

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Tinjauan penelitian terdahulu dapat digunakan untuk memberi gambaran, ide, dan masukan terhadap penelitian yang dilakukan untuk menjadi salah satu acuan dalam melakukan tugas akhir. Terdapat lima penelitian yang akan digunakan sebagai pembanding dengan penelitian yang akan dilakukan, diantaranya sebagai berikut:

Penelitian yang pertama dilakukan oleh Aprilia dkk. (2022) yang berjudul "Rancang Bangun Antarmuka Situs Web Alumni dan *Tracer Study* di Perguruan Tinggi Menggunakan Metode *User Centered Design*". Penelitian ini dipublikasikan dalam *Journal of Science, Technology and Entrepreneur (JSTE)* Universitas Muhammadiyah Bandung, berhasil merancang dan membangun antarmuka situs web alumni dan *Tracer Study* dengan metode *User Centered Design*, yang mendapatkan nilai *usability* sebesar 76,25% dan mendapat respons positif dari pengguna.

Penelitian yang kedua dilakukan oleh Djutalov dkk. (2024) yang berjudul "*Agile Methodology in Web and Mobile Based App of Tracer Study and Career Center*". Dipublikasikan dalam jurnal *Brilliance: Research of Artificial Intelligence*, penelitian ini mengembangkan aplikasi *Tracer Study* dan *career center* berbasis web dan mobile menggunakan metodologi *Agile*. Pendekatan ini menekankan iterasi berkelanjutan dan keterlibatan pengguna untuk meningkatkan fungsionalitas dan pengalaman pengguna. Hasil menunjukkan peningkatan performa aplikasi dan kepuasan pengguna, meskipun masih diperlukan penyempurnaan antarmuka dan fitur berdasarkan umpan balik yang terus dikumpulkan.

Penelitian yang ketiga dilakukan oleh Fernandy dkk. (2023) yang berjudul "Rancang Bangun Sistem *Tracer Study* UNUSIA Berbasis Web Menggunakan Metode *Rapid Application Development*" mengembangkan sistem *Tracer Study* di Universitas Nahdlatul Ulama Indonesia menggunakan PHP, Laravel, dan metode *Rapid Application Development* dengan pendekatan UCD. Sistem ini memiliki fitur

survei alumni, manajemen data, analisis, dan pelaporan hasil *Tracer Study* untuk mendukung evaluasi pendidikan dan kebutuhan dunia kerja.

Penelitian yang keempat dilakukan oleh Rizki dkk. (2023) yang berjudul “Pengembangan Aplikasi *Tracer Study* Universitas Jabal Ghafur Menggunakan Framework Laravel”. Penelitian ini mengembangkan aplikasi *Tracer Study* Universitas Jabal Ghafur berbasis *website* menggunakan Laravel dan pendekatan *User Centered Design* (UCD). Sistem ini menggantikan metode manual seperti media sosial dan email dengan fitur *login*, berita, *Tracer Study*, kontak, pengumuman, serta laporan alumni. Aplikasi ini membantu kampus menelusuri lulusan dan mempermudah alumni mencari pekerjaan, dengan pengembangan efisien berkat arsitektur MVC Laravel.

Penelitian yang kelima dilakukan oleh Putri & Pakereng (2021) yang berjudul “Pengembangan Sistem Informasi *Tracer Study* Berbasis *User Centered Design* (UCD) Menggunakan Framework Laravel”. Penelitian ini mengembangkan sistem *Tracer Study* berbasis *website* menggunakan *framework* Laravel dengan pendekatan *User Centered Design*. Sistem yang dibuat menggantikan penggunaan Google Form yang sederhana dan kurang aman dengan menambahkan fitur *login* untuk pengamanan data. Sistem ini memudahkan FKIP UKSW dalam pengumpulan data alumni dan memungkinkan akses informasi *Tracer Study* tidak hanya oleh Wakil Dekan tetapi juga oleh pimpinan fakultas dan dosen, sehingga meningkatkan efektivitas pengelolaan data alumni.

Website Tracer Study SMAS Plus Taruna Andalan dikembangkan oleh Disya Putri Nazira dengan metode *User Centered Design* (UCD) bertujuan memusatkan data alumni secara terstruktur, menyajikan visualisasi melalui *dashboard*, serta mempermudah pendataan, pemantauan, dan pelaporan alumni di SMAS Plus Taruna Andalan. Sistem ini membantu sekolah mengelola data yang tersebar, mendukung evaluasi keberhasilan pendidikan, dan mengurangi kesalahan pencatatan data.

Tabel 2. 1 Penelitian Terdahulu

Penelitian	Jenis Perbandingan		
	Studi Kasus	Metode	Hasil
(Aprilia dkk., 2021)	<i>Tracer Study</i> Universitas Muhammadiyah Bandung	Metode <i>User Centered Design</i> (UCD)	Antarmuka situs web alumni dan <i>Tracer Study</i> berhasil dibangun dengan metode <i>User Centered Design</i> (UCD), dilengkapi fitur pengisian data alumni dan pengelolaan data oleh pihak kemahasiswaan, serta dinyatakan layak digunakan berdasarkan hasil uji <i>usability</i> dengan nilai 76,25%.
(Djutalov dkk., 2024)	<i>Tracer Study and Career Center</i> Universitas Pamulang	Metode <i>Agile</i>	Penelitian ini mengembangkan aplikasi <i>Tracer Study</i> dan <i>Career Center</i> berbasis web dan mobile di Universitas Pamulang menggunakan metode <i>Agile</i> untuk melacak karier alumni, mengumpulkan data peningkatan kurikulum, serta menyediakan informasi lowongan kerja, pelatihan, dan pengembangan karier, dengan pengembangan iteratif yang melibatkan pengguna sehingga meningkatkan performa dan kepuasan meskipun antarmuka dan fitur masih perlu disempurnakan.
(Fernandy dkk., 2023)	<i>Tracer Study</i> Universitas Nahdlatul Ulama Indonesia	Metode <i>Rapid Application Development</i>	Penelitian <i>Tracer Study</i> UNUSIA dengan metode RAD menghasilkan sistem web berbasis Laravel yang memudahkan pengumpulan data alumni valid, terintegrasi dengan pangkalan data Kemendikbud, menyediakan laporan sesuai kebutuhan pengguna, serta mendukung sinkronisasi dengan <i>database</i> e-campus, dengan saran pengembangan penambahan fitur grafis untuk visualisasi hasil survei.
(Rizki dkk., 2023)	<i>Tracer Study</i> Universitas Jabal Ghafur	Metode <i>User Centered</i>	Penelitian ini menghasilkan sistem <i>Tracer Study</i> berbasis web dengan pendekatan <i>User centered design</i> (UCD) yang memungkinkan

Penelitian	Jenis Perbandingan		
	Studi Kasus	Metode	Hasil
		<i>Design</i> (UCD)	akses fleksibel bagi alumni dan admin kampus, dilengkapi fitur pengisian serta penelusuran data untuk mendukung universitas memperoleh informasi alumni secara efektif dan efisien, sekaligus memastikan antarmuka sesuai kebutuhan dan kenyamanan pengguna.
(Putri & Pakereng., 2021)	<i>Tracer Study</i> FKIP UKSW	Metode <i>User Centered Design</i> (UCD)	Penelitian ini mengembangkan sistem <i>Tracer Study</i> berbasis web dengan Laravel dan pendekatan <i>User Centered Design</i> (UCD) untuk memudahkan pendataan alumni dan pengisian kuesioner <i>online</i> , dilengkapi fitur manajemen data yang telah diuji <i>blackbox</i> dan berjalan baik, sehingga mempercepat pengumpulan data akreditasi dan memudahkan alumni mengisi <i>Tracer Study</i> .
(Nazira Disya, 2026)	<i>Tracer Study</i> SMAS Plus Taruna Andalan	Metode <i>User Centered Design</i> (UCD)	Penelitian ini mengembangkan sistem <i>Tracer Study</i> berbasis web dengan React JS menggunakan pendekatan <i>User Centered Design</i> (UCD) untuk memastikan rancangan sesuai kebutuhan pengguna, serta dievaluasi dengan <i>User Experience Questionnaire</i> (UEQ) agar pengalaman pengguna terukur, sehingga pendataan alumni di SMAS Plus Taruna Andalan menjadi lebih terintegrasi dan mendukung akreditasi sekolah.

2.2 Landasan Teori

Landasan teori digunakan sebagai dasar untuk memahami dan menganalisis permasalahan dalam penelitian. Teori-teori yang digunakan diperoleh dari berbagai sumber yang relevan dan menjadi acuan dalam pengembangan sistem yang diteliti.

2.2.1 Sistem Tracer Study

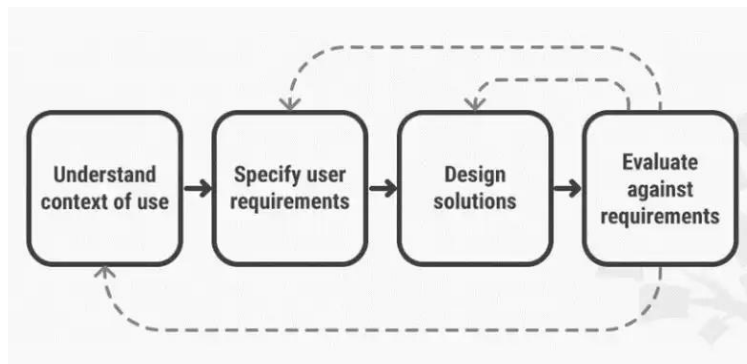
Menurut Tukino (2018), sistem merupakan suatu jaringan kerja yang terdiri dari berbagai elemen yang saling terhubung dan bekerja sama untuk mencapai tujuan tertentu. Sementara itu, menurut Efendi Erwan dkk. (2023), sistem dapat dipahami sebagai suatu kesatuan yang tersusun atas elemen, komponen, maupun prosedur yang saling berhubungan dan berfungsi secara teratur dalam rangka mencapai tujuan yang telah ditetapkan. Sistem tidak hanya terbatas pada aspek fisik berupa rangkaian aktivitas atau perangkat, tetapi juga mencakup aspek abstrak seperti konsep atau gagasan yang saling berkaitan. Dengan demikian, sistem memiliki peran penting sebagai jaringan kerja yang terintegrasi, di mana setiap elemen di dalamnya saling memengaruhi, sehingga gangguan pada satu bagian akan berdampak pada keseluruhan pencapaian tujuan.

Tracer Study adalah studi yang bertujuan untuk mengumpulkan informasi mengenai lulusan pendidikan tinggi, yang berperan penting dalam meningkatkan mutu pendidikan dan kualitas lulusan. Melalui *Tracer Study*, perguruan tinggi dapat mengetahui keberadaan terkini dari alumni serta menjaga komunikasi secara berkelanjutan, mengidentifikasi profil keterampilan dan kompetensi yang dimiliki, serta menganalisis relevansi kurikulum dengan kebutuhan dunia kerja. Selain itu, *Tracer Study* juga membantu memenuhi kriteria akreditasi, mengevaluasi jaringan komunikasi antara alumni dan *stakeholder*, serta mengetahui tingkat penyerapan kerja lulusan. Studi ini juga menelusuri kesesuaian antara kompetensi alumni dengan jabatan kerja di industri dan menghimpun data mengenai jumlah alumni yang berwirausaha. Informasi yang dihasilkan bukan hanya bermanfaat untuk sekolah dalam memperbaiki kurikulum dan proses pembelajaran, tetapi juga bagi siswa, alumni, dan dunia industri dalam memahami prospek kerja serta hubungan antara keilmuan dengan kebutuhan profesional di lapangan (Rooslyna Affandi dkk., 2025).

Selain sebagai alat evaluasi internal, *Tracer Study* juga menjadi salah satu syarat penting dalam akreditasi institusi pendidikan tinggi oleh BAN-PT. Melalui *Tracer Study*, institusi dapat mengukur efektivitas program akademik dan layanan pendidikan dari perspektif alumni, serta menjalin hubungan berkelanjutan antara kampus dan lulusan. *Tracer Study* juga bermanfaat dalam menyusun strategi peningkatan kualitas pembelajaran serta pelatihan tambahan yang sesuai dengan tuntutan pasar kerja. Dengan demikian, *Tracer Study* tidak hanya mendukung pengembangan institusi, tetapi juga membantu lulusan agar lebih siap bersaing secara profesional di dunia kerja yang terus berkembang (Rahmawati Nur dkk., 2022)

2.2.2 Metodologi User Centered Design (UCD)

Menurut Interaction Design Foundation (2020) *User Centered Design* (UCD) adalah proses desain yang dilakukan secara berulang-ulang di mana para desainer menempatkan fokus pada pengguna dan kebutuhan mereka di setiap tahap pengembangan. Dalam UCD, tim desain melibatkan pengguna secara aktif sepanjang proses dengan menggunakan berbagai metode riset dan teknik desain, guna menghasilkan produk yang sangat mudah digunakan dan dapat diakses oleh pengguna.



Gambar 2. 1 Tahap *User Centered Design*

Terdapat empat langkah utama dalam UCD sebagaimana ditunjukkan pada Gambar 2.1 (International Organization for Standardization (ISO), 2019) , yaitu:

Specify the Context of Use (Mengidentifikasi Konteks Penggunaan). Tahap ini bertujuan untuk memahami siapa pengguna sistem, apa tujuan penggunaannya, serta dalam kondisi dan lingkungan seperti apa sistem akan digunakan. Pada tahap ini dilakukan identifikasi karakteristik pengguna, tugas yang

akan dilakukan pengguna, serta lingkungan teknis maupun organisasional tempat sistem akan diterapkan. Pemahaman konteks ini menjadi dasar bagi pengembang untuk merancang solusi yang relevan dengan kondisi nyata di lapangan.

Specify User Requirements (Menentukan Kebutuhan Pengguna). Setelah konteks penggunaan dipahami, tahap selanjutnya adalah merumuskan kebutuhan pengguna secara lebih terstruktur. Kebutuhan ini diperoleh melalui proses pengumpulan data, seperti wawancara, survei, atau observasi, yang kemudian dianalisis untuk menghasilkan daftar kebutuhan fungsional maupun non-fungsional yang harus dipenuhi oleh sistem.

Produce Design Solutions (Merancang Solusi Desain). Pada tahap ini, kebutuhan yang telah dirumuskan diterjemahkan ke dalam bentuk rancangan desain, baik berupa alur proses, struktur sistem, maupun antarmuka pengguna. Rancangan ini biasanya dibuat secara bertahap mulai dari desain kasar (*low-fidelity*) hingga desain yang lebih matang (*high-fidelity*), sehingga dapat terus disesuaikan berdasarkan masukan yang diperoleh dari pengguna.

Evaluate Design (Mengevaluasi Desain). Tahap terakhir dilakukan untuk mengevaluasi apakah rancangan sistem yang telah dikembangkan telah memenuhi kebutuhan dan harapan pengguna. Evaluasi dilakukan dengan menggunakan *User Experience Questionnaire* (UEQ) yang diisi oleh pengguna setelah mencoba sistem. Hasil evaluasi digunakan untuk mengukur pengalaman pengguna pada beberapa aspek, seperti daya tarik, kejelasan, efisiensi, ketepatan, stimulasi, dan kebaruan. Apabila hasil evaluasi menunjukkan masih terdapat aspek yang perlu ditingkatkan, maka hasil tersebut menjadi dasar untuk melakukan perbaikan pada iterasi pengembangan berikutnya.

Keempat tahapan tersebut dilakukan secara iteratif, artinya proses dapat terus berulang hingga sistem yang dihasilkan benar-benar sesuai dengan kebutuhan dan harapan pengguna. Selain keempat tahapan tersebut, terdapat pula prinsip-prinsip utama yang mendasari penerapan UCD, meliputi fokus terhadap pengguna sebagai pertimbangan utama dalam setiap keputusan desain, desain yang menyatu secara keseluruhan antara fungsi dan antarmuka, pengujian berkelanjutan yang melibatkan pengguna pada setiap tahap pengembangan, serta proses desain yang bersifat

interaktif dan dapat disesuaikan berdasarkan umpan balik yang diperoleh (Yatana Saputri dkk., 2017)

Dalam penelitian ini, pendekatan UCD diterapkan melalui metode survei daring menggunakan kuesioner pada setiap tahapannya. Hasil survei tersebut digunakan untuk mengidentifikasi kebutuhan pengguna yang nantinya menjadi dasar dalam melakukan perancangan maupun perbaikan sistem pada tiap iterasi.

2.2.3 SMAS Plus Taruna Andalan

SMAS Plus Taruna Andalan adalah Sekolah Menengah Atas swasta yang berlokasi di Komplek Perumahan II PT Riau Andalan Pulp and Paper (RAPP), Kelurahan Pangkalan Kerinci Kota, Kecamatan Pangkalan Kerinci, Kabupaten Pelalawan, Provinsi Riau. Sekolah ini berdiri sejak 18 Juni 2008 berdasarkan SK Pendirian Nomor 421.4/Disdik/2008/358.1, berada di bawah naungan Kementerian Pendidikan dan Kebudayaan, serta dikelola oleh Yayasan Kerinci Citra Kasih (YKCK), yang merupakan yayasan pendidikan dari PT RAPP (Kementerian Pendidikan Dasar dan Menengah, 2025).

Menurut data resmi dari Kemdikbudristek melalui laman Dapodik (Data Pokok Pendidikan), SMAS Plus Taruna Andalan memiliki Nomor Pokok Sekolah Nasional (NPSN) 10496358 dan telah terakreditasi A. Sekolah ini dipimpin oleh kepala sekolah Yusli, S.Pd dan menerapkan Kurikulum Merdeka yang bertujuan untuk membentuk lulusan yang memiliki profil pelajar Pancasila serta siap bersaing di era global. Data Dapodik juga mencatat bahwa sekolah ini aktif menyelenggarakan kegiatan belajar dengan dukungan tenaga pendidik dan kependidikan yang profesional.

Dari sisi publikasi dan transparansi informasi, SMAS Plus Taruna Andalan memiliki situs resmi yang dapat diakses melalui www.smapta.sch.id. Melalui situs ini, sekolah menyajikan berbagai informasi penting seperti profil sekolah, program PPDB (Penerimaan Peserta Didik Baru), dokumentasi kegiatan sekolah, informasi prestasi siswa, kegiatan ekstrakurikuler, serta fitur interaktif yang melibatkan alumni. Situs ini juga menunjukkan bahwa sekolah mengadopsi pendekatan pembelajaran berbasis industri seperti Kaizen, 6S, serta pelatihan keterampilan abad ke-21 guna mempersiapkan siswa menghadapi tantangan global dan dunia kerja modern. Pendekatan ini menunjukkan komitmen sekolah dalam membangun

ekosistem pendidikan yang dinamis dan berorientasi masa depan. Dengan dukungan dari perusahaan induk dan yayasan yang berpengalaman dalam dunia industri, SMAS Plus Taruna Andalan menjadi salah satu sekolah swasta unggulan di Kabupaten Pelalawan yang terus berupaya meningkatkan mutu pendidikan melalui inovasi kurikulum, sistem informasi, serta pelibatan alumni.

2.2.4 User Interface

User Interface (UI) merupakan antarmuka yang menghubungkan pengguna dengan sistem komputer melalui tampilan grafis, elemen kontrol, serta struktur navigasi yang memudahkan pengguna dalam berinteraksi (Shneiderman dkk., 2016). UI tidak hanya sebatas layar, melainkan rangkaian komponen visual yang memungkinkan komunikasi dua arah antara pengguna dan sistem melalui proses *input* dan *output* yang jelas (Hasanudin & Adityawan, 2020). Oleh karena itu, UI berperan penting sebagai penghubung kebutuhan pengguna dengan hasil yang disediakan sistem (Satzinger dkk, 2012).

UI yang baik harus dirancang dengan memperhatikan kesederhanaan, konsistensi, dan estetika tampilan agar pengalaman pengguna semakin optimal (Auliazmi dkk., 2021). Pemilihan warna, ikon, dan elemen visual yang tepat dapat membuat interaksi lebih intuitif sekaligus meminimalkan kesalahan (Suteja & Harjoko, 2008). Selain itu, pemberian umpan balik yang jelas serta pemahaman karakteristik pengguna berperan dalam menciptakan antarmuka yang responsif dan mudah digunakan (Munthe dkk., 2018). Dengan demikian, kualitas UI menjadi faktor kunci dalam menunjang keberhasilan sistem, khususnya pada aplikasi *Tracer Study* yang menuntut kemudahan akses bagi alumni dan pihak sekolah.

2.2.5 User Experience

User Experience (UX) adalah keseluruhan pengalaman yang dirasakan pengguna saat berinteraksi dengan produk atau sistem digital, mencakup kemudahan penggunaan, kenyamanan, kepuasan, serta respon emosional yang muncul (ISO, 2009). UX tidak hanya menilai bagaimana sistem berfungsi, tetapi juga bagaimana sistem tersebut memberikan pengalaman yang bermakna dan menyenangkan bagi pengguna.

Kualitas UX ditentukan oleh aspek kemudahan, efisiensi, daya tarik visual, dan kejelasan interaksi, yang pada akhirnya memengaruhi persepsi subjektif pengguna terhadap sistem (Rizaldo, 2020). Dalam konteks sistem *Tracer Study*, UX menjadi krusial karena keberhasilan sistem sangat bergantung pada kenyamanan alumni saat mengisi kuesioner dan kemudahan pengelola dalam mengakses data. UX juga berkaitan erat dengan pendekatan *User Centered Design* (UCD), karena setiap tahapan perancangan dilakukan dengan melibatkan pengguna secara aktif sehingga hasil akhir benar-benar sesuai dengan kebutuhan dan ekspektasi mereka.

2.2.6 Website

Website merupakan sekumpulan halaman digital yang berisi berbagai jenis informasi seperti teks, gambar, animasi, suara, dan video, atau gabungan dari semuanya. Halaman-halaman ini saling terhubung melalui jaringan internet dan dapat diakses oleh siapa saja yang memiliki koneksi internet. Berdasarkan uraian tersebut, dapat disimpulkan bahwa *website* adalah media digital yang terdiri dari berbagai elemen informasi dan disimpan di server, sehingga dapat diakses secara *online* (Ganney, 2022).

Menurut Maulana (2024), *website* atau situs adalah sekumpulan halaman yang digunakan untuk menyajikan berbagai jenis berkas seperti teks, gambar (baik statis maupun bergerak), animasi, suara, atau kombinasi dari semuanya. Halaman-halaman ini dapat bersifat statis maupun dinamis dan saling terhubung membentuk suatu struktur yang terintegrasi melalui jaringan antarmuka.

2.2.7 Framework

Menurut Rony Setiawan (2021) *framework* merupakan suatu kerangka kerja yang dirancang untuk mempermudah proses pengembangan *website*. Penggunaan *framework* membantu para pengembang web dalam menyusun kode program secara lebih efisien, cepat, dan terorganisir dengan baik. Tujuan utama dari *framework* adalah untuk mempermudah *developer* dalam membangun sebuah situs web.

Menurut Satrio (2024), *framework* merupakan kumpulan instruksi yang disusun dalam bentuk kelas dan fungsi dengan tujuan tertentu untuk mempermudah pengembang dalam menyelesaikan proyek. *Framework* menyediakan berbagai

library atau kode sumber yang siap digunakan, sehingga pengembang tidak perlu menulis sintaks yang sama secara berulang. Penggunaan *framework* membantu menghemat waktu dan meningkatkan efisiensi dalam proses pengembangan.

2.2.8 React JS

React JS adalah *library* JavaScript yang dikembangkan oleh Facebook untuk membangun antarmuka pengguna (*User interface/UI*) aplikasi web dan mobile secara interaktif dan efisien. React menggunakan konsep komponen yang merupakan unit terkecil yang dapat digabungkan untuk membentuk UI yang kompleks, memungkinkan perubahan UI secara otomatis tanpa perlu menyegarkan halaman secara keseluruhan. React juga mendukung berbagai teknologi seperti *server-side rendering* dan Virtual DOM yang meningkatkan kinerja aplikasi (Facebook, 2024).

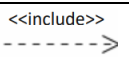
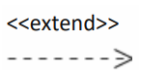
React bukanlah *framework* penuh, melainkan *library* yang fokus pada rendering komponen UI. React menggunakan JSX untuk menulis kode HTML langsung di dalam JavaScript, serta menerapkan aliran data satu arah dan pengelolaan state agar UI responsif terhadap perubahan data (W3Schools, 2024). Keunggulannya terletak pada kemudahan penggunaan, modularitas komponen, serta efisiensi rendering melalui Virtual DOM yang hanya memperbarui bagian UI yang berubah. Struktur ini mendukung performa aplikasi dan semakin ditingkatkan dengan React Fiber yang membuat proses rendering lebih mulus (Mozilla Developer Network, 2024).

2.2.9 Use Case Diagram

Menurut Malius dan Ali Hakam Dani (2021), *Use Case Diagram* adalah alat pemodelan yang digunakan untuk menggambarkan perilaku sistem informasi yang akan dibangun. Diagram ini menunjukkan bagaimana satu atau lebih aktor berinteraksi dengan sistem, sehingga mempermudah pemahaman mengenai cara pengguna menggunakan sistem tersebut. Melalui diagram ini, pengembang dapat mengenali fungsi-fungsi utama dalam sistem serta menetapkan hak akses masing-masing aktor terhadap fungsi tersebut. Selain itu, *Use Case Diagram* juga membantu dalam menentukan batasan sistem, sehingga mempermudah proses analisis dan perancangan.

Menurut Fadli (2024), *Use Case* merupakan representasi dari fungsionalitas yang diharapkan dari suatu sistem dan menggambarkan interaksi antara aktor dengan sistem tersebut. Dalam *Use Case*, aktor diartikan sebagai entitas, baik berupa manusia maupun sistem lain, yang berperan dalam menjalankan tugas atau aktivitas di dalam sistem. Penjelasan lebih lanjut mengenai hal ini dapat dilihat pada Tabel 2.2 berikut:

Tabel 2. 2 *Use Case Diagram*

No	Nama	Gambar	Keterangan
1.	<i>Actor</i>		Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2.	<i>Dependency</i>		Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang tidak mandiri (<i>dependent</i>)
3.	<i>Generalization</i>		Hubungan dimana objek anak (<i>descendent</i>) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
4.	<i>Include</i>		Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
5.	<i>Extend</i>		Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku <i>use case</i> sumber pada suatu titik yang diberikan.
6.	<i>Association</i>		Apa yang menghubungkan antara objek satu dengan objek lainnya
7.	<i>System</i>		Menspesifikasikan data paket yang menampilkan sistem secara terbatas
8.	<i>Use case</i>		Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

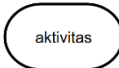
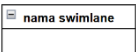
Tabel di atas menjelaskan simbol-simbol yang digunakan dalam perancangan *use case diagram* pada penelitian ini, mulai dari *actor* yang merepresentasikan pengguna sistem, hingga relasi *include*, *extend*, *generalization*, dan *dependency* yang menggambarkan hubungan antar *use case*. Simbol-simbol ini menjadi acuan dalam menyusun *use case diagram* sistem *Tracer Study SMAS Plus Taruna Andalan* pada sub-bab berikutnya.

2.2.10 Activity Diagram

Menurut Hamdi Kurniawan (2020), *activity diagram* merupakan bentuk pemodelan sistem yang menggambarkan alur aktivitas yang terjadi di dalam sistem. Diagram ini digunakan untuk menjelaskan proses atau kegiatan dalam program tanpa bergantung pada kode program maupun tampilan antarmuka.

Activity diagram digunakan untuk menggambarkan alur kerja atau aktivitas dari suatu sistem, proses bisnis, atau menu dalam perangkat lunak. Diagram ini menampilkan aktivitas yang dilakukan oleh sistem, bukan oleh aktor (Musthofa dkk, 2022). Selain itu, *Activity diagram* berfungsi untuk memodelkan proses yang berjalan secara paralel, bercabang, maupun tersinkronisasi. Diagram ini sangat berguna dalam memvisualisasikan urutan atau simultanitas langkah-langkah logis, serta membantu mengidentifikasi kemungkinan hambatan atau ketidakefisienan dalam proses bisnis atau sistem perangkat lunak. Seperti yang terdapat dalam Tabel 2.3 dibawah ini:

Tabel 2. 3 Activity Diagram

No	Nama	Simbol	Keterangan
1.	Status awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2.	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.	Percabangan / <i>decision</i>		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4.	Penggabungan / <i>join</i>		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5.	Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
6.	<i>Swimlane</i>		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Tabel di atas menjelaskan simbol-simbol yang digunakan dalam perancangan *activity diagram* pada penelitian ini, mulai dari status awal dan status akhir yang menandai batas alur aktivitas, aktivitas itu sendiri, percabangan (*decision*) dan penggabungan (*join*) yang menggambarkan alur logika sistem, hingga *swimlane* yang memisahkan tanggung jawab tiap aktor dalam suatu proses. Simbol-simbol ini menjadi acuan dalam menyusun *activity diagram* sistem *Tracer Study SMAS Plus Taruna Andalan* pada sub-bab berikutnya.

2.2.11 Entity Relationship Diagram (ERD)

Entity relationship diagram (ERD) merupakan alat bantu visual yang digunakan untuk memodelkan struktur dan hubungan antar entitas dalam perancangan basis data (Larassati dkk., 2019). Menurut Sari dkk. (2020), ERD adalah model yang menjelaskan keterkaitan antar data dalam *database* berdasarkan objek-objek dasar yang saling memiliki relasi. Dengan demikian, ERD dapat disimpulkan sebagai alat penting yang digunakan oleh analis sistem untuk merancang *database* secara sistematis, efisien, dan sesuai kebutuhan pengembangan sistem.

ERD mempermudah visualisasi elemen-elemen data dan keterhubungannya, sehingga mendukung proses analisis dan dokumentasi desain sistem. Representasi grafis yang jelas dalam ERD dapat mengurangi risiko kesalahan dalam desain dan membantu tim pengembang serta pemangku kepentingan lainnya dalam memahami struktur *database* yang dirancang.

Diagram ini terdiri dari komponen utama seperti entitas, atribut, dan relasi antar entitas. Entitas merepresentasikan objek atau konsep yang memiliki data, sedangkan atribut menggambarkan karakteristik dari entitas tersebut. Sementara itu, relasi menunjukkan bagaimana entitas saling terhubung dan berinteraksi dalam sistem. Pemahaman terhadap hubungan-hubungan ini memungkinkan pengembang menghindari redundansi data, serta meningkatkan efisiensi penyimpanan dan pengolahan data. Selain itu, ERD juga menjadi fondasi dalam perancangan skema *database* relasional yang nantinya akan diterapkan dalam bentuk tabel dan relasi pada sistem manajemen basis data (DBMS).

2.2.12 MySQL

Basis data (*database*) merupakan komponen utama yang digunakan oleh pengembang perangkat lunak untuk menyimpan dan mengatur data secara efisien. Dalam praktiknya, pengelolaan data sering kali melibatkan penggunaan sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*), dan salah satu RDBMS yang paling populer adalah MySQL. MySQL adalah perangkat lunak *database open-source* yang menyediakan kemudahan dalam pengelolaan data serta banyak digunakan dalam pengembangan aplikasi berbasis web (Munawar dkk., 2021).

Sebagai salah satu jenis RDBMS, MySQL memungkinkan pengguna untuk membuat, mengelola, dan memanfaatkan data berdasarkan model relasional. Sistem ini dirancang untuk mendukung kebutuhan pengembang dari berbagai tingkat keahlian, baik pemula maupun profesional, dengan performa yang handal dan stabil. Dalam proses pengelolaan datanya, MySQL menggunakan bahasa SQL (*Structured Query Language*), yang merupakan standar bahasa untuk mengakses dan memanipulasi data pada sistem basis data (Lifan, 2024).

Salah satu keunggulan utama dari MySQL adalah penggunaan SQL sebagai bahasa utama untuk pengelolaan data. SQL dirancang khusus untuk mengakses, mengelola, dan memanipulasi data dalam struktur berbasis tabel yang terdiri atas kolom dan baris. Pendekatan berbasis relasi ini memungkinkan data untuk dihubungkan secara sistematis, sehingga mempermudah proses pengambilan, pengolahan, dan penyimpanan informasi. Dengan keunggulan tersebut, MySQL menjadi pilihan utama dalam berbagai proyek pengembangan aplikasi, terutama yang membutuhkan sistem penyimpanan data yang andal, efisien, serta mudah diintegrasikan dengan berbagai bahasa pemrograman (Kadim dkk., 2023)

2.2.13 *Black Box Testing*

Black Box Testing merupakan salah satu metode pengujian perangkat lunak yang paling banyak digunakan. Pendekatan ini menitikberatkan pada pengujian fungsi sistem berdasarkan spesifikasi yang ditetapkan, tanpa memperhatikan bagaimana struktur internalnya dibangun. Dalam praktiknya, penguji tidak perlu memahami detail implementasi, melainkan cukup memastikan bahwa sistem berjalan sesuai dengan harapan. Teknik ini sangat efektif untuk menemukan berbagai jenis permasalahan, mulai dari fungsi yang tidak beroperasi atau hilang, kesalahan pada antarmuka pengguna, kekeliruan dalam struktur maupun akses basis data, gangguan performa, hingga masalah yang muncul pada proses inisialisasi maupun terminasi (Devyanti dkk., 2025).

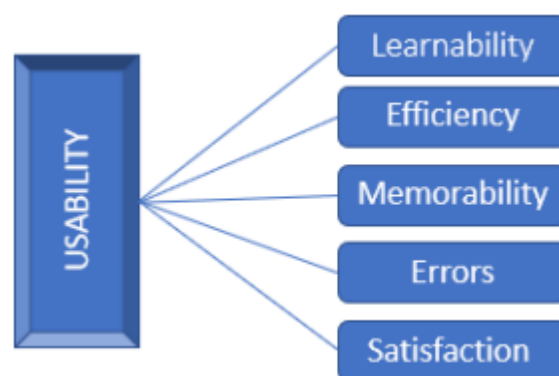
Automation testing merupakan metode pengujian perangkat lunak yang memanfaatkan perangkat lunak khusus untuk menjalankan skrip otomatis sehingga berbagai skenario uji dapat dilakukan secara berulang tanpa intervensi manual. Melalui pendekatan ini, tugas-tugas yang sebelumnya dikerjakan oleh penguji secara manual dapat digantikan oleh sistem, sehingga proses pengujian menjadi

lebih cepat, efisien, dan konsisten. Saat ini, banyak proyek pengembangan *software* telah menerapkan *automation testing* sejak tahap awal hingga akhir siklus pengembangan dengan tujuan utama meningkatkan efisiensi, mempercepat waktu rilis, serta meminimalkan potensi kesalahan manusia. Beragam *software* atau *tools* dapat dimanfaatkan sesuai kebutuhan, mulai dari program sederhana hingga aplikasi khusus yang dirancang untuk mendukung pengujian otomatis, sehingga hasil pengujian lebih andal dan dapat membantu memastikan kualitas produk perangkat lunak secara menyeluruh (Sugiarti Ucy, 2024).

Cypress merupakan sebuah *tool* yang digunakan untuk mempermudah proses pengujian sekaligus *debugging* pada aplikasi web. Dengan Cypress, dapat dilakukan berbagai jenis pengujian seperti *end-to-end test* maupun *component test*. Alat ini umumnya dipakai untuk menguji aplikasi berbasis web, misalnya yang dikembangkan dengan *React* atau *Angular*, karena seluruh proses pengujian dijalankan langsung di dalam peramban (*web browser*). Sebagai bagian dari ekosistem *Node.js*, Cypress dapat diinstal melalui *npm* atau *yarn*, maupun diunduh langsung dari situs resminya (Nosuke, 2023).

2.2.14 Usability Testing

Dalam interaksi antara manusia dan komputer, terdapat konsep penting yang dikenal sebagai *usability* atau kegunaan. *Usability* menggambarkan sejauh mana suatu sistem dapat digunakan secara mudah, efisien, aman, mudah dipelajari, mudah diingat, serta memiliki nilai guna yang tinggi bagi penggunanya. Prinsip ini sangat penting diterapkan dalam sistem *tracer study* agar pengguna (alumni maupun admin) dapat merasakan manfaat maksimal dan mendukung keberhasilan implementasi sistem tersebut.



Gambar 2. 2 Usability Testing

Usability terdiri dari lima komponen utama yang digunakan untuk mengukur kemudahan dan kenyamanan pengguna dalam menggunakan suatu sistem. Kelima komponen tersebut meliputi *learnability*, *efficiency*, *memorability*, *errors*, dan *satisfaction*, sebagaimana ditunjukkan pada Gambar 2.2 (Pambudi & Yuni Arini, 2018). Adapun penjelasan dari masing-masing komponen tersebut adalah sebagai berikut:

1. *Learnability* (Kemudahan untuk Dipelajari)
Menunjukkan seberapa mudah pengguna memahami dan menggunakan sistem saat pertama kali mengaksesnya.
2. *Efficiency* (Efisiensi Penggunaan)
Menggambarkan seberapa cepat pengguna dapat menyelesaikan tugas setelah memahami cara kerja sistem.
3. *Memorability* (Kemudahan untuk Diingat)
Mengacu pada kemampuan pengguna untuk kembali menggunakan sistem dengan lancar setelah tidak mengaksesnya dalam jangka waktu tertentu.
4. *Errors* (Tingkat Kesalahan)
Berkaitan dengan seberapa banyak kesalahan yang dilakukan oleh pengguna, tingkat keparahan kesalahan tersebut, dan kemudahan dalam mengatasinya.
5. *Satisfaction* (Kepuasan Pengguna)
Merujuk pada tingkat kenyamanan dan kepuasan yang dirasakan pengguna selama menggunakan sistem.

Dengan menerapkan lima aspek tersebut, sistem *tracer study* dapat memberikan pengalaman penggunaan yang optimal bagi penggunanya, sehingga proses pengumpulan data alumni dapat berjalan secara efektif dan efisien.

Pengujian *usability* pada penelitian ini dilakukan dengan menggunakan skala *Likert*, yaitu skala yang digunakan untuk mengukur sikap, pendapat, atau persepsi seseorang terhadap suatu objek atau fenomena tertentu (Sugiyono, 2022). Skala *Likert* dalam penelitian ini terdiri dari lima pilihan jawaban, yaitu Sangat Setuju (SS), Setuju (S), Netral (N), Tidak Setuju (TS), dan Sangat Tidak Setuju (STS), dengan bobot nilai masing-masing 5, 4, 3, 2, dan 1. Khusus untuk pernyataan yang bersifat negatif, bobot nilai dibalik sehingga jawaban TS dan STS justru memiliki

nilai yang lebih tinggi, karena responden yang tidak setuju terhadap pernyataan negatif menunjukkan penilaian yang positif terhadap sistem.

Untuk mengetahui tingkat kelayakan *usability* suatu sistem, skor yang diperoleh dari jawaban responden dihitung menggunakan rumus berikut:

$$P = \frac{\text{Skor total}}{\text{Skor Ideal}} \times 100\% = 100\% \quad 2-1$$

Keterangan:

- Skor total = Jumlah nilai riil dari jawaban responden, (SS×5) + (S×4) + (N×3) + (TS×2) + (STS×1)
- Skor ideal = skor tertinggi yang mungkin dicapai apabila seluruh responden memberikan jawaban Sangat Setuju (SS), dihitung dengan mengalikan bobot tertinggi (5) dengan jumlah responden
- P = Persentase/indeks kelayakan pernyataan tersebut, Hasil akhir dalam bentuk persen (%)

Untuk menentukan tingkat kelayakan *usability* suatu sistem, hasil perhitungan skor dari setiap aspek diinterpretasikan ke dalam bentuk persentase dan dikategorikan berdasarkan rentang nilai sebagaimana disajikan pada Tabel 2.4 berikut.

Tabel 2. 4 Kategori Penilaian *Usability Testing*

Persentase	Kategori	Keterangan
0% – 20%	Sangat Buruk	Sistem tidak memenuhi standar yang diharapkan
21% – 40%	Buruk	Hasil di bawah rata-rata
41% – 60%	Cukup	Hasil memenuhi standar minimum
61-80%	Baik	Sistem memiliki kualitas yang memadai
81%-100%	Sangat Baik	Sistem telah memenuhi standar dengan sangat baik dan memberikan hasil yang memuaskan

Berdasarkan Tabel 2.4 (Agung Sasongko, 2025) tersebut, dapat dilihat bahwa penilaian tingkat kelayakan *usability* suatu sistem terbagi ke dalam lima kategori berdasarkan rentang persentase skor yang diperoleh. Semakin tinggi persentase yang dihasilkan, semakin baik pula tingkat kelayakan *usability* sistem tersebut. Kategori ini digunakan sebagai acuan dalam menginterpretasikan hasil perhitungan skor pada setiap aspek pengujian *usability* dalam penelitian ini, sehingga hasil yang diperoleh dapat dikategorikan secara objektif ke dalam salah satu dari lima tingkatan tersebut, mulai dari "Sangat Buruk" hingga "Sangat Baik".

2.2.15 User Experience Questionnaire (UEQ)

User experience questionnaire (UEQ) merupakan instrumen pengukuran kuantitatif yang dirancang untuk menilai pengalaman pengguna terhadap suatu produk atau sistem interaktif. Instrumen ini memuat enam dimensi utama, yaitu *Attractiveness* (daya tarik), *Perspiciuity* (kejelasan), *Efficiency* (efisiensi), *Dependability* (keandalan), *Stimulation* (stimulasi), dan *Novelty* (kebaruan). Masing-masing dimensi diukur menggunakan pasangan kata bipolar pada skala *Likert* tujuh tingkat, sehingga pengguna diminta untuk memberikan penilaian yang mencerminkan persepsi mereka secara subjektif. Melalui pendekatan ini, UEQ tidak hanya mengukur aspek pragmatis seperti kemudahan dan kecepatan penggunaan, tetapi juga aspek hedonis yang mencakup kesenangan, motivasi, dan inovasi dari produk yang digunakan (Schrepp dkk, 2017)

Metode UEQ diawali dengan penyusunan kuesioner berisi 26 item pasangan kata berlawanan, seperti “mudah digunakan – sulit digunakan” atau “membosankan – menarik”. Skor diberikan pada rentang -3 (sangat negatif) hingga $+3$ (sangat positif), kemudian diolah untuk mendapatkan nilai rata-rata pada tiap dimensi. Hasil ini dibandingkan dengan *Benchmark* UEQ internasional yang dibangun dari ribuan dataset, sehingga dapat diketahui posisi performa sistem yang diuji terhadap standar umum (Schrepp dkk., 2017)

Keunggulan UEQ terletak pada kemampuannya memberikan gambaran menyeluruh tentang kualitas pengalaman pengguna secara cepat dan komprehensif. Analisis dapat dilakukan dengan *UEQ Data Analysis Tool* yang mempermudah perhitungan skor dan menyajikan hasil dalam bentuk visualisasi diagram. Metode ini telah terbukti efektif di berbagai konteks, mulai dari sistem pembelajaran daring hingga aplikasi transportasi, karena mampu mengidentifikasi kelebihan maupun kelemahan suatu sistem (Schrepp dkk., 2017)

Dalam *User Experience Questionnaire* terdapat 6 skala yang digunakan sebagai acuan, yaitu:

1. *Attractiveness*, kesan secara keseluruhan yang dirasakan pengguna terhadap produk.
2. *Perspiciuity*, kesempurnaan yang dirasakan ketika menggunakan produk.
3. *Efficiency*, interaksi yang efisien dan cepat antara pengguna dan produk.

4. *Dependability*, kenyamanan saat menggunakan produk.
5. *Stimulation*, keberhasilan dan motivasi yang diperoleh saat menggunakan produk.
6. *Novelty*, peningkatan dan eksplorasi inovasi dalam produk.

Rentang penilaian untuk *Benchmark Intervals* dapat dilihat pada Tabel 2.5 berikut:

Tabel 2. 5 *Benchmark Intervals* untuk Skala UEQ

	<i>Attractiveness</i>	<i>Perspicuity</i>	<i>Efficiency</i>	<i>Dependability</i>	<i>Stimulation</i>	<i>Novelty</i>
<i>Excellent</i>	$\geq 1,84$	$\geq 2,00$	$\geq 1,88$	$\geq 1,70$	$\geq 1,70$	$\geq 1,60$
<i>Good</i>	$\geq 1,58$ $< 1,84$	$\geq 1,73$ $< 2,00$	$\geq 1,50$ $< 1,88$	$\geq 1,48$ $< 1,70$	$\geq 1,35$ $< 1,70$	$\geq 1,12$ $< 1,60$
<i>Above average</i>	$\geq 1,18$ $< 1,58$	$\geq 1,20$ $< 1,73$	$\geq 1,05$ $< 1,50$	$\geq 1,14$ $< 1,48$	$\geq 1,00$ $< 1,35$	$\geq 0,70$ $< 1,12$
<i>Below average</i>	$\geq 0,69$ $< 1,18$	$\geq 0,72$ $< 1,20$	$\geq 0,60$ $< 1,05$	$\geq 0,78$ $< 1,14$	$\geq 0,5$ $< 1,00$	$\geq 0,16$ $< 0,70$
<i>Bad</i>	$< 0,69$	$< 0,72$	$< 0,60$	$< 0,78$	$< 0,50$	$< 0,16$

Dalam penelitian ini, UEQ dipilih karena selaras dengan pendekatan UCD yang berfokus pada pengalaman pengguna. Melalui enam dimensinya, UEQ mampu mengukur kualitas UI dan UX sistem *Tracer Study* secara objektif agar memberikan pengalaman positif bagi alumni dan admin.

2.2.16 Cypress JS

Pengujian perangkat lunak merupakan proses yang dilakukan untuk memastikan setiap fitur sistem berjalan sesuai spesifikasi yang ditetapkan serta menghasilkan produk dengan kualitas yang tinggi. Salah satu pendekatan yang digunakan adalah pengujian otomatis (*automation testing*), yaitu metode pengujian yang memanfaatkan framework atau alat bantu untuk mengeksekusi skenario pengujian secara otomatis, konsisten, dan cepat tanpa dipengaruhi faktor kelelahan maupun kesalahan manusia (Septiani dkk., 2025). Salah satu *framework* yang mendukung pendekatan ini adalah Cypress, yakni *framework* berbasis JavaScript yang mampu menjalankan pengujian end-to-end secara *real-time* dengan performa ringan dan integrasi CI/CD.

Dalam pengembangan sistem informasi berbasis web seperti *website Tracer Study*, pengujian otomatis menjadi sangat relevan untuk memastikan seluruh alur fungsionalitas sistem berjalan sesuai kebutuhan pengguna, mulai dari proses *login*, pengisian data alumni, hingga pengelolaan data oleh administrator. Pendekatan

end-to-end testing menggunakan Cypress memungkinkan setiap alur tersebut diverifikasi secara menyeluruh dan berulang, terutama ketika terjadi perubahan atau penambahan fitur pada sistem (Septiani dkk., 2025). Dengan demikian, penerapan *automation testing* pada penelitian ini bertujuan untuk menjamin kualitas fungsional sistem sebelum digunakan oleh pemangku kepentingan, termasuk alumni, guru BK, dan administrator.