

BAB 2

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Profil Kantor Camat Rantau Kopar

Kecamatan Rantau Kopar merupakan organisasi perangkat daerah yang dipimpin oleh kepala pemerintahan kecamatan (pejabat administrator) yaitu camat yang kedudukannya merupakan perpanjangan tangan dari pemerintah Kabupaten Rokan Hilir sebagai bawahan langsung dari Sekretaris Daerah. Sesuai dengan tugas dan fungsinya, pemerintah kecamatan mempunyai kewenangan mengatur dan menyelenggarakan kegiatan pemerintahan di tingkat kecamatan dan bertanggung jawab kepada Bupati Rokan Hilir. Tanggung jawab pemerintah kecamatan adalah menjalankan roda pemerintahan satu tingkat di bawah Pemerintah Kabupaten Rokan Hilir. Tanggung jawab tersebut mencakup pengaturan kewenangan dalam berbagai bidang yang berada pada wilayah tersebut sesuai dengan delegasi dan pengendalian terhadap dinamika kehidupan sosial masyarakat baik ekonomi, politik, maupun sosial budaya.



Gambar 2.1 Profil Kantor Camat Rantau Kopar

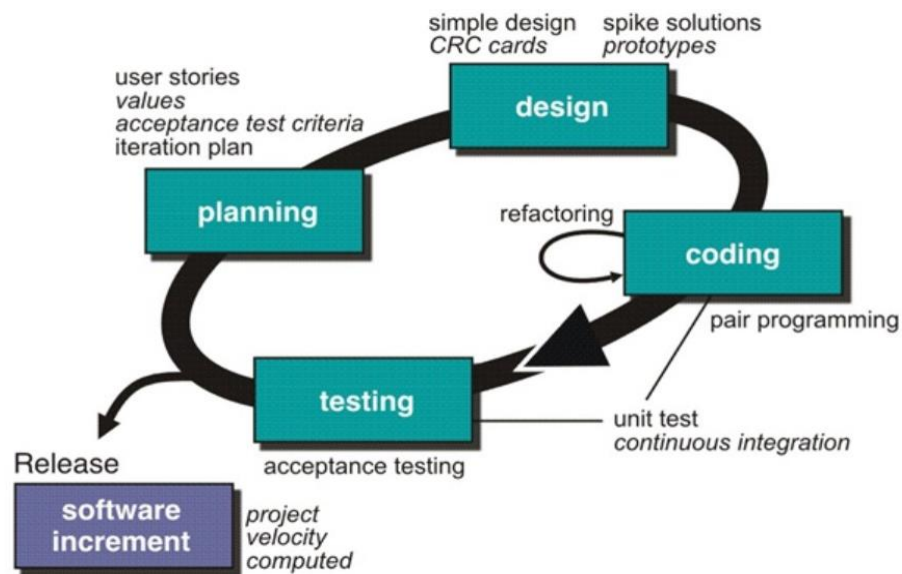
Kebutuhan data atau informasi berguna untuk perencanaan kependudukan, sosial ekonomi, dan bidang lainnya yang dirasakan mengalami peningkatan, saat ini belum banyak tersedia informasi di bidang kependudukan yang lengkap dan

menyeluruh sehingga berdampak pada pelayanan yang diberikan kepada masyarakat. Kebutuhan data jumlah kependudukan tahun 2021 bersumber dari data kependudukan yang terdiri dari 2 kelurahan dan 2 kepenghuluan se-kecamatan Rantau Kopar. Hal ini karena pendataan dan rekapitulasi data berdasarkan pengurusan pelayanan di setiap kelurahan dan kepenghuluan dilaksanakan pada akhir tahun 2020. Data kependudukan bidang pemerintahan kecamatan Rantau Kopar bersifat de jure atau data penduduk yang terdaftar sesuai dengan jumlah penduduk yang terdata dalam mengurus data kependudukan di setiap desa dan kelurahan. Kantor Camat Rantau Kopar memiliki 8 jenis layanan utama yaitu rekomendasi penelitian / riset, rekomendasi surat pindah, rekomendasi akta kelahiran, rekomendasi kartu keluarga, rekomendasi kerja, penerbitan surat tanah SKT dan SKGR, surat keterangan ahli waris, dan rekomendasi yayasan, TPQ, dan Ormas (PPID, 2021).

2.1.2 Metodologi Extreme Programming

XP atau yang disingkat dengan (*Extreme Programming*) adalah salah satu metode dalam pengembangan perangkat lunak yang dipelopori oleh Kent Beck, Ron Jeffries, dan Ward Cunningham. XP merupakan salah satu metode perangkat lunak yang sederhana pada berbagai tahapan pengembangan sistem menjadi lebih efisien, adaptif, dan fleksibel. Pada metode XP ini memiliki nilai dasar programming. Di antara nilai dasar programming itu menurut A. Fatoni adalah:

- a. *Communicator* yaitu berfokus pada komunikasi yang baik antara programming dengan user maupun sesama programming
- b. *Courage* adalah pengembangan perangkat lunak harus selalu memiliki keyakinan, keberanian, dan integrasi dalam melakukan tugasnya
- c. *Simplicity* adalah melakukan semua tugas dengan sederhana
- d. *Feedback* yaitu mendapatkan umpan balik sehingga anggota tim berkualitas dalam bekerja
- e. *Quality Work* merupakan proses akhir dari pengembangan perangkat lunak yang berkualitas.



Gambar 2.2 Tahapan XP

Extreme Programming terbentuk dari model pengembangan perangkat lunak yang memiliki tahapan pengembangan sistem menjadi lebih efisien, adaptif, dan fleksibel Panca Putra & Novrian (2023). XP tidak hanya berfokus pada coding akan tetapi meliputi seluruh bagian pengembangan perangkat lunak. Tahapan-tahapan dalam metode pengembangan Extreme Programming adalah:

1. *Planning* (Perencanaan)

Planning merupakan tahap awal dalam pengembangan sistem. Pada tahap ini dilakukan identifikasi masalah, pengumpulan kebutuhan pengguna, analisis proses bisnis, penyusunan *user story*, penentuan prioritas fitur, serta penyusunan jadwal pengembangan. Tahap ini dilakukan bersama antara pengembang *front end* dan *back end* agar fitur yang dikembangkan sesuai dengan kebutuhan masyarakat, petugas, dan kepala camat.

2. *Design* (Perancangan)

Tahap *design* dilakukan untuk memberikan gambaran mengenai sistem dan antarmuka yang akan dikembangkan. Perancangan pada penelitian ini meliputi arsitektur sistem, use case diagram, use case scenario, CRC Card, user persona, serta rancangan antarmuka. Pada bagian *front end*, perancangan difokuskan pada tata letak halaman, warna, navigasi, formulir, komponen antarmuka, dan

tampilan responsif. Perancangan juga memperhatikan proses integrasi dengan RESTful API agar data dari *back end* dapat ditampilkan pada antarmuka.

3. *Coding* (Pengkodean)

Tahap *coding* merupakan proses menerapkan hasil perancangan ke dalam bentuk program. Pada penelitian ini, pengembangan *front end* dilakukan menggunakan ReactJS. Proses pengkodean meliputi pembuatan komponen antarmuka, pengaturan navigasi, pengelolaan formulir dan validasi input, pengaturan tampilan berdasarkan hak akses pengguna, serta integrasi RESTful API. Antarmuka juga dikembangkan agar dapat menyesuaikan tampilan pada perangkat komputer maupun perangkat bergerak.

4. *Testing* (Pengujian)

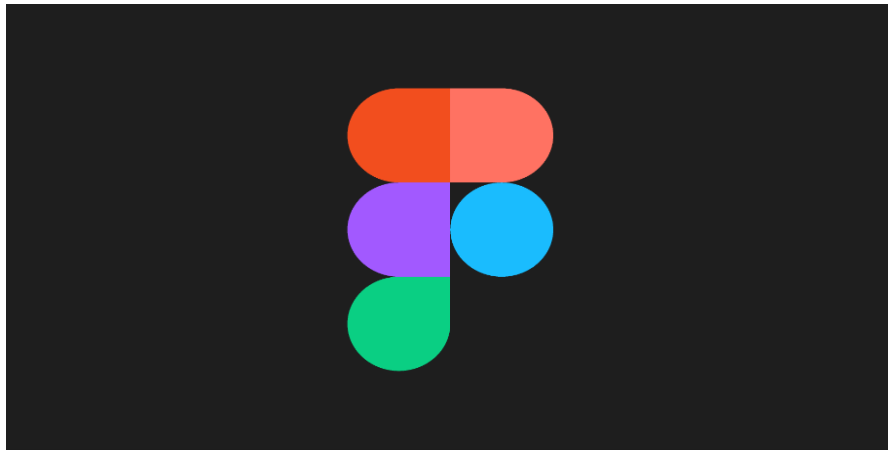
Tahap *testing* dilakukan untuk memastikan bahwa antarmuka dan fungsi sistem berjalan sesuai dengan kebutuhan pengguna. Pengujian pada penelitian ini meliputi *unit testing* untuk menguji bagian atau fungsi pada *front end*, User Acceptance *Testing* untuk mengetahui penerimaan pengguna terhadap sistem, serta *usability testing* untuk mengukur kemudahan pengguna dalam menggunakan antarmuka. Hasil pengujian digunakan sebagai umpan balik untuk melakukan perbaikan pada setiap iterasi.

2.1.3 *E-Governance*

Implementasi e-governance dalam pelayanan publik pada dasarnya diarahkan untuk mendorong terciptanya birokrasi yang lebih terbuka, efektif, dan responsif. Menurut Prihartono & Tuti (2023), prinsip e-governance tidak hanya menekankan aspek teknologi, tetapi juga pada nilai-nilai utama tata kelola pemerintahan modern, yaitu competence (profesionalisme aparatur), accountability (pertanggungjawaban setiap keputusan dan layanan), participation (pelibatan masyarakat dalam proses layanan), rule of law (kepastian hukum dalam penyelenggaraan administrasi), serta social justice (keadilan sosial agar layanan dirasakan merata). Melalui penerapan nilai-nilai tersebut, e-governance diharapkan mampu memperkuat kualitas pelayanan digital pemerintah, meningkatkan kepercayaan publik, sekaligus memastikan bahwa transformasi birokrasi berjalan sesuai prinsip good governance.

2.1.4 UI/UX (*User Interface / User Experience*)

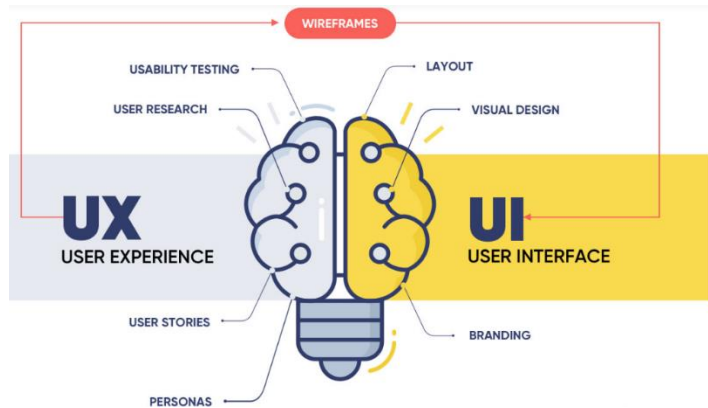
Figma adalah alat desain yang sering digunakan untuk merancang antarmuka aplikasi mobile, desktop, maupun *website*. Aplikasi ini dapat dijalankan di berbagai sistem operasi seperti Windows, Linux, dan macOS selama terhubung dengan internet. Figma banyak digunakan oleh profesional di bidang UI / UX karena memiliki fitur yang lengkap seperti Adobe XD. Ilustrasi antarmuka Figma ditunjukkan pada Gambar 2.3.



Gambar 2.3 Figma

Salah satu keunggulan utamanya adalah kemampuan kolaborasi secara *real-time*, memungkinkan beberapa desainer bekerja bersama meskipun berada di lokasi berbeda. Fitur ini menjadikan Figma sangat efektif untuk membuat *prototype* secara cepat dan efisien dalam tim (Panca Putra & Novrian, 2023).

UI (User Interface) adalah desain tampilan antarmuka yang berfokus pada estetika, seperti pemilihan warna, layout, dan elemen visual lainnya agar tampilan *website* atau aplikasi menjadi menarik dan nyaman digunakan. UX (User Experience) adalah proses yang berfokus pada kenyamanan dan kemudahan pengguna saat berinteraksi dengan sistem, memastikan navigasi mudah, alur penggunaan jelas, dan pengalaman pengguna terasa menyenangkan (Panca Putra & Novrian, 2023). Ilustrasi komponen UI/UX ditunjukkan pada Gambar 2.4.

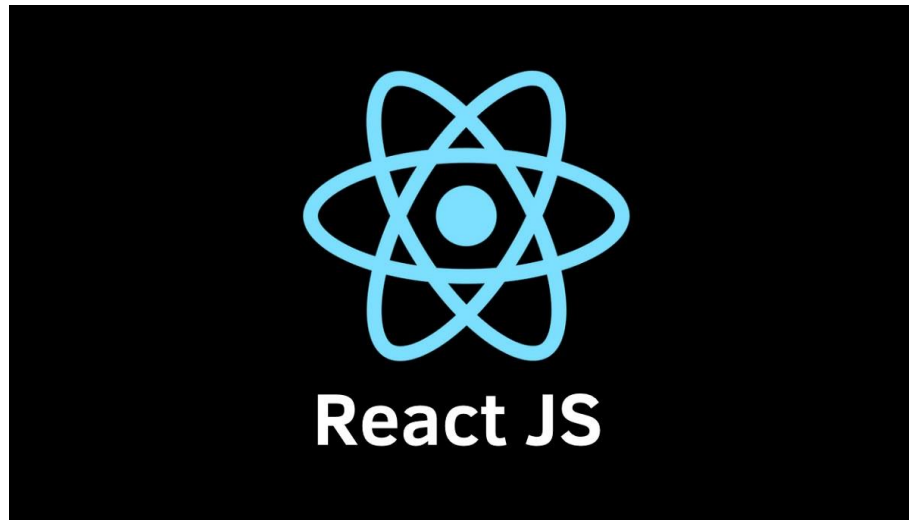


Gambar 2.4 UI/UX

Dalam desain antarmuka, terdapat dua aspek utama yang saling melengkapi, yaitu User Interface (UI) dan User Experience (UX). UI lebih berfokus pada tampilan visual, seperti tata letak, pemilihan tipografi, dan perpaduan warna, agar sistem terlihat menarik dan konsisten. Sementara itu, UX berfokus pada pengalaman pengguna saat berinteraksi dengan sistem, mencakup kemudahan navigasi, kejelasan alur, dan efektivitas dalam penggunaan. Perpaduan antara UI dan UX yang dirancang dengan baik akan menciptakan sistem yang tidak hanya bagus dilihat, tetapi juga nyaman dan efisien saat digunakan oleh masyarakat.

2.1.5 *ReactJS*

ReactJS adalah library JavaScript yang dikembangkan oleh Facebook dan banyak digunakan untuk membangun antarmuka web dan aplikasi mobile. React memudahkan pengembangan komponen UI yang interaktif, dinamis, dan dapat digunakan ulang, serta sangat ideal untuk membuat tampilan yang kompleks namun tetap memiliki performa tinggi (Muhyidin dkk., 2020) Ilustrasi ReactJS ditunjukkan pada Gambar 2.5.



Gambar 2.5 ReactJS

ReactJS memiliki sejumlah fitur utama yang dirancang untuk mempermudah pengembangan antarmuka pengguna secara efisien, modular, dan dinamis. Fitur-fitur ini mendukung proses pembangunan aplikasi yang terstruktur dan mudah dikembangkan, baik untuk proyek skala kecil maupun besar (Muhyidin dkk., 2020). Beberapa fitur penting dalam ReactJS antara lain :

1) *JavaScript XML (JSX)*

JSX adalah ekstensi sintaks *JavaScript* yang digunakan untuk membuat elemen pada *React*. Dengan JSX, pengembang dapat menulis HTML langsung dalam file *JavaScript*, sehingga mempermudah dalam menggabungkan logika pemrograman dengan tampilan antarmuka secara lebih intuitif.

2) *Virtual DOM*

React menggunakan *Virtual DOM* untuk merepresentasikan struktur halaman dalam bentuk pohon data. Ketika terjadi perubahan, hanya bagian yang terdampak saja yang diperbarui, tanpa merender ulang seluruh halaman. Hal ini membuat performa aplikasi menjadi lebih cepat dan efisien.

3) *Komponen dan Props*

React membagi UI menjadi bagian-bagian kecil yang disebut komponen. Setiap komponen memiliki fungsi dan tampilan tersendiri, sehingga mudah digunakan kembali dan dirawat. Data dapat dikirim antar komponen melalui properti atau props

4) Manajemen *State*

State adalah objek yang menyimpan kondisi dinamis suatu komponen. Ketika pengguna melakukan interaksi, *state* akan berubah dan UI akan diperbarui secara otomatis. Untuk aplikasi skala besar, pengelolaan *state* dapat dibantu dengan library seperti Redux dan Recoil yang mempermudah pengaturan data dan komunikasi antar komponen.

5) Navigasi Terprogram (Programmatic Navigation)

Navigasi terprogram memungkinkan sistem mengarahkan pengguna ke halaman tertentu berdasarkan aksi, tanpa perlu mengklik link. Contohnya pada saat login, pengguna otomatis diarahkan ke *dashboard*. React menyediakan library bernama React Router untuk mengelola navigasi ini secara dinamis dan aman antar komponen

2.1.6 *User Persona*

User Persona adalah gambaran representatif dari pengguna ideal yang dibuat berdasarkan riset dan data untuk memahami kebutuhan, tujuan, dan perilaku pengguna dalam menggunakan suatu sistem. Meskipun bersifat fiktif, user persona ini dirancang sedetail mungkin agar mencerminkan karakter nyata dari pengguna. dengan user persona, tim pengembang bisa lebih fokus merancang pengalaman dan antarmuka sesuai kebutuhan pengguna. Proses perancangan jadi lebih terarah, karena tim lebih mudah memahami fitur yang dibutuhkan, alur yang tepat, serta cara penyajian informasi yang efektif. Selain itu, user persona membantu pengambilan keputusan yang berpusat pada pengguna, bukan sekadar asumsi tim. (Mustada dkk., 2025). Ilustrasi user persona ditunjukkan pada Gambar 2.6.



Gambar 2.6 User Persona

Penerapan user persona dalam pengembangan antarmuka membantu pengembang memahami kebutuhan, tujuan, dan kendala pengguna. Dengan mengacu pada persona, tampilan dan alur interaksi dapat dirancang agar lebih relevan dan mudah digunakan. Pada penelitian ini, user persona dibuat untuk mewakili tiga jenis pengguna Sistem Pelayanan Terpadu, yaitu masyarakat, petugas, dan kepala camat.

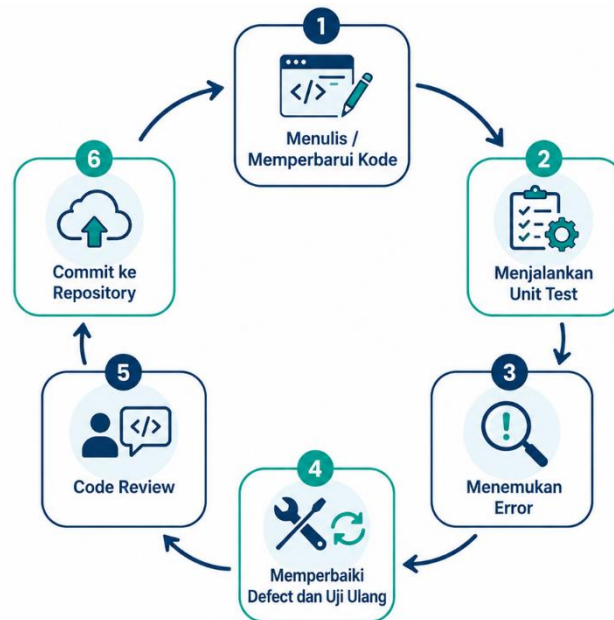
2.1.7 Unit Testing

Unit Testing merupakan teknik pengujian perangkat lunak yang berfokus pada bagian terkecil dari kode program secara terpisah. Pengujian ini dilakukan untuk memastikan bahwa setiap fungsi dapat berjalan sesuai dengan hasil yang diharapkan serta membantu menemukan kesalahan sejak awal proses pengembangan (Hasibuan & Dirgahayu, 2020).

Pada pengembangan *front end*, unit yang diuji dapat berupa fungsi validasi, pengolahan data, layanan komunikasi dengan API, serta komponen antarmuka. Pengujian setiap unit secara terpisah membantu pengembang mengetahui bagian

kode yang mengalami kesalahan sebelum digabungkan dengan fungsi lainnya. Penerapan *unit testing* secara bertahap juga mendukung proses pengembangan menggunakan metode Extreme Programming. Pengujian dapat dilakukan pada setiap iterasi agar kesalahan lebih cepat ditemukan, risiko masalah pada saat integrasi dapat dikurangi, serta proses perbaikan dan *refactoring* kode menjadi lebih mudah dilakukan (Hasibuan & Dirgahayu, 2020).

Pada penelitian ini, *unit testing* difokuskan pada fungsi-fungsi yang terdapat pada bagian *front end* Sistem Pelayanan Terpadu, seperti validasi formulir, pengolahan respons API, pembatasan akses berdasarkan peran pengguna, serta fungsi pendukung lainnya. Hasil pengujian digunakan untuk memastikan setiap fungsi berjalan sesuai dengan kebutuhan sebelum sistem diuji oleh pengguna.



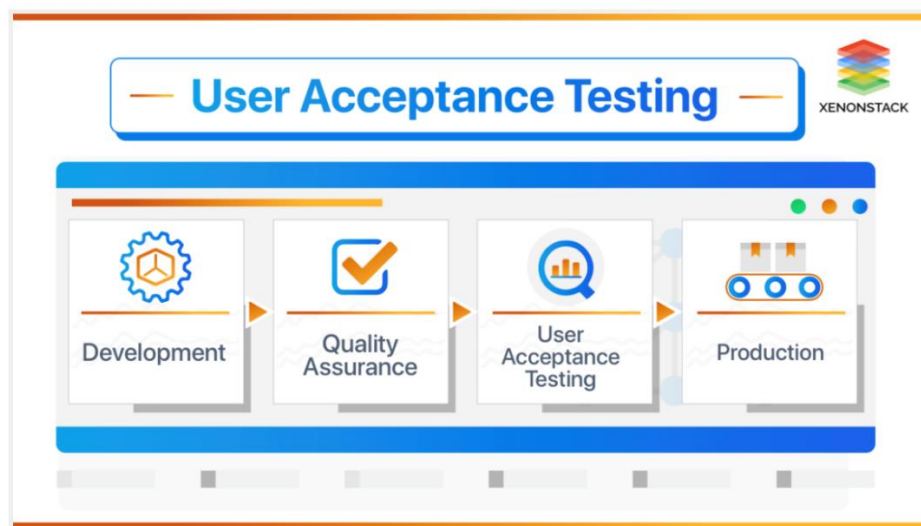
Gambar 2.7 Siklus Unit Testing

Pada penelitian ini, *unit testing* pada bagian *front end* dilakukan menggunakan Vitest sebagai *test runner*. Vitest digunakan untuk menjalankan pengujian terhadap fungsi dan komponen pada aplikasi ReactJS. Lingkungan jsdom digunakan untuk mensimulasikan lingkungan peramban, sedangkan React Testing Library digunakan untuk merender komponen serta mensimulasikan interaksi pengguna. Selain itu, `@testing-library/jest-dom/vitest` digunakan sebagai *matcher*

tambahan dalam proses pengujian. Penggunaan istilah *jest-dom* pada nama paket tersebut tidak menunjukkan bahwa *test runner* yang digunakan adalah Jest. Proses pengujian juga menerapkan *mocking* terhadap komunikasi API dan interaksi eksternal agar setiap unit dapat diuji secara terisolasi. V8 Coverage digunakan untuk mengetahui bagian kode yang telah dijalankan selama pengujian. Pada Sistem Pelayanan Terpadu PETA RANKO, pengujian difokuskan pada fungsi validasi, autentikasi, pembatasan akses berdasarkan peran, pengolahan respons API, pengelolaan dokumen, surat hasil, laporan, serta beberapa komponen antarmuka. Penerapan unit *testing* mendukung metode Extreme Programming karena pengujian dapat dilakukan secara bertahap pada setiap pengembangan fitur sehingga kesalahan dapat diketahui lebih awal dan proses *refactoring* menjadi lebih mudah dilakukan.

2.1.8 User Acceptance Testing (UAT)

User Acceptance *Testing* (UAT) merupakan pengujian yang dilakukan oleh pengguna akhir untuk memastikan bahwa sistem yang dikembangkan telah sesuai dengan kebutuhan dan proses pelayanan yang telah ditetapkan. Pengujian ini dilakukan setelah fitur sistem selesai dikembangkan dan siap digunakan oleh pengguna. Ilustrasi User Acceptance *Testing* ditunjukkan pada Gambar 2.8.



Gambar 2.8 User Acceptance *Testing*

UAT berfokus pada penerimaan pengguna terhadap fungsi sistem dalam kondisi penggunaan sebenarnya. Pengguna menjalankan beberapa skenario

pengujian berdasarkan peran masing-masing, kemudian menilai apakah hasil yang ditampilkan telah sesuai dengan hasil yang diharapkan. Melalui pengujian ini, dapat diketahui apakah fitur sistem dapat diterima atau masih memerlukan perbaikan sebelum digunakan secara menyeluruh (Saputra dkk., 2025).

Pada penelitian ini, UAT dilakukan dengan melibatkan pengguna yang mewakili masyarakat, petugas, dan kepala camat. Pengujian mencakup proses registrasi dan login, pengajuan layanan, pengunggahan dokumen, pemantauan status, verifikasi pengajuan, pengelolaan surat hasil, serta pemantauan laporan sesuai dengan hak akses masing-masing pengguna.

Hasil setiap skenario dibandingkan dengan kriteria penerimaan yang telah ditetapkan. Suatu fitur dinyatakan diterima apabila hasil aktual sesuai dengan hasil yang diharapkan, sedangkan fitur dinyatakan tidak diterima apabila terdapat ketidaksesuaian dan masih memerlukan perbaikan.

2.1.9 *Usability Testing*

Usability menggambarkan tingkat kemudahan suatu sistem ketika digunakan pengguna untuk mencapai tujuan secara efektif dan nyaman. Dalam pengembangan antarmuka, pengujian usability digunakan untuk menilai kemudahan pengguna dalam memahami alur, menyelesaikan aktivitas, dan berinteraksi dengan fitur yang tersedia. Penilaian ini membantu memastikan antarmuka dirancang sesuai kebutuhan pengguna (Mustada dkk., 2025).

Dalam konteks pelayanan administrasi berbasis *website*, *usability testing* digunakan untuk mengetahui tingkat kemudahan pengguna ketika mengakses informasi layanan, mengisi formulir pengajuan, mengunggah dokumen, dan memantau status pengajuan. Pengujian melibatkan pengguna yang mewakili masyarakat, petugas, dan kepala camat agar antarmuka yang dikembangkan dapat dinilai berdasarkan kebutuhan masing-masing pengguna (Mustada dkk., 2025).

Pada penelitian ini, pengukuran *usability* dilakukan menggunakan System Usability Scale (SUS). SUS merupakan kuesioner yang terdiri atas sepuluh pernyataan dengan skala Likert 1 sampai 5. Pernyataan dalam kuesioner terdiri atas pernyataan positif dan negatif yang digunakan untuk memperoleh gambaran mengenai kemudahan, kenyamanan, dan penerimaan pengguna terhadap sistem.

Perhitungan skor SUS dilakukan dengan mengurangi satu pada jawaban untuk pernyataan ganjil, sedangkan skor pernyataan genap diperoleh dengan mengurangi nilai jawaban dari lima. Seluruh skor kemudian dijumlahkan dan dikalikan 2,5 sehingga menghasilkan nilai akhir dalam rentang 0 sampai 100. Nilai tersebut digunakan untuk menginterpretasikan tingkat kemudahan penggunaan Sistem Pelayanan Terpadu Kantor Camat Rantau Kopar.

2.2 Penelitian Terdahulu

Syafaat (2018) melakukan penelitian berjudul “Metode Extreme Programming pada Pembangunan Web Aplikasi Seleksi Peserta Pelatihan Kerja”. Penelitian tersebut menggunakan metode Extreme Programming dalam proses pengembangan sistem. Teknologi yang digunakan meliputi PHP, HTML, CSS, JavaScript, dan basis data MySQL. Hasil penelitian berupa aplikasi berbasis website yang menyediakan informasi pelatihan, pendaftaran peserta, pelaksanaan ujian seleksi, verifikasi data, serta pengelolaan data peserta. Sistem tersebut membantu calon peserta memperoleh informasi dan mengikuti proses seleksi tanpa harus datang langsung ke Balai Latihan Kerja.

Dzurriati dkk. (2025) melakukan penelitian berjudul “Rancang Bangun Aplikasi Web Career Upgrade bagi Calon Profesional dengan Metode Extreme Programming”. Penelitian tersebut bertujuan mengintegrasikan fitur pencarian lowongan kerja, pengembangan Curriculum Vitae (CV), dan pelatihan workshop dalam satu aplikasi. Sistem dikembangkan menggunakan Next.js dan Firebase dengan menerapkan metode Extreme Programming. Pengujian dilakukan menggunakan black-box testing. Hasil pengujian menunjukkan bahwa seluruh fitur utama dapat berjalan dengan baik sehingga aplikasi dapat membantu calon profesional mencari pekerjaan, meningkatkan kualitas CV, dan mengikuti pelatihan secara lebih efisien.

Musyawir dkk. (2025) melakukan penelitian berjudul “Rancang Bangun Sistem Informasi Kependudukan dan Pelayanan Berbasis Web Menggunakan Model Extreme Programming di Desa Wewangriu”. Penelitian tersebut bertujuan mengatasi keterlambatan pelayanan administrasi yang disebabkan oleh keterbatasan kehadiran aparatur desa. Pengembangan sistem dilakukan melalui tahapan *planning*, *design*, *coding*, dan *testing* menggunakan CodeIgniter dan basis

data MySQL. Sistem diuji menggunakan *black-box testing* dan *usability testing*. Hasil pengujian menunjukkan bahwa sistem berjalan sesuai dengan kebutuhan pengguna dengan tingkat kepuasan sebesar 91%. Sistem tersebut membantu masyarakat memperoleh pelayanan kependudukan secara daring dan meningkatkan efisiensi pelayanan administrasi desa.

Mhd. Aldi Fathoni (2024) melakukan penelitian berjudul “Perancangan Sistem Informasi Pelayanan Publik Berbasis Web Menggunakan PHP MySQL di Dinas PUPR Provinsi Jambi”. Penelitian tersebut bertujuan mengatasi proses pelayanan pengaduan masyarakat yang masih dilakukan secara manual dan membutuhkan waktu yang cukup lama. Sistem dikembangkan menggunakan PHP dan basis data MySQL dengan metode Waterfall. Pengumpulan data dilakukan melalui observasi, wawancara, dan studi pustaka. Hasil penelitian berupa sistem pelayanan publik yang memungkinkan masyarakat menyampaikan pengaduan secara daring, sedangkan admin dapat melakukan verifikasi dan memberikan tanggapan melalui sistem. Sistem tersebut membantu meningkatkan efisiensi, transparansi, dan kecepatan pelayanan publik di lingkungan Dinas PUPR Provinsi Jambi.

Fika Anggraini dkk., (2024) melakukan penelitian berjudul “Rancangan dan Pembuatan Front End Website Kelurahan Tanjunganom dengan Menggunakan Metode Extreme Programming”. Penelitian tersebut bertujuan mengembangkan website yang informatif dan interaktif bagi masyarakat Kelurahan Tanjunganom. Sistem menyediakan informasi program kelurahan seperti Jimpitan, informasi layanan publik, serta fitur ulasan dari masyarakat. Pengembangan dilakukan menggunakan Laravel melalui tahapan perencanaan, perancangan, pengkodean, dan pengujian fungsional. Hasil penelitian menunjukkan bahwa website dapat membantu meningkatkan efisiensi pelayanan, keterbukaan informasi, dan partisipasi masyarakat terhadap program kelurahan.

Kategori	Peneliti 1	Peneliti 2	Peneliti 3	Peneliti 4	Peneliti 5
Penulis	Syafaat (2018)	Mhd. Aldi Fathoni (2024)	Dzurriati dkk. (2025)	Musyawir dkk. (2025)	Fika Anggraini dkk. (2024)
Fokus penelitian	Seleksi peserta	Pengembangan karier	Pelayanan kependudukan desa	Pelayanan pengaduan masyarakat	<i>Front end website</i> kelurahan

Kategori	Peneliti 1	Peneliti 2	Peneliti 3	Peneliti 4	Peneliti 5
	pelatihan kerja	calon profesional			
Metode	<i>Extreme Programming</i>	<i>Extreme Programming</i>	<i>Extreme Programming</i>	<i>Waterfal</i>	<i>Extreme Programming</i>
Target	Calon peserta dan admin Balai Latihan Kerja	Calon profesional atau pencari kerja	Masyarakat dan aparatur Desa Wewangriu	Masyarakat dan pegawai Dinas PUPR	Masyarakat dan admin Kelurahan Tanjunganom
Pengujian	Black Box Testing	Black Box Testing	Black-box testing dan usability testing	-	Functional testing
Hasil	Memperudah pendaftaran dan seleksi peserta secara daring	Seluruh fitur utama berjalan dengan baik dan membantu pengembangan karier	Sistem sesuai kebutuhan dengan tingkat kepuasan 91%	Membantu mempercepat dan meningkatkan transparansi pelayanan pengaduan	Meningkatkan keterbukaan informasi dan partisipasi masyarakat

Berdasarkan kajian terhadap beberapa penelitian terdahulu diketahui bahwa sebagian besar penelitian terdahulu menggunakan *website* untuk meningkatkan kemudahan akses informasi dan pelayanan. Beberapa penelitian juga menerapkan metode Extreme Programming karena mendukung pengembangan sistem secara bertahap dan fleksibel terhadap perubahan kebutuhan pengguna.

Perbedaan penelitian ini terletak pada fokus pengembangan *front end* Sistem Pelayanan Terpadu di Kantor Camat Rantau Kopar menggunakan ReactJS dan RESTful API. Sistem digunakan oleh masyarakat, petugas, dan kepala camat untuk mendukung proses pengajuan layanan, pemantauan status, verifikasi, pengelolaan surat hasil, dan laporan pelayanan. Pengujian dilakukan melalui *unit testing*, UAT, dan *usability testing*.