

Perancangan User Interface Registrasi Mahasiswa Berbasis Model Fungsi Data Menggunakan Metode KSKT

Nurlina Yulis^{1*}, Andi Nasri²

¹Sistem Informasi, Universitas Dipa Makassar; Jl. Perintis Kemerdekaan KM. 09

²Teknik Informatika, Universitas Handayani Makassar

e-mail: ^{*}¹nurlinayulis@undipa.ac.id, ²andinasrimansur@handayani.ac.id.

Abstrak

Perancangan User Interface (UI) pada sistem informasi umumnya dilakukan berdasarkan kebutuhan pengguna dan pengalaman pengembang sehingga sering menghasilkan ketidakkonsistenan antara proses bisnis, struktur data, dan antarmuka aplikasi. Penelitian ini bertujuan menerapkan Model Fungsi Data (MFD) berbasis Metode Karakteristik Sistem Kompleks Terintegrasi (KSKT) sebagai dasar sistematis dalam perancangan User Interface Registrasi Mahasiswa. MFD dibentuk melalui transformasi hasil Analisis Struktur Sistem menjadi atribut, entitas, relasi, data master, dan data transaksi yang selanjutnya digunakan sebagai acuan penyusunan komponen UI dan UI Behavior. Penelitian menggunakan pendekatan Design Science Research dengan tahapan Analisis Struktur Sistem, penyusunan Model Fungsi Data, perancangan spesifikasi UI, pengembangan prototipe, serta verifikasi dan validasi fungsional melalui trial and error oleh programmer dan system analyst. Hasil implementasi menunjukkan bahwa seluruh aturan fungsional yang berasal dari Model Fungsi Data berhasil diterapkan pada prototipe, meliputi pembentukan NIM otomatis bagi mahasiswa baru, penampilan IPS dan IPK berdasarkan NIM, penyajian mata kuliah sesuai semester aktif, serta perhitungan otomatis total SKS dan batas SKS. Verifikasi terhadap pemetaan Model Fungsi Data, hubungan antarfield, UI Behavior, dan skenario fungsional menunjukkan tingkat keberhasilan implementasi sebesar 100%. Hasil penelitian menunjukkan bahwa Model Fungsi Data berbasis KSKT mampu menghasilkan perancangan User Interface yang konsisten dengan struktur data dan proses bisnis sistem.

Kata kunci—KSKT, User Interface Generation, Model Fungsi Data, Sistem Informasi Akademik, Analisis Sistem.

Abstract

User Interface (UI) design in information systems is generally developed based on user requirements and developers' experience, which often leads to inconsistencies among business processes, data structures, and application interfaces. This study aims to apply the Data Function Model (DFM), based on the Integrated Complex System Characteristics (ICSC) method, as a systematic foundation for designing the Student Registration User Interface. The DFM is constructed by transforming the results of System Structure Analysis into attributes, entities, relationships, master data, and transaction data, which are subsequently used as references for defining UI components and UI behavior. This study adopts a Design Science Research approach consisting of System Structure Analysis, Data Function Model development, UI specification design, prototype development, and functional verification and validation through trial-and-error testing conducted by programmers and system analysts. The implementation results show that all functional rules derived from the Data Function Model were successfully incorporated into the prototype, including automatic student identification number generation for new students, display of semester and cumulative grade point averages based on student identification numbers, presentation of courses according to the active semester, and automatic calculation of total credit

units and credit limits. Verification of the Data Function Model mapping, inter-field relationships, UI behavior, and functional scenarios demonstrated a 100% implementation success rate. The findings indicate that the KSKT-based Data Function Model can produce User Interface designs that are consistent with the system's data structure and business processes.

Keywords—KSKT, User Interface Generation, Data Function Model, Academic Information Systems, Systems Analysis.

1. PENDAHULUAN

Perancangan User Interface (UI) merupakan salah satu tahapan penting dalam pengembangan sistem informasi karena menjadi media interaksi antara pengguna dan sistem [1], [2]. Kualitas antarmuka tidak hanya ditentukan oleh aspek visual, tetapi juga oleh kesesuaian antara struktur data, proses bisnis, serta perilaku sistem yang direpresentasikan melalui komponen-komponen antarmuka [3], [4]. Dalam praktik pengembangan perangkat lunak, perancangan UI masih didominasi oleh pendekatan yang berorientasi pada kebutuhan pengguna (user-centered) atau pengalaman pengembang, sehingga struktur antarmuka sering kali dibangun setelah analisis sistem dan perancangan basis data selesai dilakukan [5], [6]. Kondisi tersebut berpotensi menimbulkan ke tidakkonsistenan antara model sistem, struktur data, dan implementasi antarmuka, terutama ketika terjadi perubahan kebutuhan sistem pada tahap pengembangan [3], [4].

Berbagai penelitian telah mengembangkan pendekatan perancangan antarmuka, seperti User-Centered Design (UCD), Model-Driven Architecture (MDA), Model-Driven Engineering (MDE), dan Low-Code Development Platform [1], [3] – [6]. Pendekatan-pendekatan tersebut memberikan kontribusi dalam meningkatkan kualitas antarmuka dan mempercepat proses pengembangan aplikasi [1] – [6]. Namun, sebagian besar penelitian masih memisahkan proses analisis sistem, pemodelan data, dan perancangan antarmuka sebagai tahapan yang berdiri sendiri [1] – [5]. Akibatnya, hubungan antara model data, proses bisnis, dan komponen antarmuka belum terbentuk secara sistematis sehingga perubahan pada struktur data sering memerlukan penyesuaian ulang terhadap rancangan antarmuka [1] – [5].

Metode Karakteristik Sistem Kompleks Terintegrasi (KSKT) telah dikembangkan melalui beberapa penelitian sebelumnya [7] – [11]. Penelitian mengenai semantic query tuning memperlihatkan bahwa proses inferensi dapat ditingkatkan melalui pemanfaatan hubungan semantik pada basis data [7]. Penelitian berikutnya mengembangkan Virtual Blockchain Database Tuning untuk menampilkan hasil inferensi secara real-time [8], sedangkan penelitian mengenai efisiensi basis data relasional menunjukkan bahwa penyetelan basis data berbasis algoritma mampu meningkatkan efisiensi pemrosesan data [9]. Penelitian terbaru juga telah menerapkan KSKT pada desain UI/UX untuk meningkatkan pengalaman pengguna [10], [11]. Meskipun demikian, penelitian-penelitian tersebut belum secara khusus menjelaskan bagaimana Model Fungsi Data dapat digunakan sebagai dasar sistematis dalam perancangan User Interface sehingga hubungan antara struktur data, proses bisnis, dan perilaku antarmuka dapat dipertahankan secara konsisten.

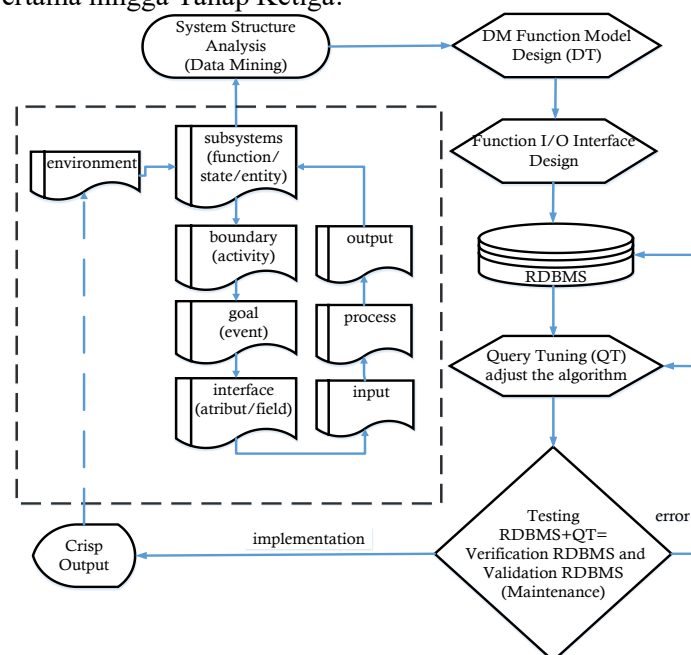
Metode KSKT merupakan suatu metodologi pengembangan sistem yang dilaksanakan secara bertahap dan terintegrasi. Tahap pertama menghasilkan Analisis Struktur Sistem melalui identifikasi delapan karakteristik sistem, yaitu Environment, Subsystems, Boundary, Goal, Interface, Input, Process, dan Output. Tahap kedua menghasilkan Model Fungsi Data sebagai representasi hubungan atribut, entitas, data master, dan data transaksi. Tahap ketiga memanfaatkan Model Fungsi Data sebagai dasar penyusunan struktur basis data yang mendukung implementasi sistem secara terintegrasi. Dalam penelitian ini, fokus diarahkan pada pemanfaatan Model Fungsi Data sebagai acuan dalam perancangan User Interface Registrasi Mahasiswa, sehingga setiap komponen antarmuka memiliki keterkaitan langsung dengan struktur data dan proses bisnis sistem.

Berbeda dengan penelitian sebelumnya yang umumnya memulai perancangan antarmuka dari kebutuhan pengguna atau rancangan visual, penelitian ini mengusulkan Model Fungsi Data sebagai dasar perancangan User Interface. Melalui pendekatan tersebut, setiap komponen antarmuka, hubungan antarfield, serta UI Behavior diturunkan secara sistematis dari struktur data yang telah dianalisis sebelumnya. Dengan demikian, Model Fungsi Data tidak hanya berfungsi sebagai representasi struktur data, tetapi juga menjadi mekanisme penghubung antara hasil Analisis Struktur Sistem dan rancangan antarmuka sehingga menghasilkan konsistensi antara proses bisnis, struktur data, dan implementasi aplikasi.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan menerapkan Model Fungsi Data berbasis Metode Karakteristik Sistem Kompleks Terintegrasi (KSKT) sebagai dasar sistematis dalam perancangan User Interface Registrasi Mahasiswa pada Sistem Informasi Akademik. Penelitian ini juga melakukan verifikasi dan validasi fungsional melalui proses trial and error oleh programmer dan system analyst terhadap seluruh aturan fungsional yang diturunkan dari Model Fungsi Data, meliputi pemetaan komponen antarmuka, hubungan antarfield, serta perilaku sistem pada proses registrasi mahasiswa baru maupun mahasiswa lama. Hasil penelitian diharapkan dapat menunjukkan bahwa pendekatan berbasis Model Fungsi Data mampu menghasilkan perancangan User Interface yang lebih konsisten dengan struktur data dan proses bisnis dibandingkan pendekatan yang hanya berorientasi pada rancangan visual.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan Design Science Research (DSR) untuk merancang User Interface Registrasi Mahasiswa berbasis Model Fungsi Data menggunakan metode Karakteristik Sistem Kompleks Terintegrasi (KSKT). Pendekatan ini dipilih karena penelitian tidak hanya menghasilkan artefak berupa prototipe antarmuka, tetapi juga menghasilkan prosedur sistematis yang menghubungkan Analisis Struktur Sistem, Model Fungsi Data, dan spesifikasi User Interface. Tahapan penelitian mengacu pada roadmap metode KSKT (Gambar 1), dengan fokus pada implementasi Model Fungsi Data sebagai dasar perancangan komponen antarmuka dan UI Behavior. Tahapan Analisis Struktur Sistem, Model Fungsi Data, dan implementasi basis data mengikuti kerangka metode KSKT sebagaimana dijelaskan pada dokumen invensi Tahap Pertama hingga Tahap Ketiga.



Gambar 1. Roadmap Algoritma KSKT

2.1 Tahapan Penelitian

Penelitian dilaksanakan melalui sembilan tahapan yang saling berkaitan, dimulai dari Analisis Struktur Sistem hingga verifikasi dan validasi prototipe. Tahapan tersebut disajikan pada Gambar 2.



Gambar 2. Flowchart Sistem Analisis UI

2.2 Analisis Struktur Sistem

Tahap pertama bertujuan mengidentifikasi karakteristik sistem menggunakan delapan karakteristik KSKT, yaitu Environment, Subsystems, Boundary, Goal, Interface, Input, Process, dan Output [13] – [16]. Hasil identifikasi digunakan untuk memperoleh kebutuhan sistem, batasan proses, tujuan operasional, serta hubungan antarentitas yang akan menjadi dasar penyusunan Model Fungsi Data. Tahapan ini tidak langsung menghasilkan rancangan antarmuka, tetapi menghasilkan representasi konseptual sistem yang menjadi acuan seluruh tahapan berikutnya.

2.3 Penyusunan Model Fungsi Data

Model Fungsi Data disusun berdasarkan hasil Analisis Struktur Sistem dengan mengelompokkan atribut yang memiliki karakteristik serupa menjadi Data Master, kemudian membentuk Data Transaksi melalui hubungan antarentitas. Setiap entitas memiliki atribut, relasi, serta aturan fungsional yang menjadi dasar pembentukan struktur data dan hubungan antarproses. Pada penelitian ini, Model Fungsi Data berfungsi sebagai sumber utama dalam menentukan komponen antarmuka, hubungan antarfield, serta perilaku sistem pada proses registrasi mahasiswa.

2.4 Perancangan User Interface

Berbeda dengan pendekatan konvensional yang memulai desain dari tata letak visual, penelitian ini menggunakan Model Fungsi Data sebagai dasar perancangan antarmuka. Setiap entitas pada Model Fungsi Data dipetakan menjadi komponen antarmuka, sedangkan relasi antarentitas digunakan untuk membentuk hubungan antarfield (field dependency) dan perilaku antarmuka (UI Behavior). Dengan demikian, rancangan antarmuka memiliki keterkaitan langsung dengan struktur data dan proses bisnis.

2.5 Perancangan UI Behavior



Gambar 3. Algoritma Design UI Behavior

UI Behavior merupakan sekumpulan aturan perilaku (functional rules) yang diturunkan dari Model Fungsi Data. Aturan tersebut mendefinisikan hubungan antarfield, validasi data, pemanggilan data, serta aktivasi komponen antarmuka Gambar 3. Contoh implementasi aturan tersebut meliputi:

1. pemilihan Program Studi memicu pembentukan NIM bagi mahasiswa baru;
2. input NIM pada mahasiswa lama menampilkan identitas mahasiswa, IPS, dan IPK;
3. semester aktif menampilkan daftar mata kuliah sesuai kurikulum;
4. pemilihan mata kuliah menghitung Total SKS dan memvalidasi Batas SKS;
5. pembayaran UKT mengaktifkan proses registrasi.

2.6 Prototype User Interface

Hasil pemetaan komponen antarmuka dan UI Behavior divisualisasikan menjadi prototipe Form Registrasi Mahasiswa. Prototype digunakan sebagai media implementasi untuk memverifikasi bahwa seluruh komponen antarmuka telah sesuai dengan Model Fungsi Data dan seluruh aturan fungsional dapat dijalankan secara konsisten.

2.7 Verifikasi and Validasi Fungsional

Verification merupakan proses pemeriksaan internal yang dilakukan oleh Programmer dan System Analyst melalui mekanisme trial and error selama proses pengembangan prototipe. Tujuan verification adalah memastikan bahwa seluruh komponen antarmuka, hubungan antarfield, dan aturan fungsional telah sesuai dengan Model Fungsi Data. Verification pada penelitian ini bertujuan mengevaluasi tingkat kesesuaian antara Model Fungsi Data, User Interface, dan aturan fungsional yang diimplementasikan pada prototipe. Proses verifikasi dilakukan menggunakan tiga indikator kuantitatif, yaitu Entity Mapping Accuracy (EMA), UI Behavior Accuracy (UBA), dan Functional Rule Verification (FRV). Ketiga indikator tersebut digunakan untuk mengukur kelengkapan pemetaan entitas, ketepatan implementasi perilaku antarmuka, serta kesesuaian aturan fungsional terhadap spesifikasi yang dirancang menggunakan metode KSKT.

2.7.1 Entity Mapping Accuracy (EMA)

Verification of Data Mapping, yaitu pemeriksaan kesesuaian antara entitas Model Fungsi Data dengan komponen User Interface. Entity Mapping Accuracy (EMA) digunakan untuk mengukur tingkat ketepatan pemetaan setiap entitas pada Model Fungsi Data menjadi komponen User Interface. Setiap entitas yang telah berhasil direpresentasikan dalam bentuk komponen antarmuka dihitung sebagai pemetaan yang benar (correct mapping). Indikator ini bertujuan memastikan bahwa seluruh entitas yang dirancang pada Model Fungsi Data memiliki representasi pada prototype sehingga tidak terdapat entitas yang hilang selama proses perancangan antarmuka.

Persamaan EMA dirumuskan sebagai berikut:

$$EMA = \frac{CE}{TE} \times 100\%$$

dengan:

EMA= *Entity Mapping Accuracy* (%)

CE (Correct Entity Mapping)= jumlah entitas yang berhasil dipetakan dengan benar menjadi komponen User Interface.

TE (Total Entity)= jumlah seluruh entitas pada Model Fungsi Data.

Semakin tinggi nilai EMA, semakin lengkap dan tepat representasi Model Fungsi Data pada rancangan antarmuka.

2.7.2 UI Behavior Accuracy (UBA)

UI Behavior Accuracy (UBA) digunakan untuk mengukur tingkat ketepatan implementasi perilaku antarmuka (UI Behavior) yang diturunkan dari hubungan antarentitas pada Model Fungsi Data. Setiap perilaku sistem yang berjalan sesuai spesifikasi, seperti aktivasi komponen, validasi data, pemanggilan informasi, maupun hubungan antarfield, dihitung sebagai perilaku yang benar (correct behavior).

Persamaan UBA dirumuskan sebagai berikut:

$$UBA = \frac{CB}{TB} \times 100\%$$

dengan:

UBA= *UI Behavior Accuracy* (%)

CB (Correct Behavior)= jumlah perilaku antarmuka yang berjalan sesuai spesifikasi.

TB (Total Behavior)= jumlah seluruh perilaku antarmuka yang dirancang.

Nilai UBA yang tinggi menunjukkan bahwa hubungan antarfield (*field dependency*) dan mekanisme respons sistem telah diimplementasikan secara konsisten sesuai Model Fungsi Data.

2.7.3 Functional Rule Verification (FRV)

Functional Rule Verification (FRV) digunakan untuk mengevaluasi tingkat kesesuaian implementasi seluruh aturan fungsional (functional rules) terhadap spesifikasi yang telah dirancang berdasarkan Model Fungsi Data. Berbeda dengan EMA yang berfokus pada pemetaan entitas dan UBA yang berfokus pada perilaku antarmuka, FRV mengukur apakah seluruh aturan proses bisnis berhasil dijalankan sesuai rancangan.

Persamaan FRV dirumuskan sebagai berikut:

$$FRV = \frac{CR}{TR} \times 100\%$$

dengan:

FRV= *Functional Rule Verification* (%)

CR (Correct Rules)= jumlah aturan fungsional yang berhasil diverifikasi sesuai spesifikasi.

TR (Total Rules)= jumlah seluruh aturan fungsional yang dirancang.

Nilai FRV menunjukkan tingkat kesesuaian implementasi aturan sistem terhadap spesifikasi yang dihasilkan oleh metode KSKT.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Analisis Struktur Sistem

Hasil analisis Struktur Sistem pada Gambar 3 menunjukkan bahwa Finance Office (FO) dan BAAK berfungsi sebagai dua subsistem utama yang menghubungkan proses administrasi keuangan dengan administrasi akademik. Pada registrasi mahasiswa baru, FO berperan sebagai tujuan akhir proses administrasi sebelum status mahasiswa dinyatakan aktif. Sebaliknya, pada mahasiswa lama, BAAK menjadi pusat pengendalian registrasi akademik setelah memperoleh konfirmasi pembayaran UKT dari bagian keuangan. Integrasi kedua subsistem tersebut membentuk alur informasi yang konsisten mulai dari environment, input, boundary/process, subsystem, hingga output (goal).

Pendekatan ini menunjukkan bahwa Model Fungsi Data berbasis KSKT tidak hanya mengidentifikasi entitas dan atribut, tetapi juga mendefinisikan hubungan fungsional antarproses sehingga setiap perubahan status pada satu subsistem secara otomatis memengaruhi subsistem lainnya. Dengan demikian, rancangan User Interface Registrasi Mahasiswa yang dihasilkan tidak disusun berdasarkan tata letak visual semata, tetapi diturunkan secara sistematis dari hubungan antarentitas, data master, data transaksi, dan aturan proses yang telah dimodelkan sebelumnya. Hal ini menghasilkan konsistensi antara proses bisnis, struktur data, dan perilaku antarmuka (*UI Behavior*) pada Sistem Informasi Akademik.

3.1.1 Struktur Registrasi Mahasiswa Baru menuju Finance Office (FO)

Gambar 3 menunjukkan bahwa proses registrasi mahasiswa baru diawali dari environment, yaitu lingkungan eksternal yang terdiri atas LLDIKTI, Rektorat, Program Studi, serta sumber informasi akademik lainnya. Informasi tersebut menjadi input bagi subsistem internal yang terdiri atas BAAK, Program Studi, Dosen, Keuangan (Keu), Tata Usaha (TU), dan Finance Office (FO). Masing-masing subsistem memiliki data master dan atribut yang saling berhubungan sehingga seluruh proses administrasi dapat dilakukan secara terintegrasi.

Pada bagian boundary/process, proses registrasi mahasiswa baru dilaksanakan melalui lima tahapan utama, yaitu Pendaftaran Mahasiswa Baru, Verifikasi Data, Pembayaran Formulir, Pengumuman Hasil Seleksi, dan Registrasi Ulang ke Finance Office (FO). Tahap pendaftaran mencakup pengisian identitas calon mahasiswa, pemilihan program studi, dan pengunggahan dokumen persyaratan. Selanjutnya, sistem melakukan verifikasi berkas, validasi data, serta pemeriksaan kuota penerimaan sesuai aturan yang telah ditentukan.

Tahap berikutnya adalah pembayaran formulir pendaftaran yang menjadi syarat untuk mengikuti proses seleksi. Setelah pembayaran berhasil diverifikasi, sistem mengumumkan hasil seleksi berupa status lulus atau tidak lulus. Bagi calon mahasiswa yang dinyatakan lulus, proses dilanjutkan dengan registrasi ulang melalui Finance Office (FO). Pada tahap ini mahasiswa menyetujui pernyataan registrasi, melengkapi administrasi, serta mencetak bukti registrasi sebagai dokumen resmi. Output akhir dari proses ini adalah status mahasiswa baru berhasil registrasi, yang selanjutnya menjadi dasar pembentukan data akademik pada Sistem Informasi Akademik.

Dari sudut pandang Model Fungsi Data, seluruh tahapan tersebut saling terhubung melalui relasi antara data calon mahasiswa, data program studi, data pembayaran, dan data registrasi. Oleh karena itu, setiap perubahan status pada satu proses, misalnya verifikasi pembayaran, secara otomatis memengaruhi proses berikutnya hingga menghasilkan status registrasi mahasiswa baru.

3.1.2 Struktur Pembayaran UKT Mahasiswa Lama menuju BAAK

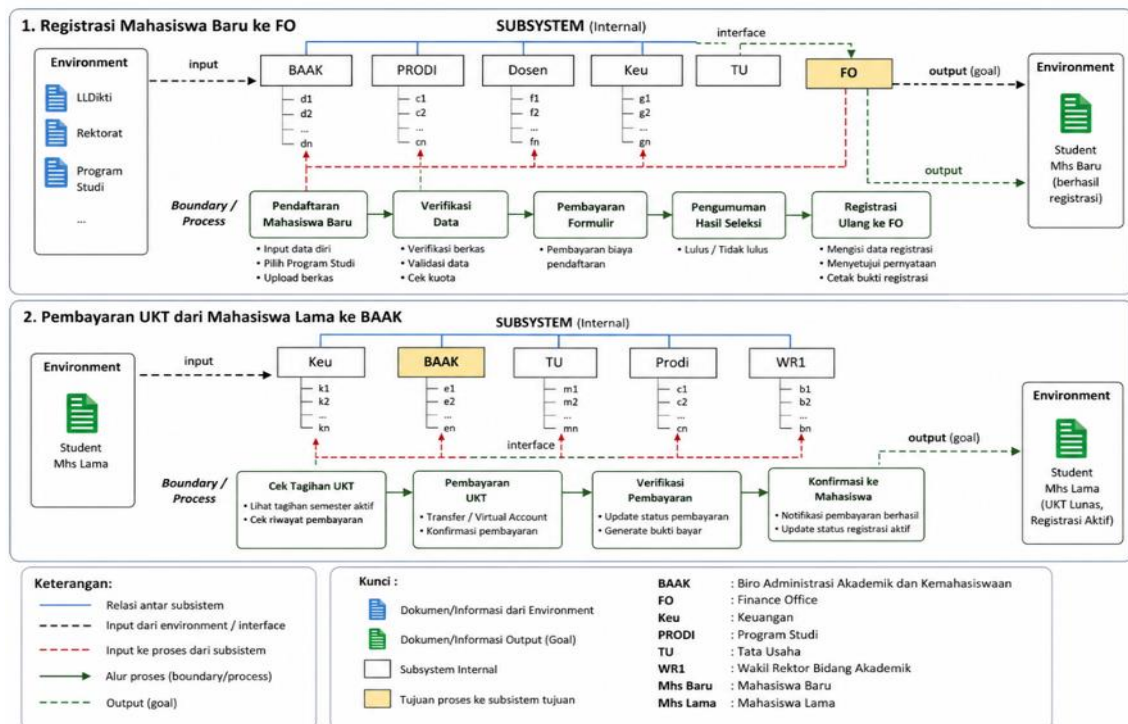
Bagian kedua Gambar 4 memperlihatkan struktur proses pembayaran Uang Kuliah Tunggal (UKT) yang dilakukan oleh mahasiswa lama. Berbeda dengan proses registrasi mahasiswa baru, proses ini dimulai dari environment berupa mahasiswa yang telah memiliki

identitas akademik (NIM). Mahasiswa mengakses sistem untuk memperoleh informasi mengenai tagihan UKT yang harus diselesaikan sebelum registrasi akademik dapat dilakukan.

Pada subsistem internal, proses pembayaran melibatkan beberapa unit, yaitu Keuangan (Keu), BAAK, Tata Usaha (TU), Program Studi, dan Wakil Rektor I (WR1). Tahapan proses dimulai dari cek tagihan UKT, yang menampilkan tagihan semester aktif beserta riwayat pembayaran sebelumnya. Setelah memperoleh informasi tersebut, mahasiswa melakukan pembayaran melalui bank atau *virtual account*. Sistem kemudian melakukan konfirmasi pembayaran dan mencatat seluruh informasi transaksi sebagai bagian dari data transaksi akademik.

Tahap berikutnya adalah verifikasi pembayaran yang dilakukan oleh bagian keuangan. Pada tahap ini sistem memperbarui status pembayaran, menghasilkan bukti pembayaran, serta mengirimkan informasi pembayaran kepada BAAK. Setelah pembayaran dinyatakan valid, sistem melakukan konfirmasi kepada mahasiswa melalui perubahan status registrasi menjadi aktif. Dengan demikian mahasiswa dapat melanjutkan proses registrasi akademik, termasuk pengisian Kartu Rencana Studi (KRS) sesuai semester yang sedang berjalan.

Model Fungsi Data menunjukkan bahwa proses pembayaran UKT tidak berdiri sendiri, tetapi terintegrasi dengan data mahasiswa, data pembayaran, data registrasi, dan data akademik. Oleh karena itu, keberhasilan pembayaran secara otomatis menjadi pemicu (*trigger*) untuk mengaktifkan layanan akademik berikutnya. Hubungan tersebut menjamin bahwa mahasiswa hanya dapat melakukan registrasi apabila seluruh persyaratan administrasi telah dipenuhi sesuai aturan sistem.



Gambar 4. Struktur Proses Registrasi Mahasiswa Baru melalui Finance Office (FO) dan Pembayaran UKT Mahasiswa Lama melalui BAAK Berdasarkan Metode KSKT

3.2 Hasil Model Fungsi Data

Hasil pemodelan menunjukkan bahwa ketiga subsistem, yaitu Student, Finance Office (FO), dan BAAK, tidak berdiri sendiri, tetapi saling terintegrasi melalui Model Fungsi Data. Setiap subsistem memiliki Data Master sebagai sumber referensi, Data Transaksi sebagai representasi aktivitas operasional, serta Interface, Input, Process, dan Output sebagai mekanisme interaksi sistem. Hubungan tersebut memungkinkan setiap perubahan data pada satu subsistem,

seperti konfirmasi pembayaran UKT oleh FO, secara otomatis memengaruhi status registrasi pada BAAK dan informasi akademik yang ditampilkan kepada mahasiswa.

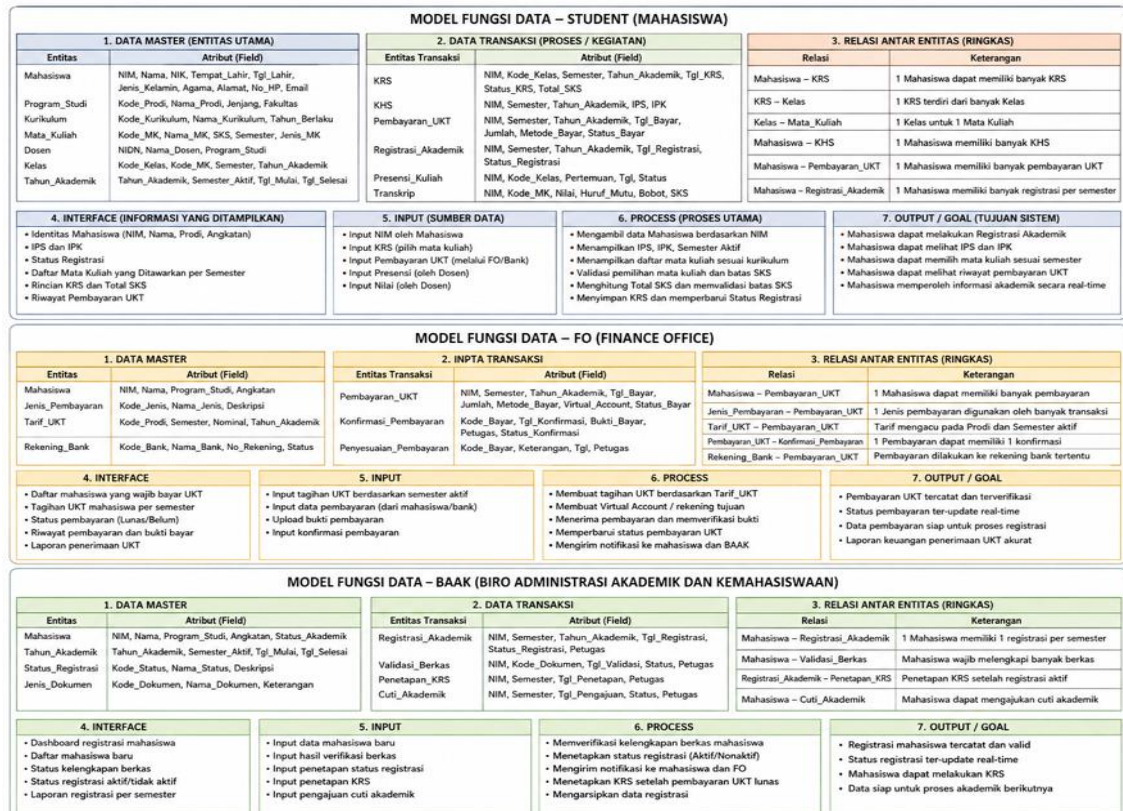
Berbeda dengan pendekatan konvensional yang umumnya merancang antarmuka berdasarkan kebutuhan visual atau pengalaman pengembang, penelitian ini menunjukkan bahwa Model Fungsi Data menjadi acuan utama dalam perancangan User Interface. Seluruh komponen antarmuka, hubungan antarfield (*field dependency*), serta *UI Behavior* diturunkan secara sistematis dari struktur data yang telah dianalisis. Dengan demikian, konsistensi antara proses bisnis, struktur data, dan antarmuka sistem dapat dipertahankan selama proses implementasi maupun pengembangan lebih lanjut.

3.2.1 Model Fungsi Data Mahasiswa (Student)

Model Fungsi Data pada Student (Mahasiswa) pada gambar 5 menggambarkan bahwa mahasiswa merupakan entitas utama yang berinteraksi dengan Sistem Informasi Akademik melalui proses registrasi akademik, pengisian Kartu Rencana Studi (KRS), pemantauan Kartu Hasil Studi (KHS), pembayaran UKT, serta kegiatan akademik lainnya. Pada model ini, Data Master terdiri atas entitas *Mahasiswa*, *Program Studi*, *Kurikulum*, *Mata Kuliah*, *Dosen*, *Kelas*, dan *Tahun Akademik*. Masing-masing entitas memiliki atribut yang saling berhubungan, seperti NIM, nama mahasiswa, program studi, kode mata kuliah, jumlah SKS, semester, serta identitas dosen yang menjadi dasar pembentukan informasi akademik.

Selanjutnya, Data Transaksi merepresentasikan aktivitas mahasiswa yang berubah setiap semester, meliputi registrasi akademik, pengisian KRS, hasil studi (KHS), pembayaran UKT, presensi perkuliahan, dan transkrip nilai. Data transaksi ini dihubungkan langsung dengan Data Master sehingga setiap perubahan aktivitas mahasiswa selalu mengacu pada identitas akademik yang telah tersimpan sebelumnya.

Pada sisi Interface, mahasiswa memperoleh informasi berupa identitas akademik, IPS, IPK, status registrasi, daftar mata kuliah sesuai semester aktif, riwayat pembayaran UKT, dan total SKS yang diambil. Seluruh informasi tersebut ditampilkan secara otomatis berdasarkan NIM sehingga mahasiswa tidak perlu melakukan pengisian data berulang. Selanjutnya, proses sistem melakukan validasi pemilihan mata kuliah, menghitung total SKS, menentukan batas maksimum SKS berdasarkan IPS, serta memperbarui status registrasi mahasiswa secara otomatis. Dengan demikian, Model Fungsi Data Mahasiswa menjadi dasar dalam menghasilkan *User Interface* yang konsisten dengan struktur data akademik dan proses registrasi.



Gambar 5. Struktur Model Fungsi Data pada Proses Registrasi Mahasiswa Baru melalui Finance Office (FO) dan Pembayaran UKT Mahasiswa Lama melalui BAAK Berdasarkan Metode KSKT

3.2.3 Model Fungsi Data BAAK (Biro Administrasi Akademik dan Kemahasiswaan)

Model Fungsi Data pada BAAK merupakan pusat pengelolaan administrasi akademik yang mengintegrasikan data mahasiswa, data registrasi, serta layanan akademik lainnya. Data Master pada subsistem ini meliputi data mahasiswa, tahun akademik, status registrasi, serta jenis dokumen akademik yang diperlukan selama proses administrasi. Data tersebut menjadi acuan dalam penyelenggaraan registrasi akademik setiap semester.

Data transaksi yang dikelola BAAK meliputi registrasi akademik, validasi berkas, penetapan KRS, serta pengelolaan cuti akademik. Seluruh transaksi tersebut selalu dikaitkan dengan identitas mahasiswa, semester akademik, dan status registrasi sehingga setiap perubahan dapat dilacak berdasarkan riwayat administrasi mahasiswa.

Melalui antarmuka BAAK, administrator memperoleh informasi mengenai daftar mahasiswa baru, status kelengkapan berkas, status registrasi aktif, hasil verifikasi pembayaran dari FO, serta laporan registrasi akademik setiap semester. Setelah memperoleh konfirmasi bahwa pembayaran UKT telah diverifikasi oleh FO, sistem secara otomatis mengubah status registrasi mahasiswa menjadi aktif dan mengizinkan mahasiswa melakukan pengisian KRS sesuai semester berjalan. Dengan demikian, Model Fungsi Data BAAK berperan sebagai pusat koordinasi administrasi akademik yang memastikan keterpaduan antara proses registrasi, pembayaran, dan layanan akademik lainnya.

3.2.2 Model Fungsi Data Finance Office (FO)

Model Fungsi Data pada Finance Office (FO) berfungsi sebagai subsistem yang mengelola seluruh proses administrasi keuangan mahasiswa, khususnya pembayaran Uang Kuliah Tunggal (UKT). Pada subsistem ini, Data Master terdiri atas data mahasiswa, jenis pembayaran, tarif UKT, dan rekening bank yang menjadi referensi dalam proses pembayaran.

Seluruh data tersebut digunakan sebagai acuan dalam menentukan besaran biaya, metode pembayaran, dan status transaksi keuangan mahasiswa.

Data transaksi pada FO meliputi pembayaran UKT, konfirmasi pembayaran, serta penyesuaian pembayaran apabila terjadi perubahan nominal atau proses koreksi administrasi. Setiap transaksi menyimpan informasi mengenai tanggal pembayaran, metode pembayaran, bukti pembayaran, status pembayaran, dan petugas yang melakukan verifikasi. Dengan demikian, seluruh riwayat pembayaran dapat ditelusuri kembali berdasarkan identitas mahasiswa dan semester akademik.

Melalui antarmuka FO, petugas keuangan dapat melihat daftar mahasiswa yang wajib membayar UKT, memantau status pembayaran, melakukan konfirmasi pembayaran, serta menghasilkan laporan penerimaan UKT. Setelah pembayaran berhasil diverifikasi, sistem secara otomatis memperbarui status pembayaran mahasiswa dan mengirimkan informasi tersebut kepada BAAK sebagai dasar pengaktifan proses registrasi akademik. Dengan demikian, Model Fungsi Data FO tidak hanya mengelola transaksi keuangan, tetapi juga menjadi penghubung antara proses pembayaran dan proses registrasi akademik:

3.3 Hasil Perancangan User Interface

Gambar 6 menunjukkan hasil implementasi Prototype Form Registrasi Mahasiswa yang dirancang berdasarkan Model Fungsi Data menggunakan metode Karakteristik Sistem Kompleks Terintegrasi (KSKT). Struktur antarmuka dibagi menjadi enam kelompok komponen utama, yaitu (1) Identitas dan Informasi Akademik, (2) Informasi Akademik Otomatis, (3) Informasi Registrasi, (4) Daftar Mata Kuliah per Semester, (5) Informasi Pembayaran, dan (6) Tombol Aksi. Pembagian tersebut tidak ditentukan berdasarkan kebutuhan visual semata, tetapi mengikuti struktur entitas dan hubungan data yang telah dibentuk pada Model Fungsi Data sehingga setiap komponen antarmuka memiliki fungsi yang jelas dalam mendukung proses registrasi akademik.

Pada bagian Identitas dan Informasi Akademik, mahasiswa memasukkan Nomor Induk Mahasiswa (NIM) sebagai identitas utama. NIM berfungsi sebagai kunci utama (primary identifier) yang menghubungkan seluruh data akademik mahasiswa dengan basis data. Setelah NIM dimasukkan, sistem secara otomatis menampilkan IPS, IPK, nama mahasiswa, program studi, jenjang pendidikan, serta semester aktif tanpa memerlukan pengisian ulang oleh pengguna. Mekanisme ini menunjukkan bahwa hubungan antarfield dibentuk berdasarkan relasi yang terdapat pada Model Fungsi Data sehingga proses pengambilan data berlangsung secara otomatis dan konsisten.

Bagian Informasi Registrasi menampilkan data registrasi yang berkaitan dengan semester berjalan, meliputi tanggal registrasi, status registrasi, jumlah SKS yang diambil, dan batas maksimum SKS. Seluruh informasi tersebut diperoleh dari data akademik yang telah tersimpan pada sistem. Dengan demikian, sistem tidak hanya menampilkan informasi registrasi, tetapi juga melakukan sinkronisasi antara status akademik mahasiswa dengan aturan registrasi yang berlaku pada semester tersebut.

PROTOTYPE FORM REGISTRASI MAHASISWA
BERBASIS MODEL FUNGSI DATA – METODE KSKT

1. IDENTITAS & INFORMASI AKADEMIK

NIM: 2022101234 (Valid)

IPS: 3,62 (Otomatis dari KHS)

IPK: 3,58 (Otomatis dari KHS)

Nama Mahasiswa: Ahmad Ramadhan

Program Studi: Teknik Informatika

Jenjang: S1

Semester Aktif: Genap 2024/2025 (Semester 2)

2. INFORMASI AKADEMIK OTOMATIS

3. INFORMASI REGISTRASI

Tanggal Registrasi: (Tampilkan ketika mahasiswa baru akan mendaftar)

Status Registrasi: AKTIF

Total SKS Diambil: 20 (Otomatis dari database)

Batas SKS: 24 (Otomatis dari database)

4. DAFTAR MATA KULIAH / SEMESTER

No.	Kode MK	Nama Mata Kuliah	SKS	Kelas	Dosen Pengampu	Jadwal	Pili
1	IF121	Struktur Data	3	A	Dr. Andi S.T., M.Kom.	Senin, 08.00-09.40	☒
2	IF122	Basis Data	3	A	Dr. Budi S.T., M.T.	Selasa, 10.00-11.40	☒
3	IF123	Perancangan Web	3	B	Ir. Citra, M.Kom.	Rabu, 13.00-14.40	☐
4	IF124	Jaringan Komputer	3	A	Dr. Dedi S.T., M.Kom.	Kamis, 08.00-09.40	☐
5	IF125	Kecerdasan Buatan	3	A	Dr. Eka S.T., M.Kom.	Jumat, 10.00-11.40	☐
6	IF126	Rekayasa Perangkat Lunak	3	B	Ir. Fajar, M.Kom.	Jumat, 13.00-14.40	☐

Total SKS (Terpilih): 20

5. INFORMASI PEMBAYARAN

Jenis Pembayaran: UKT

Nominal: Rp. 5.000.000

Tanggal Bayar: (Tampilkan ketika mahasiswa lama sudah bayar UKT)

Metode Pembayaran: Transfer Bank

Status Pembayaran: BELUM LUNAS

No. Referensi: -

6. TOMBOL AKSI

Simpan Registrasi, Cetak KRS, Cetak KHS, Batalkan Registrasi

KETERANGAN KOMPONEN (UI)

- Identitas & Informasi Akademik
- IPS dan IPK (otomatis dari KHS saat NIM diinput)
- Informasi Registrasi
- Daftar Mata Kuliah / Semester (otomatis)
- Informasi Pembayaran
- Tombol Aksi

LEGENDA HUBUNGAN ANTARFIELD

- Relasi: NIM → Identitas Mahasiswa
- Relasi: Data Akademik → IPS, IPK
- Relasi: Matakuliah → Total SKS
- Relasi: Pembayaran → Status Registrasi
- Relasi: Aksi → Perubahan Data

CATATAN UI BEHAVIOR (OTOMATIS)

- IPS dan IPK otomatis terisi saat NIM diinput.
- Tanggal Registrasi terisi jika mahasiswa baru akan mendaftar.
- Tanggal Bayar terisi setelah mahasiswa lama sudah bayar UKT.
- Daftar mata kuliah muncul otomatis sesuai Program Studi & Semester Aktif.
- Total SKS Diambil dan Batas SKS otomatis terakumulasi dari database.

Prototype ini dibentuk berdasarkan Model Fungsi Data menggunakan Metode KSKT, sehingga struktur komponen, hubungan antarfield, dan perilaku antarmuka (UI Behavior) mengikuti karakteristik sistem dan proses bisnis registrasi mahasiswa.

Gambar 6. Prototype Form Registrasi Mahasiswa Berbasis Model Fungsi Data–Metode KSKT

Komponen Daftar Mata Kuliah per Semester memperlihatkan implementasi hubungan antara kurikulum, program studi, semester aktif, dan mata kuliah. Setelah sistem mengenali program studi dan semester mahasiswa, daftar mata kuliah yang sesuai akan ditampilkan secara otomatis. Mahasiswa cukup memberikan tanda centang pada mata kuliah yang akan diambil, sedangkan sistem secara otomatis menghitung Total SKS berdasarkan mata kuliah yang dipilih. Pendekatan ini mengurangi kesalahan pemilihan mata kuliah sekaligus memastikan bahwa mata kuliah yang ditampilkan selalu sesuai dengan kurikulum dan semester aktif mahasiswa.

Bagian Informasi Pembayaran mengintegrasikan proses administrasi keuangan dengan proses registrasi akademik. Prototype menampilkan jenis pembayaran, nominal UKT, tanggal pembayaran, metode pembayaran, status pembayaran, dan nomor referensi transaksi. Informasi tersebut diperoleh dari data pembayaran yang tersimpan pada basis data. Setelah pembayaran berhasil diverifikasi, status registrasi mahasiswa akan diperbarui sehingga mahasiswa dapat melanjutkan proses registrasi akademik sesuai aturan yang telah ditetapkan.

Pada bagian Aksi, sistem menyediakan fungsi untuk menyimpan registrasi, mencetak KRS, mencetak KHS, serta membatalkan registrasi. Keempat fungsi tersebut tidak berdiri sendiri, tetapi diaktifkan sesuai kondisi proses yang telah didefinisikan pada UI Behavior. Misalnya, tombol cetak KRS hanya dapat digunakan setelah proses registrasi berhasil disimpan, sedangkan pencetakan KHS bergantung pada ketersediaan data hasil studi pada basis data.

Hubungan antarfield yang ditunjukkan melalui garis penghubung pada prototype memperlihatkan bahwa setiap perubahan data pada suatu komponen akan memicu respons otomatis pada komponen lain. Sebagai contoh, input NIM akan memanggil data IPS dan IPK, program studi menentukan daftar mata kuliah yang ditampilkan, pemilihan mata kuliah memengaruhi total SKS, sedangkan status pembayaran menentukan status registrasi mahasiswa. Dengan demikian, UI Behavior tidak dibentuk secara manual untuk setiap komponen, tetapi diturunkan dari relasi antarentitas yang telah didefinisikan pada Model Fungsi Data.

Pada Tabel 1 hasil implementasi menunjukkan bahwa Model Fungsi Data berfungsi sebagai dasar sistematis dalam perancangan User Interface, bukan sekadar sebagai dokumentasi struktur data. Setiap entitas pada Model Fungsi Data dipetakan menjadi komponen antarmuka, setiap atribut direpresentasikan sebagai *field* masukan atau keluaran, sedangkan setiap relasi antarentitas diterjemahkan menjadi hubungan antarfield (*field dependency*) dan aturan perilaku antarmuka (*UI Behavior*). Pendekatan ini menghasilkan antarmuka yang konsisten dengan proses bisnis, karena perubahan pada data atau status proses akan langsung tercermin pada komponen antarmuka yang terkait.

Tabel 1. Aturan Fungsional Berdasarkan Model Fungsi Data

Rule	Kondisi (Condition)	Proses Sistem	Output
R1	No. KTP diinput	Validasi identitas	Data mahasiswa ditampilkan
R2	Mahasiswa Baru	Pilih Program Studi	Sistem membentuk NIM
R3	Pembayaran UKT berhasil	Aktivasi registrasi	Status Registrasi Aktif
R4	NIM tersedia	Ambil data akademik	Nama Mahasiswa tampil
R5	NIM tersedia	Ambil KHS terakhir	IPS tampil otomatis
R6	NIM tersedia	Hitung IPK	IPK tampil otomatis
R7	Semester Aktif tersedia	Ambil Kurikulum	Daftar Mata Kuliah muncul
R8	Mata Kuliah dicentang	Hitung SKS	Total SKS bertambah
R9	IPS diketahui	Hitung batas SKS	Batas SKS tampil

R10	Total SKS $\leq$ Batas SKS	Validasi	Registrasi dapat disimpan
R11	Total SKS $>$ Batas SKS	Validasi	Pesan melebihi batas SKS
R12	Registrasi berhasil	Simpan KRS	Data KRS tersimpan
R13	Registrasi selesai	Cetak	KRS dapat dicetak
R14	Semester dipilih	Ambil nilai	KHS ditampilkan
R15	User = Admin	Hak akses Admin	Seluruh menu aktif
R16	User = Mahasiswa	Hak akses User	Menu sesuai hak akses
R17	Tahun Masuk diketahui	Generate 2 digit awal NIM	Tahun angkatan terbentuk
R18	Jurusan dipilih	Generate digit jurusan	Kode jurusan terbentuk
R19	Nomor urut terakhir diketahui	Increment otomatis	3 digit akhir NIM terbentuk
R20	Seluruh validasi berhasil	Finalisasi	Registrasi selesai

Berbeda dengan pendekatan konvensional yang umumnya memulai perancangan antarmuka dari kebutuhan visual atau pengalaman pengembang, metode KSKT memulai proses dari Analisis Struktur Sistem, kemudian membentuk Model Fungsi Data, dan selanjutnya menerjemahkan model tersebut menjadi rancangan User Interface. Oleh karena itu, seluruh komponen antarmuka memiliki asal-usul yang dapat ditelusuri kembali ke entitas, atribut, dan relasi pada Model Fungsi Data. Pendekatan ini meningkatkan konsistensi antara struktur data, proses bisnis, dan implementasi antarmuka, sekaligus memudahkan proses verifikasi dan validasi selama pengembangan sistem.

3.4 Hasil Perancangan UI Behavior

Tabel 2 menunjukkan hasil pemetaan antara Model Fungsi Data, komponen User Interface, dan UI Behavior yang dihasilkan menggunakan metode KSKT. Berbeda dengan pendekatan perancangan antarmuka secara konvensional yang umumnya menentukan komponen berdasarkan kebutuhan visual atau pengalaman pengembang, penelitian ini memulai proses dari Model Fungsi Data sehingga setiap komponen antarmuka memiliki asal-usul yang dapat ditelusuri kembali ke entitas, atribut, dan hubungan data yang telah dianalisis sebelumnya. Dengan demikian, setiap komponen UI memiliki fungsi yang jelas serta keterkaitan langsung dengan proses bisnis sistem.

Hasil pemetaan memperlihatkan bahwa seluruh komponen antarmuka berhasil dibentuk berdasarkan struktur data yang terdapat pada Model Fungsi Data. Sebagai contoh, NIM berfungsi sebagai identitas utama (primary identifier) yang menjadi penghubung seluruh informasi akademik mahasiswa. Setelah NIM dimasukkan, sistem secara otomatis menampilkan nama mahasiswa, Program Studi, IPS, IPK, serta semester aktif melalui mekanisme hubungan antarfield (field dependency) yang telah didefinisikan sebelumnya. Proses tersebut menunjukkan bahwa pengambilan data tidak dilakukan secara manual, melainkan mengikuti hubungan fungsional antarentitas pada Model Fungsi Data.

Selain itu, hubungan antara Program Studi, Semester Aktif, dan Kurikulum menghasilkan mekanisme otomatis dalam menampilkan daftar mata kuliah yang sesuai dengan semester berjalan. Mahasiswa hanya perlu memilih mata kuliah yang akan diambil, sedangkan sistem secara otomatis menghitung Total SKS serta menentukan Batas SKS berdasarkan nilai IPS yang tersimpan pada basis data. Mekanisme ini menunjukkan bahwa perilaku antarmuka (UI Behavior) tidak dibentuk melalui pemrograman terpisah pada setiap komponen, tetapi merupakan hasil transformasi hubungan antarentitas yang telah dirancang pada Model Fungsi Data.

Tabel 2. Hasil pemetaan antara Model Fungsi Data, komponen User Interface, dan UI Behavior

No. Sub	Nama SubSistem	Asal dari Model Fungsi Data	Aturan Perilaku
01	No. KTP Mahasiswa	Entitas Mahasiswa	Wajib diisi sebagai identitas utama mahasiswa.
02	Tanggal Registrasi	Relasi Registrasi	Terisi otomatis ketika mahasiswa baru melakukan registrasi pertama. Tidak ditampilkan pada registrasi ulang mahasiswa lama.
03	Program Studi (Jurusan)	Entitas KRS	Dipilih pengguna. Menentukan kurikulum, semester aktif, pembentukan NIM, dan daftar mata kuliah.
04	NIM	Entitas Mahasiswa (Aturan Pembentukan NIM)	Terbentuk otomatis setelah identitas mahasiswa dan Program Studi tervalidasi. Menjadi kunci untuk menampilkan data akademik mahasiswa.
05	Nama Mahasiswa	Entitas Mahasiswa	Ditampilkan otomatis berdasarkan data mahasiswa setelah NIM terbentuk atau dipanggil.
06	IPS	Relasi RegSPP	Ditampilkan otomatis (read only) berdasarkan KHS terakhir setelah NIM tersedia.
07	IPK	Relasi RegSPP	Ditampilkan otomatis (read only) berdasarkan akumulasi nilai mahasiswa setelah NIM tersedia.
08	Semester Aktif	Relasi RegSPP	Ditampilkan otomatis sesuai kalender akademik dan status registrasi mahasiswa.
09	Daftar Mata Kuliah	Relasi RegSPP	Muncul otomatis berdasarkan Program Studi dan Semester Aktif. Mahasiswa cukup memberi tanda centang pada mata kuliah yang akan diambil.
10	Mata Kuliah Terpilih	Relasi RegSPP	Terbentuk dari hasil centang mahasiswa pada daftar mata kuliah.
11	Total SKS Diambil	auto terakumulasi	Dihitung otomatis berdasarkan jumlah SKS dari mata kuliah yang dipilih.
12	Batas SKS	auto terakumulasi	Ditampilkan otomatis berdasarkan IPS/IPK dan ketentuan akademik yang berlaku.
13	Jenis Pembayaran	Relasi RegSPP	Menampilkan jenis pembayaran akademik (UKT/SPP) sesuai status mahasiswa.
14	Nominal Pembayaran	Relasi RegSPP	Ditampilkan otomatis berdasarkan tarif UKT/SPP sesuai Program Studi dan kategori mahasiswa.
15	Tanggal Bayar	Relasi RegSPP	Ditampilkan otomatis setelah transaksi pembayaran UKT/SPP berhasil. Berlaku untuk registrasi ulang mahasiswa lama.
16	Status Pembayaran	Relasi RegSPP	Ditampilkan otomatis menjadi Lunas atau Belum Lunas berdasarkan data pembayaran.
17	Status Registrasi	Relasi RegSPP	Menjadi Aktif apabila pembayaran telah tervalidasi dan registrasi berhasil dilakukan.
18	Cetak KRS	Relasi RegSPP	Aktif hanya jika registrasi selesai dan mata kuliah telah dipilih.
19	Cetak KHS	Relasi RegSPP	Aktif apabila data hasil studi tersedia pada semester yang dipilih.
20	Simpan Registrasi	Relasi RegSPP	Menyimpan seluruh data registrasi, KRS, dan pembayaran ke dalam basis data.

Pada sisi administrasi, Model Fungsi Data juga menghubungkan proses registrasi dengan proses pembayaran. Status pembayaran yang telah diverifikasi akan memperbarui status registrasi mahasiswa secara otomatis sehingga fungsi Simpan Registrasi, Cetak KRS, dan layanan akademik lainnya hanya aktif apabila seluruh persyaratan administrasi telah dipenuhi. Pendekatan ini menghasilkan hubungan yang konsisten antara data akademik, data keuangan, dan proses registrasi sehingga seluruh komponen antarmuka mengikuti aturan proses bisnis yang sama.

Secara keseluruhan, hasil pemetaan Tabel 3 menunjukkan bahwa seluruh komponen antarmuka, hubungan antarfield, dan UI Behavior berhasil diturunkan dari Model Fungsi Data. Hasil verifikasi memperlihatkan bahwa seluruh aturan fungsional yang dirancang dapat diimplementasikan pada prototipe tanpa ditemukan ketidaksesuaian antara struktur data dan perilaku antarmuka. Temuan ini memperkuat bahwa Model Fungsi Data berbasis KSKT tidak hanya berfungsi sebagai representasi struktur data, tetapi juga sebagai dasar sistematis dalam perancangan User Interface yang konsisten, terintegrasi, dan sesuai dengan proses bisnis Sistem Informasi Akademik.

Tabel 3. Pemetaan Model Fungsi Data terhadap Komponen User Interface dan UI Behavior

No	Model Fungsi Data	Komponen User Interface	UI Behavior	Hasil Verifikasi
1	Mahasiswa	NIM	Menjadi identitas utama dan memanggil seluruh informasi akademik	Berhasil
2	Mahasiswa	Nama Mahasiswa	Ditampilkan otomatis setelah NIM dikenali	Berhasil
3	Akademik	IPS	Ditampilkan otomatis berdasarkan data KHS terakhir	Berhasil
4	Akademik	IPK	Ditampilkan otomatis berdasarkan riwayat akademik	Berhasil
5	Program Studi	Program Studi	Ditampilkan sesuai data mahasiswa	Berhasil
6	Semester Aktif	Semester	Menampilkan semester berjalan secara otomatis	Berhasil
7	Kurikulum	Daftar Mata Kuliah	Mata kuliah muncul sesuai Program Studi dan Semester Aktif	Berhasil
8	Mata Kuliah	Total SKS	Total SKS dihitung otomatis berdasarkan mata kuliah yang dipilih	Berhasil
9	Akademik	Batas SKS	Ditentukan otomatis berdasarkan IPS	Berhasil
10	Registrasi	Status Registrasi	Berubah menjadi aktif setelah seluruh proses registrasi selesai	Berhasil
11	Pembayaran	Status Pembayaran	Status berubah setelah pembayaran UKT tervalidasi	Berhasil
12	Pembayaran	Nomor Referensi	Dibentuk otomatis setelah transaksi berhasil	Berhasil
13	Registrasi	Tanggal Registrasi	Diisi otomatis pada registrasi mahasiswa baru	Berhasil
14	Pembayaran	Tanggal Pembayaran	Ditampilkan otomatis setelah pembayaran berhasil	Berhasil
15	Registrasi	Tombol Simpan	Aktif setelah seluruh data valid	Berhasil
16	Akademik	Cetak KRS	Aktif setelah registrasi berhasil disimpan	Berhasil
17	Akademik	Cetak KHS	Aktif apabila data KHS tersedia	Berhasil
18	Registrasi	Batalan Registrasi	Mengembalikan status registrasi sesuai aturan sistem	Berhasil

3.5 Verifikasi dan Validasi Fungsional

Tabel 4. Mapping Rule KSKT

Rule	Model Fungsi Data	Trigger	UI Behavior	Verification
R1	Mahasiswa	Input KTP	Validasi identitas	✓
R2	Program Studi	Jurusan dipilih	Generate NIM	✓
R3	Registrasi	Pembayaran	Registrasi Aktif	✓
R4	Akademik	Input NIM	IPS tampil	✓
R5	Akademik	Input NIM	IPK tampil	✓
R6	Kurikulum	Semester aktif	Daftar MK tampil	✓
R7	Mata Kuliah	Checklist	Total SKS dihitung	✓
R8	SKS	Total berubah	Validasi batas SKS	✓
R9	KRS	Simpan	Data tersimpan	✓
R10	Registrasi	Final	Cetak KRS	✓

Tabel 5. Hasil Evaluasi Entity Mapping Accuracy (EMA)

No	Entitas Model Fungsi Data	Komponen User Interface	Status
1	Mahasiswa	NIM	✓
2	Mahasiswa	Nama Mahasiswa	✓
3	Mahasiswa	Jenjang Pendidikan	✓
4	Program Studi	Program Studi	✓
5	Fakultas	Fakultas	✓
6	Tahun Akademik	Semester Aktif	✓
7	Registrasi	Status Registrasi	✓
8	Registrasi	Tanggal Registrasi	✓
9	Akademik	IPS	✓
10	Akademik	IPK	✓
11	Kurikulum	Mata Kuliah	✓
12	Mata Kuliah	Jumlah SKS	✓
13	Pembayaran	Jenis Pembayaran	✓
14	Pembayaran	Status Pembayaran	✓
15	Pembayaran	Nomor Referensi	✓

No	Entitas Model Fungsi Data	Komponen User Interface	Status
16	Pembayaran	Tanggal Pembayaran	✓
17	Administrator	User Login	✓
18	KRS	Total SKS	✓

Jumlah Entitas = 18

Berhasil = 18

Perhitungan EMA

$$EMA = \frac{18}{18} \times 100\% = 100\%$$

Tabel 6. Hasil Evaluasi UI Behavior Accuracy (UBA)

No	Trigger	UI Behavior	Status
1	Input NIM	Menampilkan Nama Mahasiswa	✓
2	Input NIM	Menampilkan Program Studi	✓
3	Input NIM	Menampilkan IPS	✓
4	Input NIM	Menampilkan IPK	✓
5	Input NIM	Menampilkan Semester Aktif	✓
6	Pilih Semester	Menampilkan Mata Kuliah	✓
7	Checklist Mata Kuliah	Menghitung Total SKS	✓
8	IPS berubah	Mengubah Batas SKS	✓
9	Pembayaran Berhasil	Status Registrasi Aktif	✓
10	Registrasi Berhasil	Aktifkan Cetak KRS	✓
11	Data KHS tersedia	Aktifkan Cetak KHS	✓
12	Klik Simpan	Simpan Registrasi	✓
13	Klik Batal	Batal Registrasi	✓
14	Mahasiswa Baru	Generate NIM	✓
15	Logout	Keluar Sistem	✓

Jumlah UI Behavior = 15

Berhasil = 15

Perhitungan UBA

$$UBA = \frac{15}{15} \times 100\% = 100\%$$

Tabel 7. Hasil Functional Rule Verification (FRV)

No	Functional Rule	Status
1	Generate NIM otomatis	✓
2	Validasi Jurusan	✓
3	Validasi Semester	✓
4	Menampilkan Identitas Mahasiswa	✓
5	Menampilkan IPS	✓
6	Menampilkan IPK	✓
7	Menampilkan Mata Kuliah	✓
8	Validasi Mata Kuliah	✓
9	Perhitungan Total SKS	✓
10	Perhitungan Batas SKS	✓
11	Validasi Pembayaran	✓
12	Status Registrasi	✓
13	Cetak KRS	✓
14	Cetak KHS	✓
15	Simpan Registrasi	✓
16	Batal Registrasi	✓
17	Update Database	✓
18	Validasi User	✓
19	Logout Sistem	✓
20	Riwayat Registrasi	✓

Jumlah Rule = 20

Berhasil = 20
Perhitungan FRV

$$FRV = \frac{20}{20} \times 100\% = 100\%$$

Validation merupakan proses pengujian fungsional terhadap prototipe menggunakan dua skenario penggunaan, yaitu Administrator dan Mahasiswa. Pada tahap ini, programmer dan system analyst menjalankan aplikasi dengan memainkan kedua peran tersebut untuk memastikan bahwa seluruh proses bisnis berjalan sesuai spesifikasi.

Skenario mahasiswa baru meliputi validasi pembentukan NIM otomatis setelah proses registrasi dan pembayaran selesai. Skenario mahasiswa lama meliputi validasi penampilan otomatis identitas mahasiswa, IPS, IPK, semester aktif, daftar mata kuliah, Total SKS, dan Batas SKS setelah NIM dimasukkan. Skenario administrator digunakan untuk memverifikasi proses pengelolaan data registrasi, data master, serta data transaksi.

Hasil verifikasi selanjutnya digunakan untuk mengevaluasi tingkat konsistensi antara Model Fungsi Data, komponen User Interface, dan UI Behavior.

Berdasarkan hasil evaluasi pada Tabel 5–7, diperoleh nilai Entity Mapping Accuracy (EMA) sebesar 100%, UI Behavior Accuracy (UBA) sebesar 100%, dan Functional Rule Verification (FRV) sebesar 100%. Hasil tersebut menunjukkan bahwa seluruh entitas pada Model Fungsi Data berhasil dipetakan menjadi komponen *User Interface*, seluruh *UI Behavior* berhasil diimplementasikan sesuai hubungan antarentitas, serta seluruh aturan fungsional yang dirancang dapat dijalankan pada prototype sesuai spesifikasi sistem.

Nilai EMA sebesar 100% menunjukkan bahwa tidak terdapat entitas pada Model Fungsi Data yang kehilangan representasi pada antarmuka pengguna. Seluruh entitas utama, mulai dari data mahasiswa, data akademik, kurikulum, registrasi, hingga pembayaran berhasil ditransformasikan menjadi komponen antarmuka yang sesuai. Hal ini menunjukkan bahwa proses Perancangan User Interface berhasil mempertahankan konsistensi antara struktur data dengan rancangan antarmuka.

Nilai UBA sebesar 100% menunjukkan bahwa seluruh perilaku antarmuka (*UI Behavior*) telah berjalan sesuai rancangan. Setiap aksi pengguna menghasilkan respons sistem yang sesuai, seperti penampilan otomatis identitas mahasiswa setelah NIM diinput, pemunculan daftar mata kuliah berdasarkan semester aktif, perhitungan otomatis Total SKS dan Batas SKS, serta perubahan status registrasi setelah pembayaran UKT berhasil diverifikasi. Temuan ini memperlihatkan bahwa hubungan antarfield (*field dependency*) dapat diimplementasikan secara konsisten berdasarkan Model Fungsi Data.

Sementara itu, FRV sebesar 100% menunjukkan bahwa seluruh aturan fungsional berhasil diverifikasi melalui skenario pengujian mahasiswa baru, mahasiswa lama, administrator, serta petugas akademik. Seluruh proses registrasi, pembayaran, validasi akademik, pengisian KRS, hingga pencetakan KHS berjalan sesuai spesifikasi tanpa ditemukan penyimpangan terhadap aturan proses bisnis. Dengan demikian, prototype yang dihasilkan telah memenuhi spesifikasi fungsional yang dirancang menggunakan metode KSKT.

Ketiga indikator evaluasi tersebut membentuk mekanisme verifikasi yang saling melengkapi. Entity Mapping Accuracy (EMA) mengevaluasi ketepatan transformasi Model Fungsi Data menjadi komponen *User Interface*. Setelah struktur antarmuka terbentuk, UI Behavior Accuracy (UBA) memastikan bahwa hubungan antarfield dan perilaku sistem telah diimplementasikan sesuai spesifikasi. Selanjutnya, Functional Rule Verification (FRV) memverifikasi bahwa seluruh aturan proses bisnis dapat dijalankan pada prototype. Dengan demikian, evaluasi dilakukan secara bertahap mulai dari struktur data, struktur antarmuka, perilaku antarmuka, hingga fungsi sistem, sehingga menghasilkan proses verifikasi yang menyeluruh terhadap implementasi metode KSKT.

KSKT Verification Framework (KVF) yang terdiri atas tiga lapisan evaluasi yaitu:

1. Entity Mapping Accuracy (EMA)→memverifikasi transformasi Model Fungsi Data→Komponen UI.
2. UI Behavior Accuracy (UBA)→memverifikasi transformasi Komponen UI→UI Behavior.

3. Functional Rule Verification (FRV)→memverifikasi UI Behavior→Implementasi Fungsional.

Tabel 8. Perbandingan Algoritma A dan Algoritma B

No	Komponen UI	Sumber Model Fungsi Data	Trigger	Respons Sistem
01	No. KTP	Mahasiswa	Input KTP	Validasi identitas mahasiswa
02	Program Studi	Program Studi	Dipilih	Membentuk NIM & Kurikulum
03	NIM	Mahasiswa	Terbentuk	Menampilkan IPS, IPK, Riwayat Akademik
04	Semester Aktif	Kalender Akademik	Semester dipilih	Menampilkan daftar mata kuliah
05	Mata Kuliah	Kurikulum	Dicentang	Menghitung Total SKS
06	Total SKS	KRS	Mata kuliah berubah	Total SKS diperbarui
07	Status Pembayaran	Pembayaran	UKT lunas	Registrasi menjadi Aktif
08	Registrasi	Registrasi	Simpan	KRS dapat dicetak

Tabel 9. Perbandingan Penelitian Sebelumnya

Penelitian Sebelumnya	Keterbatasan
UI/UX konvensional	Berorientasi pada layout dan usability
Model Driven UI	Belum menghubungkan karakteristik sistem secara menyeluruh
Penelitian KSKT Sebelumnya	Berfokus pada analisis sistem, query tuning, dan database tuning

Setiap komponen antarmuka memiliki tiga aspek sekaligus: yaitu asal-usulnya (Model Fungsi Data), pemicu (trigger) yang mengaktifkannya, dan respons sistem yang dihasilkan (Tabel 5). Dengan demikian, pada KSKT bersifat *behavior-driven*, yaitu setiap komponen tidak hanya memiliki posisi pada layar, tetapi juga mempunyai aturan perilaku yang berasal langsung dari Model Fungsi Data (Tabel 5). Pada Tabel 6 yang menjadi pembeda paling kuat antara pendekatan KSKT dan perancangan UI konvensional.

Meskipun berbagai penelitian telah membahas perancangan UI dan UX, sebagian besar pendekatan masih berorientasi pada aspek visual, kegunaan (*usability*), dan pengalaman pengguna tanpa menjelaskan bagaimana komponen antarmuka dirancang secara sistematis berdasarkan model sistem. Demikian pula, penelitian-penelitian KSKT sebelumnya lebih berfokus pada analisis struktur sistem, pemodelan fungsi data, optimasi query, dan penyietan basis data, sehingga penerapan KSKT dalam pembentukan antarmuka pengguna belum pernah dibahas secara khusus. Oleh karena itu, penelitian ini mengisi kesenjangan tersebut dengan mengusulkan Perancangan Struktur User Interface berbasis Model Fungsi Data menggunakan Metode Karakteristik Sistem Kompleks Terintegrasi (KSKT). Kontribusi utama penelitian ini adalah menunjukkan bahwa komponen antarmuka, hubungan antarfield, urutan aktivasi, serta perilaku antarmuka (*UI Behavior*) dapat dirancang secara sistematis berdasarkan karakteristik sistem dan Model Fungsi Data yang dihasilkan pada tahapan KSKT. Dengan demikian, antarmuka yang dihasilkan tidak hanya berfungsi sebagai media interaksi pengguna, tetapi juga merepresentasikan struktur sistem dan proses bisnis secara terpadu.

4. KESIMPULAN

Penelitian ini berhasil menerapkan Model Fungsi Data berbasis Metode Karakteristik Sistem Kompleks Terintegrasi (KSKT) sebagai dasar sistematis dalam perancangan User Interface Registrasi Mahasiswa pada Sistem Informasi Akademik. Berbeda dengan pendekatan konvensional yang umumnya memulai perancangan antarmuka dari kebutuhan pengguna atau rancangan visual, penelitian ini menunjukkan bahwa proses perancangan dapat dilakukan secara bertahap melalui Analisis Struktur Sistem, Model Fungsi Data, Perancangan User Interface, dan

Perancangan UI Behavior, sehingga setiap komponen antarmuka memiliki keterkaitan langsung dengan struktur data, hubungan antarentitas, dan proses bisnis sistem.

Hasil implementasi menunjukkan bahwa seluruh entitas pada Model Fungsi Data berhasil dipetakan menjadi komponen User Interface, seluruh UI Behavior berhasil diimplementasikan sesuai spesifikasi, dan seluruh aturan fungsional dapat dijalankan pada prototipe. Hal tersebut dibuktikan melalui evaluasi menggunakan Entity Mapping Accuracy (EMA), UI Behavior Accuracy (UBA), dan Functional Rule Verification (FRV) yang menunjukkan tingkat kesesuaian implementasi sebesar 100% terhadap spesifikasi yang dirancang. Hasil tersebut menunjukkan bahwa pendekatan berbasis Model Fungsi Data mampu menghasilkan rancangan antarmuka yang konsisten, terstruktur, dan sesuai dengan proses bisnis Sistem Informasi Akademik.

Kontribusi utama penelitian ini adalah memperkenalkan Model Fungsi Data sebagai dasar metodologis dalam perancangan User Interface, sehingga hubungan antara struktur data, komponen antarmuka, dan perilaku sistem dapat ditelusuri secara sistematis. Dengan demikian, penelitian ini tidak hanya menghasilkan sebuah prototipe Form Registrasi Mahasiswa, tetapi juga menawarkan suatu pendekatan perancangan yang dapat diterapkan pada pengembangan sistem informasi di berbagai domain yang memiliki karakteristik proses bisnis terstruktur.

Dengan demikian, penelitian ini memberikan dua kontribusi utama, yaitu (1) memperkenalkan kerangka sistematis perancangan User Interface berbasis Model Fungsi Data menggunakan metode KSKT, serta (2) mengusulkan kerangka evaluasi implementasi melalui indikator Entity Mapping Accuracy (EMA), UI Behavior Accuracy (UBA), dan Functional Rule Verification (FRV) untuk menilai konsistensi antara Model Fungsi Data, User Interface, dan aturan fungsional. Kedua kontribusi tersebut diharapkan dapat menjadi landasan bagi pengembangan metode perancangan sistem informasi yang lebih terstruktur dan dapat diterapkan pada berbagai domain aplikasi.

5. SARAN

Penelitian ini masih berfokus pada perancangan User Interface berdasarkan Model Fungsi Data dan belum mencakup proses pembangkitan kode antarmuka (automatic user interface generation) maupun implementasi lintas platform. Oleh karena itu, penelitian selanjutnya dapat diarahkan pada pengembangan mekanisme transformasi otomatis dari Model Fungsi Data menjadi kode antarmuka pada berbagai framework pengembangan aplikasi, seperti aplikasi berbasis web, perangkat bergerak (mobile), maupun desktop, dengan tetap mempertahankan konsistensi struktur data dan perilaku antarmuka.

Selain itu, evaluasi pada penelitian ini dilakukan menggunakan indikator Entity Mapping Accuracy (EMA), UI Behavior Accuracy (UBA), dan Functional Rule Verification (FRV) yang berfokus pada kesesuaian implementasi terhadap spesifikasi. Penelitian berikutnya dapat memperluas evaluasi dengan mengukur aspek kualitas perangkat lunak lainnya, seperti kemudahan pemeliharaan (maintainability), kemudahan pengembangan (extensibility), waktu pengembangan (development time), serta kepuasan pengguna (usability) melalui pengujian pada beberapa studi kasus dan domain aplikasi yang berbeda. Dengan demikian, generalisasi metode KSKT dalam perancangan User Interface dapat dievaluasi secara lebih komprehensif.

UCAPAN TERIMA KASIH

Alhamdulillah Rabbil ‘aalamiin atas anugrah dan karunia NYA diberi kesempatan dan kesehatan menyusun paper ini. Terimakasih kepada tim LPPM atas keaktifannya memotivasi kami memasukkan paper, semoga senantiasa diberi kesehatan dan terimakasih kepada Universitas Dipa Makassar, semoga jaya selalu.

DAFTAR PUSTAKA

- [1] Irmak, G., & Mahmoud, Q.H. (2026). "Design Behaviour and Interface Consistency in Generative No-Code Tools: A Systematic Literature Review." *Computers*, 15(4),238. MDPI.
 - [2] Redesigning Website Interface Using UI/UX and Design Thinking Approaches to Improve User Performance and Satisfaction. (2026). *Procedia Computer Science*, 284, 797–804.
 - [3] Fiqriansyah, G.R., Zikrian, F., Sasongko, J., & Albana, I. (2026). "Systematic Literature Review: Implementasi Metode User-Centered Design (UCD) Dalam Meningkatkan Kualitas dan Usability Desain Antarmuka Sistem Informasi." *Nusantara Journal of Multidisciplinary Science*, 3(9), 1414–1422.
 - [4] Kleppe, A., Warmer, J., & Bast, W. (2025). "Model Driven Architecture: Applying MDA to Enterprise Systems." *Software and Systems Modeling*, Springer.
 - [5] Bézivin, J., & Jouault, F. (2024). "On the Unification Power of Models." *Software and Systems Modeling*, 23(2), 311–329.
 - [6] Nopiyan, A. (2026). "UI/UX 2026: Integrating Predictive Interaction and Adaptive Design." *Journal of Human–Computer Interaction Advances*, 18(1), 47–53.
 - [7] N. Yulis, A. A. Ilham, A. Achmad, and F. A. Samman, "Query tuning in semantic inference fuzzy logic algorithm for real-time recommendations," in *Proc. 2023 Int. Conf. Networking, Electrical Engineering, Computer Science, and Technology (IConNECT)*, Aug. 2023, pp. 42–47, IEEE.
 - [8] N. Yulis, A. A. Ilham, A. Achmad, and F. A. Samman, "Virtual-blockchain method database-tuning: Displaying real-time aggregate-fire-strength for fuzzy logic database," *SSRN*, 2024. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4766631
 - [9] N. Yulis, A. A. Ilham, A. Achmad, and F. A. Samman, "Enhancing relational database efficiency through algorithmic query tuning in virtual memory systems," *JOIV: Int. J. on Informatics Visualization*, vol. 9, no. 4, pp. 1390–1400, 2025.
 - [10] N. Yulis, F. Faizal, M. A. Anhar, and A. S. Rombe, "Peningkatan pengalaman pengguna melalui desain UI/UX pada sistem prediksi kebutuhan bahan baku: Studi kasus Kafe Kanal Social Space," in *Proc. SISITI: Seminar Ilmiah Sistem Informasi dan Teknologi Informasi*, vol. 15, no. 1, pp. 41–49, Jan. 2026.
-