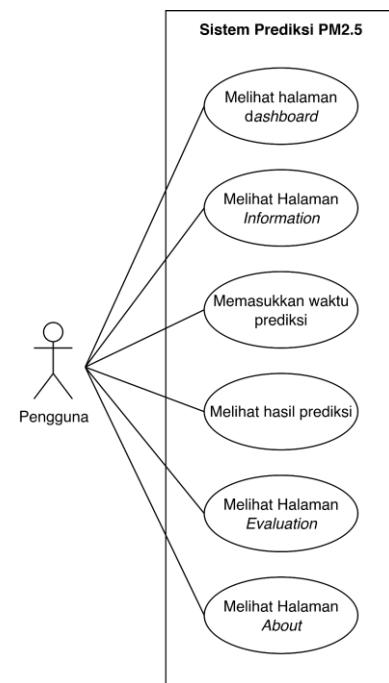


Gambar 1 Alur Metode Waterfall

Requirement merupakan tahapan awal yang untuk menentukan apa ketentuan dan tujuan dari antarmuka. *Analysis* merupakan tahap untuk melakukan analisis terhadap ketentuan yang telah ditetapkan sebelumnya dan bagaimana cara untuk memenuhinya. *Design* merupakan tahap untuk memberikan rincian spesifikasi desain teknis dari antarmuka. Spesifikasi tersebut menjelaskan secara visual bagaimana persyaratan akan terpenuhi oleh solusi teknis. *Implementation* merupakan proses untuk merealisasikan desain melalui proses pemrograman. Pada tahapan ini, spesifikasi dari desain akan diimplementasikan untuk menghasilkan produk akhir yang siap untuk digunakan. *Testing* merupakan tahapan untuk melakukan uji coba terhadap antarmuka. Uji coba dilakukan dengan tujuan untuk mengetahui apakah fitur pada antarmuka telah memenuhi syarat yang ditentukan. Terakhir, *maintenance*, merupakan tahapan akhir pada alur, di mana perlu dilakukan pemeliharaan untuk memastikan fungsionalitas antarmuka tersebut [7].

3. Hasil dan Pembahasan

Pertama, dirancang terlebih dahulu *use case diagram* untuk antarmuka yang ingin dirancang. *Use case diagram* adalah diagram yang digunakan untuk menggambarkan kebutuhan fungsional dari perspektif pengguna. Diagram ini memvisualisasikan berbagai skenario penggunaan yang sedang dalam proses pengembangan [8].



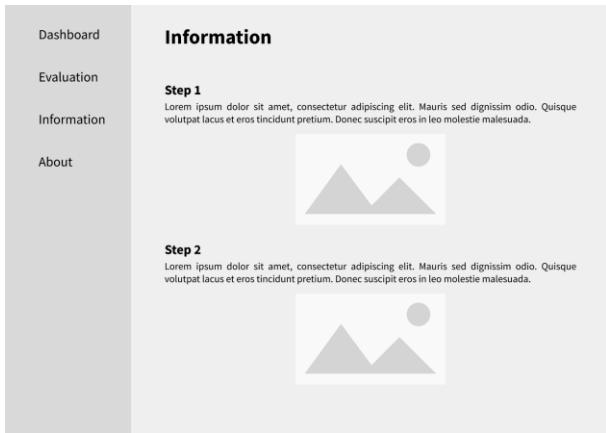
Gambar 2 Use Case Diagram Rancangan Antarmuka

Gambar 2 memperlihatkan desain use case diagram yang merincikan bagaimana interaksi antar pengguna dengan antarmuka terdiri dari empat halaman, yaitu: (i) halaman *dashboard*; (ii) halaman *information*; (iii) halaman *evaluation*; dan (iv) halaman *about*. **Gambar 3** memperlihatkan halaman *dashboard* yang berfungsi sebagai halaman utama, yang memungkinkan pengguna untuk memberikan masukan berupa waktu yang ingin dilihat, lalu rancangan akan memberikan luaran berupa kondisi PM2.5 di saat itu dan juga prediksinya.



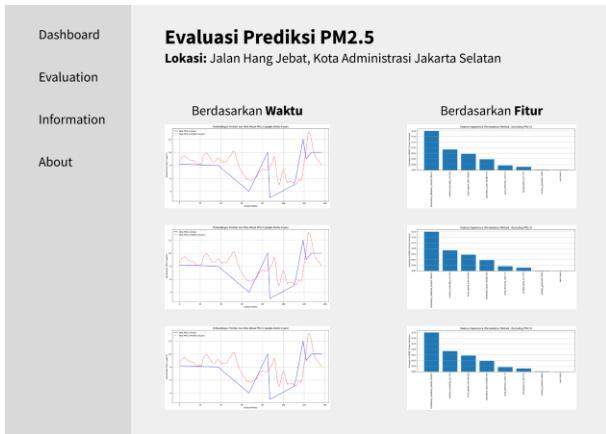
Gambar 3 Desain Halaman Dashboard

Untuk halaman berikutnya, adalah halaman *information* yang bertujuan memberikan informasi seputar cara penggunaan aplikasi langkah demi langkah, terlihat pada **Gambar 4**.



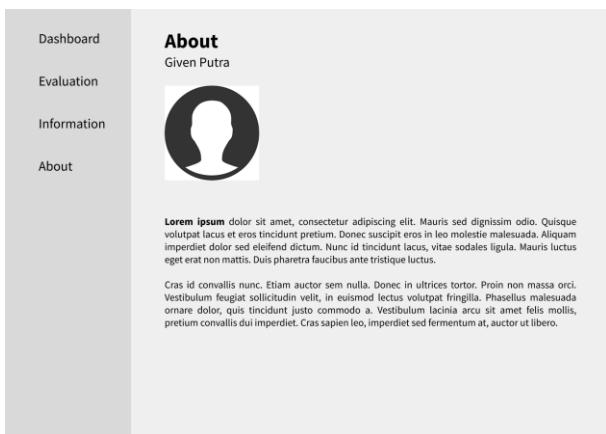
Gambar 4 Desain Halaman *Information*

Tersedia halaman *evaluation* yang menyediakan gambaran evaluasi prediksi yang dihasilkan sebelumnya, terlihat pada **Gambar 5**.



Gambar 5 Desain Halaman *Evaluation*

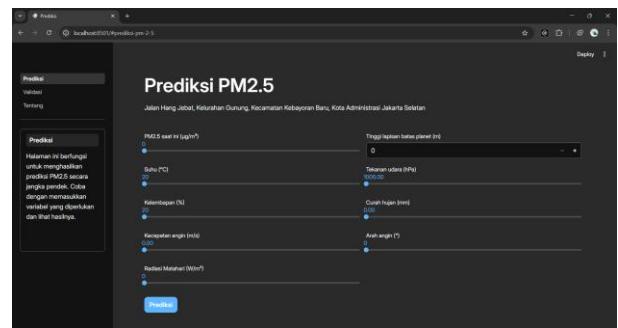
Terakhir, terdapat halaman yang berfungsi sebagai tambahan, yaitu halaman *about*, terlihat pada **Gambar 6**. Halaman *about* menceritakan secara singkat kisah di balik antarmuka serta pihak yang bertanggung jawab dan berkontribusi di dalamnya.



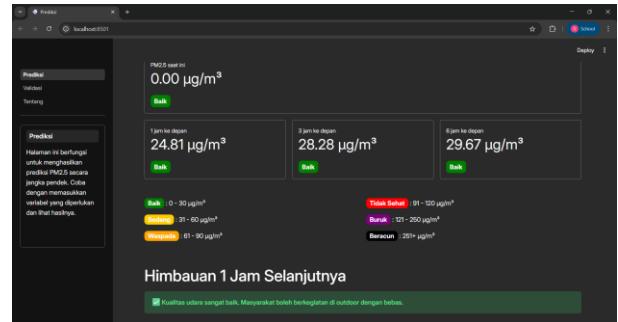
Gambar 6 Desain Halaman *About*

Terkait desain antarmuka tersebut, terdapat perubahan halaman-halaman pada realisasinya. Hasil realisasi dari desain antarmuka dari tiga halaman, yaitu: (i) halaman prediksi; (ii) halaman validasi; dan (iii) halaman tentang.

Halaman prediksi yang terlihat pada **Gambar 7** merupakan sebagian dari realisasi halaman *dashboard*. Halaman ini berperan sebagai halaman dan fitur utama pada rancangan. Alih-alih memberikan masukan berupa waktu yang diinginkan seperti sebelumnya, sekarang halaman prediksi menerima masukan berupa fitur-fitur terkait pada saat ini. Luaran masih sama seperti pada desain, di mana akan menghasilkan prediksi tingkat PM2.5 beserta kategorinya. Perbedaan lain pada halaman ini adalah grafik pada desain diganti dengan fitur himbauan kepada masyarakat terkait prediksi kategori PM2.5 terlihat pada **Gambar 8**.

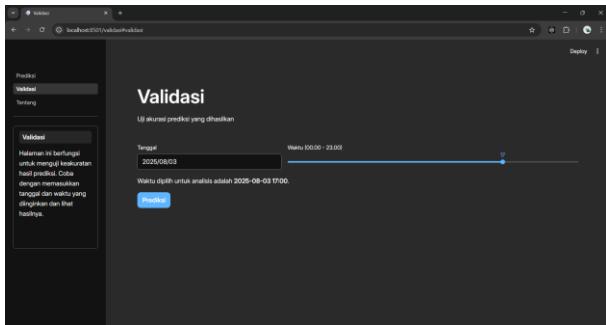


Gambar 7 Hasil Realisasi Halaman Prediksi

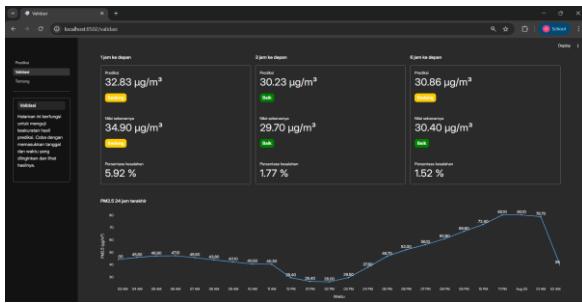


Gambar 8 Luaran Halaman Prediksi

Fitur pada desain rancangan di mana pengguna dapat memilih waktu yang diinginkan berdasarkan data historis dipindahkan menjadi halaman validasi. Halaman validasi terlihat pada **Gambar 9** berfungsi untuk menguji akurasi prediksi. Halaman ini juga merupakan bentuk realisasi dari halaman *evaluation* pada desain rancangan. Pengguna akan diminta untuk memberikan masukan berupa waktu historis yang diinginkan untuk memberikan prediksi beserta nilai aktualnya. Lalu sebagai pembanding terdapat juga metrik evaluasi untuk mengetahui sebesar apa kesalahan prediksi tersebut. Selain itu, pengguna juga dapat melihat grafik historis dari kadar tingkat PM2.5. **Gambar 10** menunjukkan luaran yang dihasilkan pada halaman validasi.

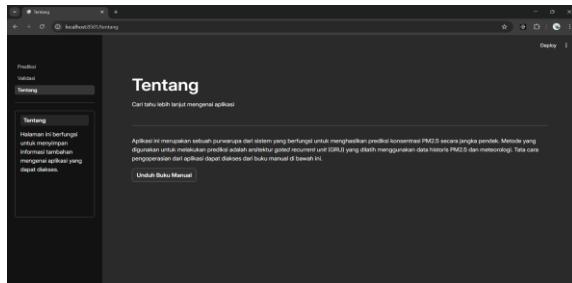


Gambar 9 Hasil Realisasi Halaman Validasi



Gambar 10 Luaran Halaman Validasi

Realisasi halaman terakhir merupakan halaman tentang. Halaman ini merupakan perpaduan antara halaman *information* dan halaman *about*. Halaman akan menampilkan informasi singkat seputar antarmuka, lalu alih-alih memaparkan langsung panduan penggunaan pada halaman tersebut, sekarang panduan berbentuk buku dan tidak dicantumkan langsung pada halaman. Jika pengguna ingin membaca buku panduan tersebut, maka pengguna dapat mengunduhnya pada halaman tentang melalui tombol "Unduh Buku Manual". **Gambar 11** menunjukkan tampilan halaman tentang.



Gambar 11 Hasil Realisasi Halaman Tentang

Pengujian pada antarmuka dilakukan menggunakan dua metode, yaitu *black box testing*. *Black box testing* adalah metode pengujian di mana fungsi pada antarmuka tersebut diuji hanya berdasarkan pada masukan, luaran, dan apa yang terlihat dari antarmuka tersebut tanpa mengetahui mengenai struktur dan program internalnya. Pola pikir pada metode ini menganggap bahwa sistem internal yang menjalani protokol sebuah rancangan sebagai sebuah kotak hitam yang tidak akan dilihat oleh pengguna, sehingga tidak dianggap penting selama fungsionalitas perangkat lunak berjalan dengan lancar [9]. *Black box testing* dilakukan dengan membuat tabel

yang berisi skenario pada antarmuka yang ingin diuji, diikuti dengan status apakah skenario tersebut berhasil atau gagal ketika dioperasikan. *Black box testing* pada antarmuka ini terdiri dari 12 skenario yang terlihat pada **Tabel 1**.

Tabel 1 Skenario Black Box Testing

No	Skenario	Berhasil	Gagal
1	Aplikasi dapat menampilkan halaman pertama sesuai rancangan, yaitu halaman prediksi.	Y	
2	Pada halaman prediksi, aplikasi dapat menerima masukan dari pengguna.	Y	
3	Pada halaman prediksi, setelah memberikan masukan, pengguna dapat menekan tombol "Prediksi" dan menampilkan luaran.	Y	
4	Luaran pada halaman prediksi berupa PM2.5 saat ini, prediksi PM2.5 beserta kategorinya, dan himbauan kepada pengguna.	Y	
5	Pengguna dapat beralih ke halaman validasi melalui kolom navigasi.	Y	
6	Aplikasi dapat menampilkan halaman kedua sesuai rancangan, yaitu halaman validasi.	Y	
7	Pada halaman validasi, aplikasi dapat menerima masukan dari pengguna berupa pilihan tanggal dan waktu.	Y	
8	Pada halaman validasi, setelah memberikan masukan, pengguna dapat menekan tombol "Prediksi" dan menampilkan luaran.	Y	
9	Luaran pada halaman validasi berupa PM2.5 saat ini, prediksi untuk PM2.5, nilai sebenarnya dari prediksi tersebut, persentase kesalahan prediksi, kategori PM2.5, serta grafik PM2.5 historis.	Y	
10	Pengguna dapat beralih ke halaman tentang melalui kolom navigasi.	Y	
11	Aplikasi dapat menampilkan halaman ketiga sesuai rancangan, yaitu halaman tentang.	Y	
12	Pada halaman tentang, pengguna dapat mengunduh buku manual melalui tombol "Unduh Buku Manual".	Y	

4. Kesimpulan

Penelitian ini telah berhasil merancang antarmuka untuk aplikasi prediksi PM2.5 dengan tujuan agar dapat diterima dan digunakan oleh masyarakat umum. Proses perancangan antarmuka ini menerapkan metode *software development life cycle* dengan model *waterfall*. Tahapan perancangan desain dilakukan menggunakan perangkat lunak Figma, sementara implementasi atau realisasinya dikembangkan menggunakan *framework* Streamlit dengan bahasa pemrograman Python.

Desain awal antarmuka direncanakan memiliki empat halaman: (i) *dashboard*; (ii) *information*; (iii) *evaluation*; dan (iv) *about*. Namun, dalam realisasinya, terjadi perubahan struktur dan fungsionalitas yang menghasilkan tiga halaman utama, yaitu: (i) halaman prediksi; (ii) halaman validasi; dan (iii) halaman tentang. Halaman prediksi merupakan pengembangan dari halaman *dashboard*, dengan perubahan input dari berbasis waktu menjadi berbasis fitur saat ini serta penambahan fitur himbauan. Halaman validasi mengkombinasikan fungsi dari halaman *evaluation* dan fitur data historis, yang memungkinkan pengguna membandingkan prediksi dengan data aktual dan melihat metrik evaluasi. Terakhir, halaman Tentang merupakan penggabungan dari halaman *information* dan *about*, di mana panduan penggunaan disajikan dalam bentuk buku manual yang dapat diunduh. Pengujian antarmuka dilakukan oleh pengembang menggunakan metode *black box testing*, yang memberikan hasil bahwa semua fitur pada antarmuka dapat dijalankan sesuai dengan ekspektasi. Antarmuka yang telah direalisasikan ini menyajikan fungsionalitas inti untuk memprediksi dan memvalidasi kadar PM2.5 sebagai upaya untuk membantu masyarakat dalam menghadapi masalah polusi udara.

REFERENSI

- [1] H. E. Institute, “State of global air 2024,” Health Effects Institute, Boston, MA, 2024, [Diakses 28 September 2025]. [Daring].
<https://www.stateofglobalair.org/resources/report/state-global-air-report-2024>
- [2] IQAir, “2024 world air quality report,” IQAir, 2025, [Diakses 29 September 2025]. [Daring].
<https://www.iqair.com/world-air-quality-report>
- [3] W. H. Organization, “Who global air quality guidelines: particulate matter (pm2.5 and pm10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide,” World Health Organization, 2021, [Diakses 29 September 2025]. [Daring].
<https://www.who.int/publications/i/item/9789240034228>
- [4] J. Kezia Halim, D. Erny Herwindiati, dan J. Hendryli, “Penerapan gated recurrent unit untuk prediksi zat pencemar udara,” *JIKSI: Jurnal Ilmu Komputer dan Sistem Informasi*, 2022.
- [5] M. Ratchagit, “An improved weight optimization of hybrid machine learning models for forecasting daily pm2.5 concentration,” *Contemporary Mathematics*, 2024.
- [6] T. Handhayani, “An integrated analysis of air pollution and meteorological conditions in jakarta,” *Scientific Reports*,

2023.

- [7] J. Ingino, *Software Architect’s Handbook: Become a successful software architect by implementing effective architecture concepts*. Packt Publishing Ltd., 2018.
- [8] Seidl, M., Scholz, M., Huemer, C., dan Kappel, G. *UML @ Classroom*. Springer, 2015.
- [9] P. C. Jorgensen, *Software Testing: A Craftsman’s Approach*. CRC Press, 2016.

Given Putra, saat ini sebagai mahasiswa Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Tarumanagara.

Tony, memperoleh gelar S.Kom. pada tahun 2005 dari Universitas Tarumanagara, M.Kom. pada tahun 2010 dari Universitas Indonesia, dan Ph.D. pada tahun 2021 dari Curtin University. Saat ini sebagai staf pengajar Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Tarumanagara.