

Rancang Bangun Sistem Informasi Manajemen Berbasis KPI Untuk Monitoring dan Evaluasi Anggota Menggunakan Metode Prototyping (Studi Kasus: Pelayanan Pastoral Mahasiswa Keuskupan Surabaya)

1st Yohanes Mahardika Arya Putra
Cahyana

Sistem Informasi
Telkom University Surabaya
Surabaya, Indonesia

[yohanesmma@student.telkomuniversit
y.ac.id](mailto:yohanesmma@student.telkomuniversit
y.ac.id)

2nd Anfazul Faridatul Azizah, S.Kom.,
M.Kom.

Sistem Informasi
Telokom University Surabaya
Surabaya, Indonesia

anfazulazizah@telkomuniversity.ac.id

3rd Rosyid Abdillah, S.Si., M.Kom.
Sistem Informasi

Telkom University Surabaya
Surabaya, Indonesia

rosyidabdillah@telkomuniversity.ac.id

Pelayanan Pastoral Mahasiswa Keuskupan Surabaya (PPMKS) merupakan organisasi kerohanian mahasiswa yang menghadapi tantangan dalam manajemen internal, terutama dalam monitoring dan evaluasi kinerja anggota yang masih dilakukan secara manual. Dokumentasi dan evaluasi yang tidak teratur menyebabkan rendahnya efektivitas kegiatan, dengan tingkat kegagalan program kerja mencapai 55%. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi manajemen berbasis Key Performance Indicator (KPI) yang mampu meningkatkan akuntabilitas dan efisiensi organisasi. Metode yang digunakan adalah prototyping, yang memungkinkan iterasi desain dan pengujian secara berkelanjutan melalui keterlibatan langsung pengguna. Proses dimulai dari identifikasi kebutuhan, perancangan antarmuka (wireframe, UML), pembuatan prototipe, hingga pengujian dan analisis sistem. Sistem dikembangkan menggunakan Laravel dan mencakup fitur seperti pengelolaan kegiatan, tugas, evaluasi kinerja, pencatatan notulen, dan dashboard visualisasi KPI. Hasil pengujian usability menggunakan pendekatan Nielsen Attributes of Usability (NAU) menunjukkan tingkat kepuasan pengguna yang tinggi, dengan skor berkisar antara 4,5 hingga 4,8 pada lima atribut utama: learnability, efficiency, memorability, error, dan satisfaction. Uji black box juga memastikan bahwa seluruh fitur berfungsi sesuai dengan kebutuhan tiap peran pengguna. Sistem ini berhasil meningkatkan transparansi dan efektivitas pengelolaan anggota PPMKS. Platform ini dapat menjadi model awal pengembangan sistem informasi serupa bagi organisasi non-profit lain yang membutuhkan sistem monitoring dan evaluasi berbasis data.

Kata kunci— Sistem Informasi Manajemen, Monitoring, Evaluasi, KPI, Prototyping, Pelayanan Pastoral Mahasiswa

I. PENDAHULUAN

Kemajuan teknologi telah menggantikan banyak sistem manual dengan sistem digital, mempercepat akses informasi dan komunikasi [1]. Perubahan ini mendorong peningkatan efisiensi kerja dan fleksibilitas akses bagi pengguna. Studi juga menunjukkan bahwa Sistem Informasi Manajemen, komunikasi, dan motivasi kerja berpengaruh signifikan terhadap kinerja pegawai [2]. Maka, penerapan teknologi

menjadi aspek krusial dalam mendukung operasional di era digital.

Di era digital, pemanfaatan teknologi informasi menjadi strategi utama untuk mengoptimalkan proses organisasi [3]. Platform digital dan aplikasi web dipilih karena mampu mempermudah pengelolaan data secara efisien. Penerapan sistem seperti arsip digital berbasis web, sebagaimana digunakan oleh Biro ISDA Sekretariat Daerah Provinsi Jawa Tengah, terbukti efektif dalam mengurangi risiko kehilangan dokumen, mempercepat pencarian, dan menghemat ruang penyimpanan [4]. Teknologi ini turut mendorong peningkatan efisiensi, transparansi, dan akuntabilitas dalam operasional organisasi.

Pencapaian tujuan organisasi kini dapat dipercepat melalui teknologi dan alat ukur yang tepat. Integrasi Sistem Informasi Manajemen (SIM) dengan teknologi informasi terbukti meningkatkan efisiensi dan efektivitas kerja secara signifikan [5], serta memudahkan pengelolaan data secara cepat dan akurat [6]. Selain itu, penggunaan *Key Performance Indicator* (KPI) memungkinkan organisasi *non-profit* menilai kinerja dari berbagai aspek, termasuk kepuasan pelanggan, proses internal, dan pengembangan organisasi [7]. Pengelolaan yang terstruktur menjadi krusial, terutama bagi organisasi dengan banyak program dan kegiatan.

Organisasi Pelayanan Pastoral Mahasiswa Keuskupan Surabaya (PPMKS) merupakan lembaga kerohanian yang berperan dalam koordinasi, fasilitasi, dan pengembangan Keluarga Mahasiswa Katolik (KMK) di berbagai PTN dan PTS di wilayah Keuskupan Surabaya. Organisasi ini berada di bawah pendampingan Pastor Kapelan Mahasiswa yang bertindak atas otoritas ordinari setempat. Salah satu tantangan utama PPMKS adalah lemahnya pemantauan sumber daya manusia internal, karena proses *monitoring* masih terbatas pada rapat berkala.

PPMKS memiliki 36 anggota dari 18 kampus, dengan kegiatan seperti retreat dan misa yang dapat melibatkan lebih dari 80 mahasiswa dari 27 universitas. Namun, seluruh manajemen masih dilakukan secara manual, sehingga memunculkan berbagai kendala operasional. Berdasarkan evaluasi internal, 55% dari 56 program kerja yang direncanakan mengalami kegagalan. Wawancara dengan anggota mengungkapkan bahwa faktor utama kegagalan adalah kurangnya efektivitas rapat dan evaluasi, di mana kendala sering tidak tersampaikan sepenuhnya karena disampaikan secara lisan. Ketiadaan sistem digital juga menghambat evaluasi kinerja divisi dan anggota secara objektif.

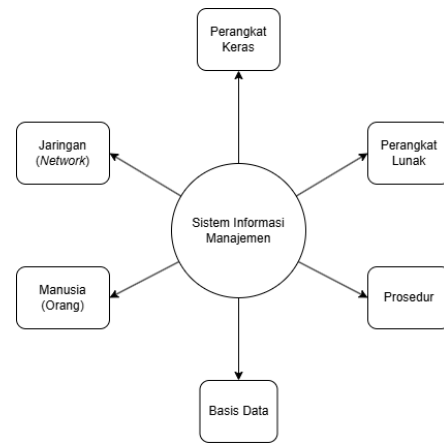
PPMKS belum memiliki KPI yang terdokumentasi secara formal untuk setiap aktivitas anggotanya, perancangan sistem manajemen berbasis KPI akan dimulai dari tahap dasar. Proses ini mencakup penyusunan KPI secara kolaboratif, selaras dengan tujuan organisasi dan ruang lingkup pelayanan pastoral mahasiswa. Dengan demikian, platform yang dikembangkan tidak hanya mengelola KPI yang sudah ada, tetapi juga berfungsi sebagai sarana untuk memformalkan dan mengintegrasikan indikator kinerja serta prosedur standar ke dalam operasional organisasi.

Dengan demikian, dibutuhkan suatu platform yang dapat memfasilitasi manajemen sumber daya manusia secara efektif, khususnya dalam hal pemantauan dan evaluasi anggota. Penerapan Sistem Informasi Manajemen berbasis KPI diharapkan mampu meningkatkan efisiensi operasional organisasi. Platform ini dikembangkan menggunakan Laravel, sebuah *framework* PHP yang mendukung pengembangan aplikasi web secara terstruktur dan efisien. Metode *prototyping* digunakan sebagai pendekatan pengembangan, karena memungkinkan proses iteratif yang melibatkan pengguna secara langsung dalam tiap siklus perancangan, evaluasi, dan perbaikan. Pendekatan ini memungkinkan penyesuaian sistem terhadap kebutuhan riil di lapangan melalui umpan balik berkelanjutan [8].

II. KAJIAN TEORI

A. Sistem Informasi Manajemen

Sistem Informasi Manajemen (SIM) merupakan sistem terintegrasi yang memadukan teknologi informasi, prosedur bisnis, dan peran manusia untuk mengelola informasi secara efektif guna mendukung pengambilan keputusan dan perencanaan strategis organisasi. SIM terdiri dari lima komponen utama, yaitu perangkat keras (server, komputer, media penyimpanan), perangkat lunak (aplikasi pengelola data), prosedur operasional (aturan dan kebijakan kerja), basis data (penyimpanan data terstruktur), serta unsur manusia yang berperan dalam analisis dan pengambilan keputusan [9]. Komponen utama Sistem Informasi Manajemen (SIM) mencakup perangkat keras (seperti server, komputer, dan media penyimpanan), perangkat lunak (aplikasi untuk pengumpulan dan analisis data), prosedur operasional (aturan dan langkah kerja organisasi), basis data (sistem penyimpanan terstruktur untuk data yang konsisten), serta unsur manusia yang berperan dalam pengelolaan, interpretasi, dan pengambilan keputusan berbasis informasi.

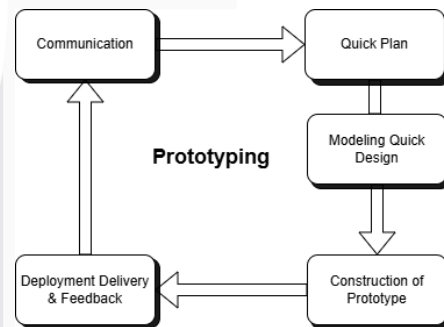


GAMBAR 1

(Komponen Sistem Informasi Manajemen)

B. Metode Prototyping

Metode *prototyping* merupakan pendekatan pengembangan sistem yang menekankan pembuatan versi awal produk (*prototype*) untuk menguji ide, fungsionalitas, dan interaksi sebelum sistem dikembangkan sepenuhnya. Terdapat dua tipe utama: *low-fidelity* (sketsa/wireframe sederhana) dan *high-fidelity* (desain interaktif mendekati produk akhir). Proses ini mencakup lima tahap: *communication* (pengembang dan pengguna merumuskan kebutuhan, seperti fitur pengumuman, jadwal, dan percakapan antaranggota), *quick plan* (penyusunan *use case* berdasarkan fungsi inti), *modeling quick design* (pembuatan *prototype* awal dan pengujian dasar), *construction of prototype* (evaluasi *prototype* dengan pengguna), serta *deployment delivery & feedback* (diskusi umpan balik untuk perbaikan). Metode ini mendukung desain iteratif berbasis umpan balik cepat sehingga risiko pengembangan dapat diminimalkan [10].



GAMBAR 2

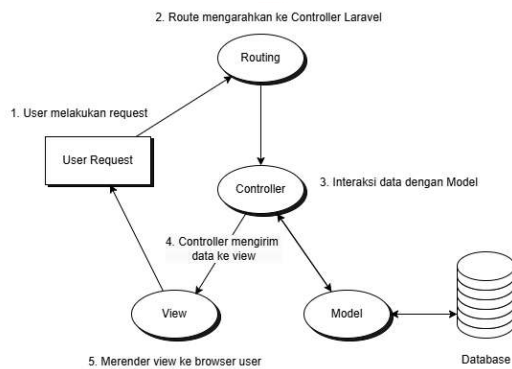
(Metode Prototyping)

C. Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan visual yang digunakan untuk mendeskripsikan, merancang, dan mendokumentasikan sistem perangkat lunak berorientasi objek. UML menyatukan berbagai notasi pemodelan untuk menggambarkan struktur dan perilaku sistem secara seragam, sehingga memudahkan pengembang dalam memahami dan merancang sistem yang kompleks [11].

D. Laravel

Laravel adalah *framework* PHP berbasis arsitektur *Model-View-Controller* (MVC) yang dikembangkan oleh Taylor Otwell untuk mempermudah dan mempercepat pengembangan aplikasi web melalui sintaks yang elegan dan fitur lengkap. Laravel dilengkapi dengan *Eloquent* ORM untuk interaksi database, *Blade Templating Engine* untuk tampilan dinamis, serta *Middleware* untuk pengelolaan akses. *Framework* ini juga mendukung *dependency injection*, menggunakan Composer untuk manajemen pustaka, dan menyediakan Artisan CLI untuk otomatisasi tugas. Dengan ekosistem luas dan komunitas aktif, Laravel menjadi pilihan populer dalam pengembangan aplikasi web yang efisien, aman, dan terstruktur [12].

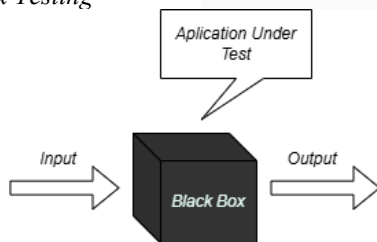


GAMBAR 3
(Alur MVC dalam Laravel)

E. Hypertext Preprocessor (PHP)

PHP adalah bahasa pemrograman *server-side* yang digunakan untuk membangun aplikasi web dinamis. PHP sering digunakan untuk memproses form, mengelola sesi, berinteraksi dengan database, dan menghasilkan konten web dinamis. PHP dijalankan di server dan dapat menghasilkan HTML yang dikirim ke browser pengguna [13].

F. Blackbox Testing



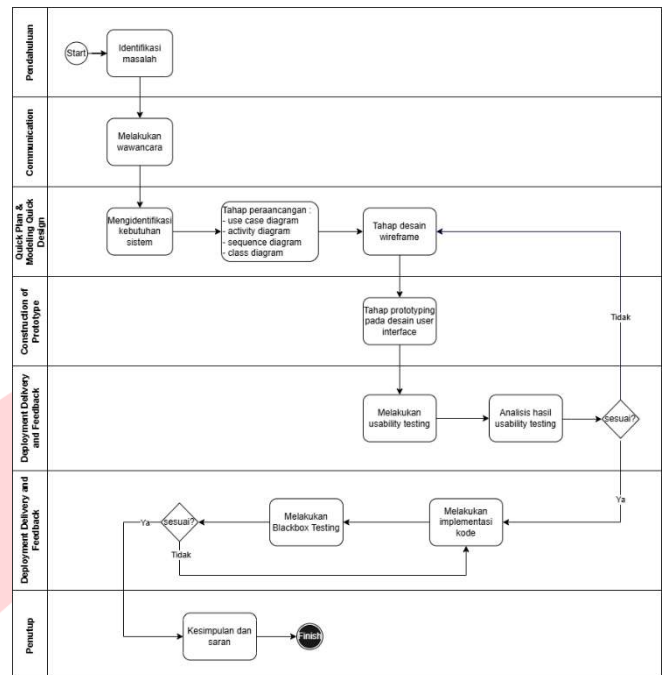
GAMBAR 3
Konsep Blackbox Testing

Blackbox Testing adalah metode pengujian perangkat lunak yang berfokus pada fungsi dan antarmuka aplikasi tanpa mempertimbangkan bagaimana kode sumbernya diimplementasikan. Metode ini digunakan untuk mengevaluasi kelayakan aplikasi dengan cara melakukan serangkaian tes yang dirancang untuk memastikan bahwa aplikasi memenuhi kebutuhan pengguna dan berfungsi sesuai spesifikasi yang telah ditetapkan [14].

III. METODE

Bab ini menjelaskan tahapan penelitian secara sistematis dan terstruktur, mulai dari identifikasi masalah, pengumpulan

data, hingga pengembangan dan pengujian sistem, guna memastikan proses berjalan logis, terarah, dan dapat dipertanggungjawabkan.



GAMBAR 4
(Sistematika Penyelesaian Masalah)

A. Identifikasi Masalah

Observasi dilakukan untuk memahami proses manajemen sumber daya manusia di Pelayanan Pastoral Mahasiswa Keuskupan Surabaya, yang saat ini masih mengandalkan pelaporan lisan dari koordinator divisi. Hal ini sering menyebabkan ketidakakuratan informasi, berisiko menimbulkan kesalahpahaman, keputusan yang kurang tepat, dan kegagalan program. Oleh karena itu, dibutuhkan perbaikan sistem pelaporan agar informasi lebih akurat dan dapat diandalkan.

B. Wawancara

Wawancara dilakukan untuk memvalidasi hasil observasi dan menggali informasi lebih mendalam terkait permasalahan dalam proses manajemen di PPMKS. Melalui wawancara dengan PLT, koordinator divisi, dan anggota divisi, diperoleh perspektif, pengalaman, serta kendala yang dihadapi langsung oleh pihak-pihak yang terlibat dalam kegiatan manajerial.

C. Identifikasi Kebutuhan Sistem

Analisis kebutuhan dilakukan sebagai dasar perancangan sistem, dengan memanfaatkan hasil wawancara untuk mengidentifikasi aktor yang terlibat dan fitur utama yang dibutuhkan. Tahap ini juga membantu menentukan skala prioritas pengembangan agar sistem yang dibangun sesuai dengan kebutuhan dan harapan pengguna akhir.

D. Perancangan

Pada tahap perancangan, digunakan beberapa diagram UML untuk menggambarkan struktur dan alur sistem. *Use case diagram* mengidentifikasi aktor dan fungsi seperti *login* dan *monitoring KPI*, *activity diagram* menunjukkan alur aktivitas, *sequence diagram* menggambarkan urutan interaksi antara aktor dan sistem, sedangkan *class diagram* merancang

struktur data dan hubungan antar kelas untuk memastikan sistem sesuai kebutuhan.

E. Wireframe

Pembuatan *wireframe* bertujuan memberikan gambaran awal tentang struktur dan tata letak antarmuka pengguna, sebagai *blueprint* visual yang menunjukkan susunan informasi, elemen navigasi, dan interaksi pengguna. Tahap ini membantu mengidentifikasi potensi masalah desain dan pengalaman pengguna sejak awal.

F. Prototyping Desain i Interface

Setelah *wireframe* dibuat, tahap selanjutnya adalah pembuatan *prototype* antarmuka pengguna untuk mensimulasikan penggunaan sistem secara langsung. Proses ini memungkinkan iterasi desain berdasarkan masukan pengguna, guna memastikan antarmuka mudah digunakan, sesuai kebutuhan, serta mendukung efisiensi dan efektivitas sistem.

G. Usability Tersting

Pada tahap ini dilakukan *usability testing* menggunakan alat ukur *Nielsen Attributes of Usability (NAU) Questionnaire*. Pengujian dilakukan untuk mengetahui sejauh mana sistem telah memenuhi aspek kegunaan (*usability*) berdasarkan lima atribut utama menurut Jakob Nielsen yaitu kemudahan dipelajari, efisiensi penggunaan, kemudahan diingat, tingkat kesalahan, serta kepuasan pengguna.

H. Analisis Testing

Setelah *usability testing* dengan metode NAU, dilakukan analisis hasil kuesioner berdasarkan lima atribut: *learnability*, *efficiency*, *memorability*, *errors*, dan *satisfaction*. Setiap pernyataan dikategorikan sesuai atributnya dan dianalisis dengan menghitung nilai rata-rata. Hasil analisis digunakan untuk mengidentifikasi area yang memenuhi ekspektasi serta bagian yang perlu diperbaiki, terutama jika terdapat atribut dengan skor rendah yang memerlukan evaluasi lebih lanjut.

I. Implementasi

Setelah analisis hasil testing menunjukkan hasil yang baik, maka implementasi kode siap dilakukan. Tahap implementasi dilakukan menggunakan *framework* Laravel untuk memastikan efisiensi dan kemudahan pengembangan. Proses ini mencakup implementasi kode berdasarkan desain yang telah melalui tahap *usability testing* dan mendapatkan persetujuan pengguna. Pengembangan memastikan setiap fitur berfungsi sesuai dengan spesifikasi yang telah ditentukan dalam tahap perencanaan. Setelah implementasi selesai, dilakukan pengujian kembali untuk memastikan tidak ada bug atau kesalahan dalam sistem.

J. Blackbox Testing

Tahap ini merupakan tahap untuk menguji kode apakah output yang dihasilkan oleh sistem sesuai dengan input yang diberikan, serta memeriksa apakah semua fitur berfungsi dengan baik.

IV. HASIL DAN PEMBAHASAN

Bab ini menjelaskan hasil dan pembahasan dari setiap tahapan alur penelitian yang telah dilakukan oleh penulis, mulai dari pengumpulan data saat wawancara, identifikasi kebutuhan sistem, tahap perancangan, desain *user interface*, *usability testing*, analisis hasil testing, implementasi kode, *white box testing*, *black box testing* serta analisis hasil testing.

A. Wawancara

Hasil wawancara menunjukkan bahwa manajemen organisasi masih bersifat manual tanpa sistem terintegrasi. Pemantauan kinerja divisi hanya dilakukan saat rapat, dengan dokumen tersebar di berbagai *platform* (Google Drive, Synology), sehingga menyulitkan evaluasi. Tidak adanya aplikasi pendukung distribusi tugas dan pemantauan progres menyebabkan hambatan divisi sering tidak teridentifikasi atau tersampaikan dengan baik.

TABEL 1
(Hasil Requirement Gathering)

Jabatan	Topik	Hasil
PLT	Alur kerja organisasi, permasalahan utama serta pembahasan sistem yang ingin dibuat.	Proses manajemen masih manual dan belum didukung sistem informasi terintegrasi, sehingga menghambat efisiensi kerja. Diperlukan sistem yang mampu memantau kinerja anggota dalam pelaksanaan program kerja.
Ketua	Struktur organisasi, rancangan sistem yang akan dikembangkan, serta kebutuhan nonfungsionalnya.	Penjelasan mencakup struktur kepengurusan PPMKS serta harapan agar sistem mampu menyajikan informasi kegiatan, mengatasi hambatan, memiliki tampilan menarik, dan mudah dioperasikan.
Koordinator	Kebutuhan fitur yang akan dikembangkan dalam sistem.	Kebutuhan akan fitur distribusi tugas, laporan berkala serta pemantauan progress.
Anggota	Kebutuhan fitur untuk peran anggota divisi	Kebutuhan akan fitur yang memungkinkan anggota untuk menerima dan mengelola tugas dari Koordinator serta fitur yang memungkinkan anggota untuk melaporkan hasil kegiatan dan kendala yang dihadapi

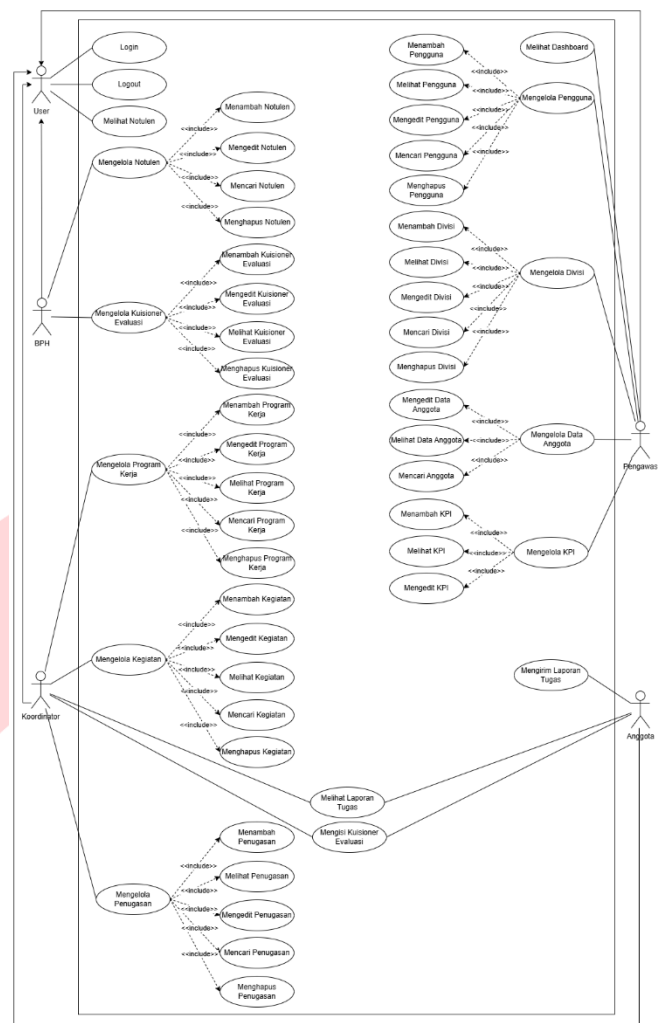
B. Identifikasi Kebutuhan Sistem

Berdasarkan hasil wawancara dan analisis, sistem yang akan dikembangkan melibatkan empat aktor utama, yaitu Pengawas, Sekretaris (BPH), Koordinator Divisi, dan Anggota Divisi, dengan masing-masing peran dan wewenang yang telah terdefinisi. Kebutuhan sistem mencakup fitur fungsional seperti manajemen program kerja, penugasan, pelaporan tugas, dan pengisian kuisisioner evaluasi. Analisis kebutuhan dilakukan melalui studi literatur dan benchmarking terhadap sistem serupa, sehingga diperoleh daftar fitur inti yang mendukung pemantauan dan evaluasi kinerja secara efektif. Selain itu, kebutuhan *non-fungsional* seperti tampilan yang ramah pengguna dan pengujian kegunaan juga menjadi fokus untuk memastikan sistem mudah dioperasikan oleh seluruh aktor.

TABEL 2
(Identifikasi Kebutuhan Fungsional)

Aktor	Fitur	Deskripsi
Pengawas	Login	Pengawas dapat masuk ke sistem dengan menggunakan akun sesuai dengan peran.
	Kelola Pengguna	Pengawas dapat menambah, mengedit, dan menghapus pengguna.
	Kelola Anggota	Pengawas dapat mengelola informasi setiap anggota
	Kelola Divisi	Pengawas dapat menambah, mengedit, dan menghapus data divisi.
	Lihat Data Anggota	Pengawas dapat melihat daftar anggota seluruh divisi.
	Lihat KPI	Pengawas dapat melihat <i>Dashboar</i> d KPI secara keseluruhan.
	Logout	Pengawas dapat keluar dari sistem dan mengakhiri sesi
BPH	Login	BPH dapat masuk ke sistem dengan akun masing-masing.

	Input Notulen Rapat	BPH dapat menambahkan notulen rapat ke sistem.
	Edit & Hapus Notulen	BPH dapat mengedit dan menghapus notulen rapat yang telah dibuat.
	Buat Kuisisioner Evaluasi	BPH dapat membuat kuisisioner baru, menentukan KPI, periode, dan responden.
	Input Pertanyaan Kuisisioner	BPH dapat menambahkan pertanyaan dalam kuisisioner.
	Logout	BPH dapat keluar dari sistem dan mengakhiri sesi
Koordinator Divisi	Login	Koordinator dapat masuk ke sistem dengan akun sesuai peran.
	Kelola Program Kerja	Koordinator dapat membuat, mengedit, dan menghapus program kerja divisinya.
	Kelola Kegiatan	Koordinator dapat menambah, mengedit, dan menghapus kegiatan di bawah program kerja.
	Kelola Penugasan	Koordinator dapat memberikan tugas ke anggota dan mengedit atau menghapus tugas.
	Lihat Laporan Tugas	Koordinator dapat melihat laporan dari anggota atas tugas yang diberikan.
	Isi Kuisisioner	Koordinator dapat mengisi kuisisioner yang aktif sesuai perannya.
	Logout	Koordinator dapat keluar dari sistem dan mengakhiri sesi
Anggota Divisi	Login	Anggota dapat masuk ke sistem dengan akun masing-masing.
	Lihat Tugas	Anggota dapat melihat daftar tugas yang diterima.
	Isi Laporan Tugas	Anggota dapat mengisi dan mengunggah laporan tugas.
	Edit Laporan Tugas	Anggota dapat mengedit laporan yang sudah dikirim (jika diizinkan sistem).
	Isi Kuisisioner	Anggota dapat mengisi kuisisioner evaluasi yang aktif.
	Logout	Anggota dapat keluar dari sistem dan mengakhiri sesi

GAMBAR 5
(Usecase)

C. Perancangan

Pada tahap perancangan, *use case diagram* disusun untuk menggambarkan fungsionalitas utama sistem informasi manajemen berbasis KPI serta interaksi antara pengguna (aktor) dan sistem. Sistem ini melibatkan empat aktor utama, yaitu Pengawas, BPH (Badan Pengurus Harian), Koordinator, dan Anggota, yang masing-masing memiliki hak akses sesuai perannya dalam proses monitoring dan evaluasi pelayanan pastoral mahasiswa. Aktor umum, yaitu *User*, memiliki akses dasar untuk melakukan *login* dan *logout*. BPH memiliki tanggung jawab dalam pengelolaan notulen dan kuesioner evaluasi, termasuk fitur untuk menambah, mengedit, menghapus, dan melihat data, serta mengakses data responden kuesioner.

Koordinator berwenang mengelola program kerja, kegiatan, serta mendistribusikan tugas kepada anggota. Koordinator juga dapat memantau laporan tugas yang telah dikirimkan. Anggota hanya memiliki akses untuk melihat tugas, mengunggah laporan, dan mengisi kuesioner evaluasi. Sementara itu, Pengawas bertindak sebagai super-user yang memiliki akses penuh terhadap *dashboard* sistem, pengelolaan data pengguna, divisi, anggota, serta indikator KPI. Seluruh akses mencakup fungsi pencarian, perubahan status, serta pengelolaan data secara menyeluruh. Perancangan ini memastikan sistem mampu mendukung proses kerja dan evaluasi secara terstruktur sesuai dengan peran masing-masing aktor.

D. Wireframe

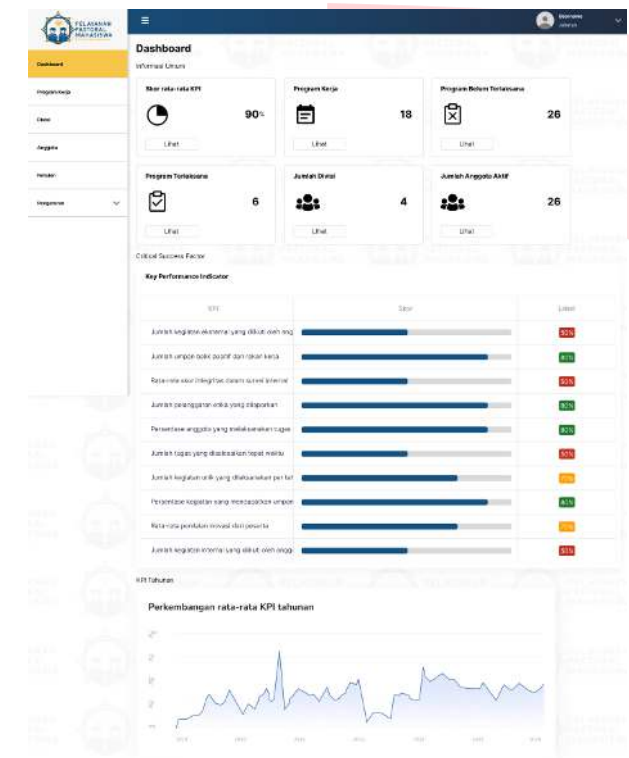
GAMBAR 6
(Wireframe)

Tahap ini merupakan pembuatan rancangan awal antarmuka pengguna (UI) dalam bentuk *wireframe*. Bertujuan untuk memberikan gambaran struktur tata letak

dan alur navigasi sistem sebelum dilakukan desain visual yang lebih detail. *Wireframe* juga membantu tim pengembang dan pemangku kepentingan menyamakan persepsi mengenai kebutuhan fungsional dan user flow aplikasi.

E. Prototyping User Interface

Setelah *wireframe* selesai, tahap selanjutnya adalah proses *prototyping user interface*. Dalam proses ini dilakukan pembuatan *high fidelity user interface* yang menampilkan desain visual lebih rinci, termasuk warna, ikon, dan elemen-elemen UI lain yang menyerupai tampilan akhir aplikasi. Tahap ini juga memungkinkan pengujian awal terhadap interaksi dan alur pengguna sebelum masuk ke tahap pengembangan.



GAMBAR 7
(Prototype)

F. Usability Testing

Berdasarkan hasil *Usability Testing*, diperoleh rata-rata penilaian tertinggi pada hampir seluruh pertanyaan berada pada kategori “Sangat Puas (SP)” dan “Puas (P)”. Dimensi *Learnability* memiliki rata-rata sebesar 4,77, menunjukkan bahwa pengguna merasa sistem mudah dipelajari. Dimensi *Efficiency* memiliki nilai rata-rata sebesar 4,68, menandakan sistem dianggap efisien saat digunakan. Dimensi *Error* mendapatkan nilai 4,8, menunjukkan bahwa sistem minim kesalahan atau mudah dikoreksi. *Memorability* juga memperoleh nilai tinggi yaitu 4,8, menandakan pengguna mudah mengingat cara menggunakan sistem. Terakhir, dimensi *Satisfaction* mendapat rata-rata 4,72 yang mencerminkan kepuasan pengguna secara umum terhadap sistem. Hasil ini menunjukkan bahwa sistem telah memenuhi aspek usability dengan baik.

TABEL 3
(Hasil Usability Testing)

Dimensi	Pertanyaan	Kepala Kolom Tabel					Rata-Rata
		STP	TP	CP	P	SP	
Learnability	L1	0	0	0	5	15	4,75
	L2	0	0	0	4	16	4,8
	L3	0	0	0	5	15	4,75
Efficiency	EF1	0	0	0	10	10	4,5
	EF2	0	0	0	5	15	4,75
	EF3	0	0	0	4	16	4,8
Error	ER1	0	0	0	4	16	4,8
Memorability	M1	0	0	0	4	16	4,8
Satisfaction	S1	0	0	0	4	16	4,8
	S2	0	0	0	7	13	4,65

G. Analisis Hasil

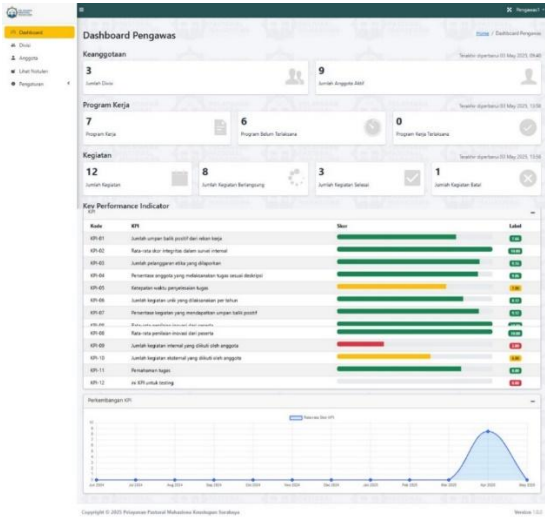
Dimensi *learnability* yang mencakup aspek *visibility of system status*, *match between system and the real world*, serta *help and documentation* menunjukkan bahwa sistem mudah dipahami sejak awal penggunaan. Pada dimensi *Efficiency*, indikator seperti *user control and freedom*, *flexibility and efficiency of use*, serta *help users diagnose and recover from errors* mencerminkan performa sistem yang efisien dan mendukung kontrol pengguna. Dimensi *Error* menunjukkan bahwa sistem mampu meminimalkan kesalahan serta memberikan pencegahan yang memadai. *Memorability* memperoleh respons “Sangat Puas” pada indikator *recognition rather than recall*, menandakan pengguna mudah mengingat cara penggunaan sistem setelah jeda waktu. Sementara itu, dimensi *Satisfaction* juga mendapat penilaian tinggi pada indikator *consistency and standards* serta *aesthetic and minimalist design*, yang mencerminkan tampilan antarmuka sistem yang menarik dan konsisten.

TABEL 4
(Analisis Usability Testing)

Dimensi	Prinsip	Interpretasi
Learnability	<i>Visibility of system status</i>	Sangat Puas
	<i>Match between system and the real world</i>	Sangat Puas
	<i>Help and documentation</i>	Sangat Puas
Efficiency	<i>User control and freedom</i>	Sangat Puas
	<i>Flexibility and efficiency of use</i>	Sangat Puas
	<i>Help users, diagnose and recover from errors</i>	Sangat Puas
Error	<i>Error prevention (Errors)</i>	Sangat Puas
Memorability	<i>Recognition rather than recall</i>	Sangat Puas
Satisfaction	<i>Consistency and standards</i>	Sangat Puas
	<i>Aesthetic and minimalist design</i>	Sangat Puas

H. Implementasi

Proses ini menggunakan *framework* Laravel sebagai alat pengembangan utama, serta MySQL sebagai sistem manajemen basis data untuk menyimpan seluruh informasi yang dikelola oleh aplikasi.



GAMBAR 8
(Hasil Implementasi)

I. Blackbox Testing

Pengujian perangkat lunak dilakukan untuk memastikan sistem berfungsi sesuai kebutuhan pengguna dan spesifikasi yang telah ditentukan. Pengujian dilakukan berdasarkan skenario fitur utama untuk mengevaluasi kinerja, keandalan, dan ketepatan fungsi, serta menilai kesiapan sistem untuk diterapkan secara nyata.

TABEL 1
(Hasil Blackbox Testing)

Skenario Pengujian	Prosedur Pengujian	Hasil yang diharapkan	Hasil
Login	Pengawas mencoba login dengan akun yang salah	Sistem menampilkan pesan <i>error</i> bahwa akun tidak ditemukan atau kredensial salah	Valid
	Pengawas mencoba login dengan format email salah	Sistem menampilkan pesan <i>error</i> validasi format <i>email</i>	Valid
	Pengawas mencoba login dengan akun yang benar (hak akses Pengawas)	Pengguna berhasil login dan diarahkan ke halaman pengawas	Valid
Kelola Pengguna	Pengawas menekan tombol tambah, mengisi formulir dan menekan tombol simpan	Data pengguna baru berhasil ditambahkan dan muncul di daftar pengguna	Valid
	Pengawas menekan tombol edit, mengisi form dan menekan tombol simpan	Data pengguna berhasil diperbarui sesuai input	Valid
	Pengawas menekan tombol lihat	Detail informasi pengguna ditampilkan dengan benar	Valid
Kelola Anggota	Pengawas menekan tombol edit, mengisi form dan menekan tombol simpan	Data anggota berhasil diperbarui dan ditampilkan pada daftar anggota	Valid

Skenario Pengujian	Prosedur Pengujian	Hasil yang diharapkan	Hasil
	Pengawas menekan tombol lihat	Detail informasi anggota ditampilkan dengan benar	Valid
Kelola Divisi	Pengawas menekan tombol tambah , mengisi formulir lalu	Data divisi baru berhasil ditambahkan dan muncul di daftar divisi	Valid
	Pengawas menekan tombol tambah anggota, memilih list anggota dan menekan tombol simpan	Anggota berhasil ditambahkan ke dalam divisi yang dipilih	Valid
	Pengawas menekan tombol edit , mengisi formulir dan menekan tombol simpan	Data divisi berhasil diperbarui dan ditampilkan pada daftar divisi	Valid
	Pengawas menekan tombol lihat	Detail informasi divisi dan daftar anggota ditampilkan dengan benar	Valid
Melihat Dashboard KPI	Pengawas menekan menu Dashboard	Informasi KPI ditampilkan sesuai data, dan riwayat KPI bulanan ditampilkan sesuai rata-rata nilai KPI setiap bulan	Valid
Melihat Notulen	Pengawas menekan tombol lihat pada halaman notulen	Detail notulen dan iterasinya ditampilkan dengan benar	Valid
Logout	Pengawas menekan tombol logout	Sistem berhasil keluar dan diarahkan ke halaman login	Valid

V. KESIMPULAN

Penelitian ini berhasil merancang dan mengembangkan sistem informasi manajemen (SIM) berbasis KPI untuk mendukung monitoring dan evaluasi anggota di lingkungan PPMKS. Pengembangan dilakukan dengan metode prototyping yang melibatkan pengguna secara aktif untuk memastikan kesesuaian sistem dengan kebutuhan organisasi. Tahapan penelitian mencakup identifikasi kebutuhan melalui observasi dan wawancara, perancangan sistem menggunakan UML, pengembangan antarmuka berbasis prototipe, serta pengujian usability dua kali menggunakan NAU. Implementasi sistem dilakukan dengan framework Laravel dan basis data MySQL. Sistem yang dihasilkan menyediakan fitur utama seperti login, dashboard KPI, manajemen data pengguna dan divisi, pengelolaan notulen, kuisioner evaluasi, program kerja, kegiatan, penugasan, dan pelaporan tugas. Antarmuka dirancang intuitif dan mudah digunakan oleh seluruh peran pengguna. Pengujian usability menunjukkan tingkat kepuasan pengguna yang tinggi. Blackbox testing memastikan seluruh fitur berjalan sesuai skenario. Sistem ini juga telah mempertimbangkan keterkaitan dengan indikator KPI organisasi meskipun evaluasi kuantitatif KPI belum dilakukan karena di luar ruang lingkup penelitian.

REFERENSI

- [1] T. Fajriah and E. R. Ningsih, "PENGARUH TEKNOLOGI KOMUNIKASI TERHADAP INTERAKSI SOSIAL DI ERA DIGITAL," 2024.
- [2] S. Dewi and L. Warlina, "PENGARUH PENERAPAN SISTEM INFORMASI MANAJEMEN (SIM), KOMUNIKASI DAN MOTIVASI KERJA TERHADAP KINERJA PEGAWAI PADA DINAS SOSIAL KOTA BIMA," 2022.
- [3] Norliani, M. N. Sari, M. S. Safarudin, R. Jaya, Baharuddin, and A. R. Nugraha, "TRANSFORMASI DIGITAL DAN DAMPAKNYA PADA ORGANISASI: TINJAUAN TERHADAP IMPLEMENTASI TEKNOLOGI INFORMATIKA," Jul. 2024.
- [4] R. Arya Dilla and I. Nugroho, "Sistem Kearsipan Surat Berbasis Web Pada Biro ISDA Sekretariat Daerah Provinsi Jateng," 2023.
- [5] F. Ali, T. Siregar, M. Irwan, and P. Nasution, "Peran Sistem Informasi Manajemen dalam Meningkatkan Kinerja Organisasi," *Jurnal Sistem Informasi dan Ilmu Komputer*, no. 2, pp. 137–145, 2024, doi: 10.59581/jusiik-widyakarya.v2i2.3849.
- [6] N. M. Fadilla and W. Setyonugroho, "Sistem Informasi Manajemen Rumah Sakit Dalam Meningkatkan Efisiensi: Mini Literature Review," *Jurnal Teknik Informatika dan Sistem Informasi*, 2021.
- [7] M. Panjaitan *et al.*, "Manajemen Kinerja (Perspektif Balanced Scorecard)," 2020.
- [8] I. A. Musdar and H. Arfandy, "RANCANG BANGUN SISTEM INFORMASI PARIWISATA SULAWESI SELATAN BERBASIS ANDROID DENGAN MENGGUNAKAN METODE PROTOTYPING," *SINTECH JOURNAL*, 2020, [Online]. Available: <https://doi.org/10.31598>
- [9] S. E. , M. M. , C. DMP. , C. Erwin *et al.*, "SISTEM INFORMASI MANAJEMEN (Teori, Prinsip dan Penerapan)," 2024.
- [10] A. W. Budyasmoto and Mei Prabowo, *Metodologi Pengembangan Sistem Informasi*. 2020.
- [11] Indriyani Fintri, Yunita, Muthia Dinda Ayu, Surniandari Artika, and Sriyadi, "Analisa Perancangan Sistem Informasi," 2020.
- [12] Laravel, "Laravel Documentation."
- [13] A. B. Santoso, "PEMROGRAMAN WEB PHP DASAR," 2022.
- [14] E. Hakimah Kusuma Dewi, I. Shiddiq Pratama, A. Sukma Putera, T. Informatika, and U. Singaperbangsa Karawang, "STRING (Satuan Tulisan Riset dan Inovasi Teknologi) BLACK BOX TESTING PADA APLIKASI PENCATATAN PEMINJAMAN BUKU MENGGUNAKAN BOUNDARY VALUE ANALYSIS," 2022.