

KATA PENGANTAR

Bismillahirrahmannirrahim

Dengan mengucapkan Alhamdulillah segala puji syukur peneliti panjatkan kehadiran Allah SWT, yang telah memberikan rahmat dan hidayah-Nya serta kenikmatan-kenikmatan yang tidak pernah putus kepada peneliti, sehingga peneliti dapat menyelesaikan penyusunan tugas akhir yang berjudul “Pengembangan Aplikasi Manajemen Aset Kesehatan Berbasis *Website* Menggunakan *Low Code Platform*”. Guna memenuhi salah satu syarat wajib dalam menyelesaikan pendidikan pada Program Studi Sistem Informasi Universitas Telkom.

Perjalanan panjang yang penuh tantangan telah peneliti lalui dalam menyelesaikan penulisan tugas akhir ini. Namun, berkat kehendak-Nya peneliti berhasil menyelesaikan penyusunan tugas akhir ini yang mana bukan hanya sebagai syarat wajib dalam menyelesaikan Pendidikan tetapi juga tugas akhir ini memberikan banyak manfaat terkhusus kepada peneliti yang sebelumnya tidak dapatkan selama bangku perkuliahan.

Selesainya penyusunan tugas akhir ini tentunya bukan hanya peneliti yang terlibat. Akan tetapi, banyak sekali pihak-pihak lain yang terlibat membantu peneliti dalam menyelesaikan penyusunan tugas akhir ini, patutlah kiranya peneliti mengucapkan terima kasih kepada :

1. Ibu Nia Ambarsari, S.Si., M.T. selaku dosen wali dan pembimbing I yang telah meluangkan waktunya untuk memberikan bimbingan, arahan dan dukungan selama peneliti melakukan pembelajaran di Telkom University.
2. Bapak Sinung Suakanto, S.T., M.T. selaku pembimbing II yang telah bersedia membimbing peneliti dengan memberikan arahan dan saran selama penelitian berlangsung.
3. Bapak Mochammad Harris Sanjaya selaku user dan Rumah Sakit Permata Dalima yang telah memberikan waktu, tenaga dan tempat untuk membantu peneliti dalam menyelesaikan tugas akhir ini.

4. Kedua orang tua peneliti yang selalu memberikan doa, nasihat, motivasi, dan materi serta kesabaran yang luar biasa bagi peneliti dalam seluruh perjalanan kehidupan peneliti.
5. Trisye Amedia selaku pasangan peneliti yang selalu membantu, memberikan semangat dan kesabarannya selama menemani peneliti menyelesaikan tugas akhir.
6. Vasya, Arya dan Nisa selaku teman dan rekan kerja peneliti selama melakukan *internship* yang telah membantu peneliti dalam *brainstorming idea* untuk tugas akhir ini.
7. Komunitas Orang Tangerang Telkom yang telah menjadi rumah bagi peneliti.
8. Warga F's Server yang telah menemani dan menyemangati peneliti dalam menyelesaikan tugas akhir ini.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem manajemen aset di rumah sakit merupakan salah satu area yang sangat membutuhkan penerapan teknologi yang dapat meningkatkan efisiensi dan aksesibilitas. Rumah Sakit Permata Dalima menghadapi tantangan dalam pengelolaan aset yang belum terintegrasi dan sulit diakses oleh pengguna di luar lingkungan rumah sakit. Saat ini, sistem manajemen aset di rumah sakit tersebut hanya dapat diakses secara lokal, yang menghambat fleksibilitas dalam pelacakan dan pemeliharaan aset secara *real-time* (Asih dkk., 2018). Selain itu, sistem yang ada belum mendukung integrasi dengan sistem lain di rumah sakit, yang semakin memperparah inefisiensi operasional.

Pada penelitian oleh (Suakanto et al., 2023) dalam pengelolaan aset, khususnya di sektor kesehatan, manajemen yang efektif memainkan peran penting dalam meningkatkan efisiensi operasional dan kualitas pelayanan. Rumah sakit, sebagai salah satu pilar utama dalam layanan kesehatan, sering menghadapi tantangan terkait pengelolaan aset yang kurang optimal, seperti lamanya waktu pencarian alat, aset yang hilang, serta kurangnya data yang akurat dan real-time. Hal ini dapat berdampak langsung pada penurunan kualitas layanan, peningkatan keluhan, dan bahkan risiko fatal terhadap keselamatan pasien, seperti kesalahan penggunaan alat atau ketidakterediaan alat medis yang dibutuhkan pada saat kritis.

Pada penelitian oleh (Bock & Frank, 2021) disebutkan bahwa perkembangan teknologi dalam beberapa dekade terakhir telah mengubah cara pengelolaan aset dilakukan, terutama dengan adanya solusi digital. Salah satu pendekatan yang semakin populer dalam pengembangan aplikasi adalah penggunaan *low-code platform*. *Low-code programming* memungkinkan pengembangan perangkat lunak dengan menggunakan antarmuka visual dan komponen *drag-and-drop*, yang mengurangi kebutuhan akan penulisan kode

manual secara mendalam. Pendekatan ini memberikan kemudahan bagi pengembang, terutama dalam mempercepat proses pengembangan aplikasi.

Teknologi *low-code* menawarkan solusi potensial untuk mengatasi masalah ini melalui pengembangan aplikasi berbasis web yang terintegrasi dan dapat diakses dari mana saja. Menurut penelitian oleh (Saputra dkk., 2020), penerapan sistem ERP berbasis *low-code* di rumah sakit mampu meningkatkan efisiensi operasional, mempercepat proses pengambilan keputusan, dan mempermudah pemeliharaan serta pelacakan aset. Selain itu, platform *low-code* juga memungkinkan pengguna untuk menyesuaikan aplikasi dengan mudah tanpa perlu keahlian teknis tinggi, yang semakin mempercepat proses implementasi dan pengembangan.

Studi sebelumnya oleh (Ganesh dkk., 2014) menekankan pentingnya pengujian penerimaan pengguna (*User Acceptance Test*, UAT) untuk memastikan bahwa sistem yang dikembangkan benar-benar memenuhi kebutuhan pengguna. Pengujian ini penting dalam konteks rumah sakit, di mana keberhasilan implementasi sistem manajemen aset sangat bergantung pada kemudahan penggunaan dan keandalan aplikasi oleh staf rumah sakit.

Rumah Sakit Permata Dalima perlu mengembangkan aplikasi manajemen aset berbasis *website* dengan memanfaatkan *low-code* platform untuk mengatasi masalah aksesibilitas dan integrasi. Dengan aplikasi berbasis *website*, pengelolaan aset dapat dilakukan dengan lebih luas dan fleksibel, tidak terbatas pada area rumah sakit saja. Serta fungsionalitas *website* yang diuji menggunakan *User Acceptance Test* untuk memastikan bahwa setiap fitur sudah memenuhi kebutuhan aktor yang terlibat. Hal ini diharapkan dapat meningkatkan efisiensi pengelolaan aset dan memberikan kemudahan akses informasi bagi pengelola di berbagai lokasi.

1.2 Perumusan Masalah

Berdasarkan uraian dari latar belakang tersebut, maka dapat dirumuskan beberapa permasalahan pada penelitian ini :

1. Bagaimana sistem informasi manajemen aset dapat membantu menyelesaikan masalah aksesibilitas dan integrasi pengelolaan aset kesehatan pada Rumah Sakit Permata Dalima?
2. Bagaimana hasil implementasi manajemen aset kesehatan menggunakan dukungan *low-code development*?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah untuk mendapatkan jawaban serta mendapat gambaran secara jelas mengenai masalah yang dikaji oleh peneliti. Tujuan penelitian antara lain:

1. Mengembangkan arsitektur sistem informasi untuk membantu mengelola aset Kesehatan.
2. Mengembangkan sistem manajemen pengelolaan distribusi aset kesehatan menggunakan dukungan *low-code development*.

1.4 Batasan Penelitian

Batasan penelitian fokus pada proses perancangan, pengembangan, dan pengujian *website* meliputi:

1. Perancangan dan pembuatan *Website* menggunakan Outsystems.
2. Pengujian *Website* menggunakan *User Acceptance Test* (UAT).
3. Aset yang dikelola pada *website* hanya aset habis pakai.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat bagi kajian ilmu Sistem Informasi baik secara umum maupun secara khusus. Peneliti juga berharap penelitian ini dapat berguna dalam kehidupan sehari-hari. Adapun manfaat penelitian ini, yaitu:

1. Bagi organisasi, pengembangan dengan metode ini dapat memberikan dukungan untuk meningkatkan pengelolaan distribusi aset kesehatan.
2. Dari sudut pandang akademis, dapat mengusulkan suatu metode atau teknik baru dalam mengelola distribusi aset kesehatan dengan memanfaatkan *low-code development*.
3. Bagi penulis, dapat memperkenalkan cara untuk mengembangkan aplikasi dengan lebih efektif dan efisien kepada pembaca.

BAB II

TINJAUAN PUSTAKA

2.1 *Low-Code Programming*

Low-code adalah pendekatan pengembangan perangkat lunak secara visual yang menyederhanakan pembuatan aplikasi dengan meminimalkan kebutuhan untuk penulisan kode manual yang rumit. Pendekatan ini memperkenalkan lingkungan pengembangan yang ramah pengguna, dengan fitur *drag-and-drop*, sehingga pengembangan aplikasi menjadi lebih mudah diakses baik bagi pemula maupun pengembang berpengalaman. Dengan pendekatan ini, pembuatan dan inovasi perangkat lunak bisnis dapat dilakukan lebih cepat, lebih inklusif, dan lebih efisien (Outsystems, 2024).

Penggunaan platform *low-code* dalam otomatisasi proses bisnis manufaktur telah menjadi semakin penting (Waszkowski, 2019). Pendekatan ini memungkinkan pengembangan aplikasi dengan usaha minimal dalam penulisan kode pemrograman. Kelebihan utama dari platform *low-code* adalah kemampuannya untuk mengotomatiskan berbagai proses bisnis dalam sektor manufaktur, yang pada gilirannya dapat meningkatkan efisiensi operasional terhadap perubahan kebutuhan pasar.

Platform *low-code* memungkinkan pengembangan aplikasi tanpa harus menulis kode secara manual, serta secara efisien meningkatkan kemampuan adaptasi terhadap perubahan atau evolusi (Bock & Frank, 2021). Selain itu, platform *low-code* dikenal dengan fitur desain Antarmuka Pengguna Grafis (GUI). Fitur ini memungkinkan pengguna untuk membuat aplikasi tanpa memerlukan latar belakang pendidikan di bidang teknologi informasi.

Selain itu, sebagian besar sistem pengembangan *low-code* mendukung berbagai target *environment*, termasuk desktop browser, tablet, dan *smartphone*, yang dapat diakses melalui satu platform.

2.1.1 Outsystems

Dalam penelitian yang dilakukan oleh (Martins, 2020), dijelaskan bahwa OutSystems adalah platform pengiriman cepat untuk aplikasi berbasis kode rendah yang mempercepat pengembangan aplikasi *mobile* dan web. Platform ini menyediakan tiga jenis entitas, yaitu entitas, entitas statis, dan entitas OutSystems. Entitas digunakan untuk menyimpan informasi dalam *database*, sedangkan entitas statis digunakan untuk mengelompokkan data yang jarang berubah. Sementara itu, entitas OutSystems secara otomatis dibuat oleh platform untuk memastikan fungsi aplikasi, seperti menyimpan menu aplikasi secara otomatis dalam entitas yang dihasilkan oleh platform tersebut. Selain itu, OutSystems menyediakan lingkungan pengembangan terintegrasi yang mencakup seluruh siklus pengembangan, mulai dari pengembangan, jaminan kualitas, implementasi, pemantauan, hingga manajemen.

OutSystems digunakan dalam pengembangan aplikasi web reaktif yang memudahkan pengelolaan tugas-tugas harian di kantor secara otomatis. Ini termasuk pengelolaan informasi dasar seperti keterampilan, manajemen dokumentasi internal, dan berbagai tugas lainnya. Platform ini memungkinkan pengguna untuk mengelola dokumentasi pribadi, mengajukan pengeluaran terkait proyek, dan mencatat jam kerja untuk setiap proyek. Selain itu, OutSystems memungkinkan pembuatan diagram hirarki institusi dan penugasan jabatan kepada karyawan, terutama yang berkaitan dengan *SuperUser* dan sumber daya manusia (Martins, 2020).

Arsitektur OutSystems terdiri dari Platform Server sebagai inti, yang merupakan komponen server yang melengkapi *stack* server aplikasi web standar dengan serangkaian layanan. Platform Server ini mengurus semua langkah yang diperlukan untuk menghasilkan, mengoptimalkan, mengompilasi, dan *deploy* aplikasi ke server aplikasi web standar atau dikemas ke dalam aplikasi *mobile native*. Selain itu, *OutSystems* dilengkapi dengan generator kode yang berperan sebagai kompilator untuk aplikasi. Hal ini memudahkan pengembang dalam memperbarui aplikasi dan melakukan perubahan pada *database* tanpa menghasilkan kesalahan, berkat arsitekturnya yang fleksibel (Martins, 2020).

OutSystems juga memiliki kurva pembelajaran yang cepat dan rendah karena menggunakan berbagai alat yang sudah dikenal seperti HTML, JavaScript, dan CSS. Hal ini memungkinkan pengembang untuk menghasilkan hasil dengan kualitas lebih baik dalam waktu yang lebih singkat. Platform ini dirancang untuk membantu pengembang dalam meningkatkan efisiensi dan efektivitas kerja mereka, sehingga aplikasi yang dihasilkan dapat memenuhi kebutuhan pengguna dengan lebih baik. Dengan demikian, OutSystems menawarkan solusi yang komprehensif dan efisien untuk pengembangan aplikasi berbasis kode rendah.

Menurut (Outsystems, 2017) dalam (Golovin, 2017) OutSystems adalah lingkungan pengembangan visual untuk membuat aplikasi *mobile* dan *web*. Platform ini dipromosikan sebagai "platform *low-code* nomor 1 untuk transformasi digital" dan memiliki banyak opsi bawaan yang dirancang untuk memudahkan pekerjaan seorang *programmer*. Menurut situs resminya, OutSystems digunakan oleh perusahaan besar seperti AXA, Bacardi Limited, Volkswagen, Honda Motor Co., Ltd., GDF Suez (sekarang dikenal sebagai Engie), Hewlett-Packard Company, ING Groep N.V., Intel Corporation, Liberty Seguros, Mercedes-Benz, Randstad Holding NV, Siemens AG, Charles River Laboratories, Inc., Vodafone Group, dan banyak perusahaan lainnya di 22 industri. Di sektor pendidikan, Bob Jones University, Georgia Institute of Technology, Kent State University, dan University of Georgia sudah menggunakan perangkat lunak ini. OutSystems adalah produk Platform as a Service (PaaS) yang berjalan di *cloud*, sehingga persyaratan sistem untuk perangkat keras komputer para pengembang rendah, namun diperlukan koneksi internet dengan kecepatan tinggi dan latensi rendah yang konstan.

Platform as a Service (PaaS) sedang memasuki era persaingan ketat, dengan SAP Business ByDesign dan Salesforce bersaing untuk mendominasi segmen pasar tertentu. Masih belum jelas apakah pasar ini akan menjadi "pemenang mengambil semua" (seperti Microsoft di pasar sistem operasi desktop) atau "pemenang mengambil sebagian besar" (seperti Google di pasar mesin pencari), dan para pengembang mungkin siap untuk beralih dari satu platform ke platform lain dengan mudah. Faktor keberhasilan terbesar bagi perusahaan di pasar PaaS

bukanlah produk itu sendiri atau teknologi di baliknya, melainkan strategi dan ekosistem yang mendukungnya. Perusahaan yang paling sukses akan mengikat para pengembang ke platform mereka dan mencegah mereka beralih ke solusi kompetitif (Giessmann & Stanoevska-Slabeva, 2012) dalam (Golovin, 2017).

2.2 Aset dan Aset Kesehatan

Aset merujuk pada benda atau barang yang memiliki nilai ekonomi, nilai komersial, atau nilai tukar yang dimiliki oleh suatu badan usaha, instansi, atau individu (Siregar, 2004). Dalam konteks ini, (Siregar, 2004) juga menjelaskan bahwa aset dapat berupa barang bergerak dan tidak bergerak, yang melibatkan aktiva untuk barang bergerak dan tanah atau bangunan untuk barang tidak bergerak. Perspektif International Federation of Accountants (2020) menambahkan bahwa aset merupakan sumber daya yang kini dikendalikan oleh entitas sebagai hasil dari peristiwa masa lalu. Sifat suatu sumber daya dapat dianggap sebagai aset ketika nilai manfaat di masa depan dapat diukur dengan pasti.

Aset kesehatan dapat dijelaskan sebagai segala jenis sumber daya fisik atau informasi yang digunakan dalam konteks layanan kesehatan, perawatan medis, dan manajemen kesehatan. Rentang aset ini mencakup, tetapi tidak terbatas pada, obat-obatan, peralatan medis, bahan medis, alat bantu medis, data medis elektronik, dan sumber daya manusia yang berperan dalam memberikan pelayanan kesehatan. Pentingnya aset kesehatan sangat signifikan dalam memastikan penyediaan layanan medis yang efektif, perawatan yang berkualitas, dan peningkatan kesejahteraan pasien.

2.3 Manajemen Aset

Manajemen aset merupakan salah satu aspek penting dalam mengoptimalkan penggunaan sumber daya dan investasi. Menurut dalam bukunya "Asset Management: A Systematic Approach to Factor Investing," manajemen aset didefinisikan sebagai proses sistematis dalam mengelola berbagai jenis aset untuk mencapai tujuan investasi tertentu. Manajemen ini tidak hanya berkaitan dengan pemilihan kelas aset seperti saham, obligasi, atau properti, tetapi

juga mencakup pengelolaan risiko faktor yang dapat memengaruhi kinerja aset tersebut. Risiko faktor ini meliputi berbagai kondisi ekonomi, seperti perlambatan pertumbuhan ekonomi, inflasi tinggi, dan volatilitas pasar. Ang menekankan pentingnya diversifikasi portofolio untuk meminimalkan dampak dari periode sulit, yang disebut sebagai *bad times*. Pada masa ini, investor yang mampu bertahan cenderung memperoleh *risk premiums*, yaitu imbal hasil yang lebih tinggi sebagai kompensasi atas risiko yang diambil. Selain itu, pengalokasian aset yang dinamis juga dibutuhkan untuk menyesuaikan portofolio berdasarkan perubahan kondisi pasar. Dalam konteks aset yang kurang likuid (*illiquid assets*), seperti properti atau aset jangka panjang lainnya, Ang menekankan bahwa meskipun aset-aset ini menawarkan potensi imbal hasil yang lebih tinggi, mereka juga membawa risiko likuiditas yang harus dikelola dengan hati-hati. Secara keseluruhan, pendekatan menyeluruh ini menjadikan manajemen aset sebagai strategi yang mempertimbangkan risiko, ketahanan jangka panjang, dan optimalisasi hasil investasi.

Menurut (Suakanto dkk., 2023), manajemen aset dapat membantu dalam mengurangi biaya siklus hidup, meningkatkan keandalan peralatan, dan memperbaiki komunikasi antara pemangku kepentingan internal dan eksternal. Di sektor kesehatan, penerapan manajemen aset yang efektif menjadi semakin penting karena dapat memaksimalkan penggunaan peralatan medis, mengurangi waktu yang diperlukan untuk pemeliharaan, menurunkan keluhan terkait ketersediaan peralatan, dan meningkatkan produktivitas rumah sakit. Selain itu, manajemen aset yang baik juga mampu mencegah kesalahan dalam penggunaan atau pelacakan aset, yang dapat berdampak langsung pada kualitas pelayanan dan keselamatan pasien. Dengan adanya sistem manajemen aset yang baik, risiko kehilangan atau kerusakan aset dapat diminimalkan, sehingga operasional rumah sakit dapat berjalan lebih efisien dan efektif.

Manajemen Aset terdiri dari dua kata, yaitu "Manajemen" dan "Aset". Menurut (Terry, 2012), Manajemen adalah suatu proses yang unik yang meliputi perencanaan, pengorganisasian, pelaksanaan, dan pengendalian. Proses ini

memanfaatkan berbagai pengetahuan dan keterampilan serta seni manajemen, yang dijalankan untuk mencapai tujuan yang telah ditetapkan.

Menurut (Sugiama, 2013) dalam (Wahyuni, 2023), manajemen aset adalah keahlian yang berfokus pada pengelolaan aset, yang mencakup berbagai kegiatan seperti identifikasi kebutuhan aset, perolehan aset, inventarisasi, audit hukum, penilaian aset, pengoperasian, dan pengalihan aset secara efektif dan efisien.

Manajemen aset dapat juga didefinisikan sebagai upaya untuk mengelola aset dengan tujuan memberikan manfaat optimal dengan biaya minimal dan menjaga agar aset tersebut tidak menjadi usang, kecuali jika memang diperlukan untuk dimusnahkan atau dihapuskan (Asih, 2018).

2.4 *Website*

Dikutip dari Kadir dalam (Syndi, 2018) , World Wide Web (WWW) atau lebih dikenal sebagai *website* adalah sebuah sistem yang memungkinkan akses informasi di internet melalui halaman-halaman yang terhubung menggunakan teknologi jaringan. Sebagaimana disampaikan oleh Suharto dalam (Syndi M, 2018), perlu dicatat bahwa jaringan dan Internet merupakan entitas yang berbeda. Internet merujuk pada infrastruktur fisik, sementara jaringan lebih mengacu pada aspek perangkat lunak. Lebih lanjut, Internet dan Web menggunakan protokol yang berbeda, di mana Internet menggunakan TCP/IP sebagai protokolnya, sedangkan Web menggunakan HTTP (*Hypertext Transfer Protocol*).

Menurut (Sari, 2019), dalam bukunya diuraikan bahwa *website* terbagi menjadi tiga kategori, yaitu:

- a. Web Statis
Jenis *website* yang memiliki halaman-halaman yang tetap tidak berubah. Untuk mengubah konten suatu halaman, perlu dilakukan secara manual dengan mengedit kode yang merupakan struktur dari *website* tersebut.
- b. Web Dinamis

Ini adalah jenis *website* yang didesain secara terstruktur untuk diperbarui secara reguler. Biasanya, *website* ini dilengkapi dengan halaman *backend* yang memungkinkan pembaruan konten secara teratur. Contoh-contoh *website* ini termasuk portal web, situs berita, dan sebagainya.

c. Web Interaktif

Ini adalah jenis *website* yang memfasilitasi interaksi antara pengguna. Umumnya, *website* ini berupa forum diskusi atau blog, di mana moderator bertugas mengatur jalannya diskusi.

2.5 Unit Testing

Dikutip dari (Tillmann & Schulte, 2005), *unit testing* adalah teknik pengujian perangkat lunak yang dilakukan untuk menguji bagian terkecil atau unit dari program, seperti fungsi atau metode, secara terpisah dari keseluruhan sistem. Tujuan utama dari *unit testing* adalah untuk memverifikasi bahwa setiap unit kode berfungsi sesuai dengan yang diharapkan sebelum unit tersebut diintegrasikan ke dalam sistem yang lebih besar. *Unit testing* menggunakan *test methods* yang tidak memiliki parameter, yang dirancang untuk menguji berbagai aspek perilaku dari kelas yang diuji. Dalam proses ini, sebuah *test oracle* digunakan untuk membandingkan hasil aktual dari eksekusi metode dengan hasil yang diharapkan, dan pengujian akan dianggap gagal jika terjadi perbedaan antara keduanya, seperti dalam kasus kegagalan pernyataan atau pengecualian yang tidak terduga.

Pendekatan tradisional dalam *unit testing* ini sering kali menggunakan serangkaian kasus uji statis dengan *input* tetap untuk mengeksplorasi berbagai jalur kode. Setiap unit diuji untuk memastikan bahwa komponen terkecil dari perangkat lunak bekerja sebagaimana mestinya, dan bila terjadi kesalahan, *unit testing* membantu menemukan masalah tersebut pada tahap pengembangan yang lebih awal, sehingga biaya perbaikan menjadi lebih rendah dibandingkan jika ditemukan pada tahap pengujian sistem atau integrasi (Tillmann & Schulte, 2005).

2.6 *User Acceptance Testing*

Dikutip dari (Mohamad & Yassin, 2016), *User Acceptance Testing* (UAT) adalah tahap akhir dalam proses pengujian perangkat lunak yang bertujuan untuk memastikan bahwa aplikasi atau sistem yang dikembangkan dapat memenuhi kebutuhan dan ekspektasi pengguna dalam situasi dunia nyata. Pengujian ini dilakukan oleh pengguna akhir atau pihak yang memiliki pengalaman langsung dengan proses bisnis yang terkait, untuk memverifikasi bahwa solusi yang ditawarkan bekerja dengan baik dalam skenario operasional sebenarnya. UAT fokus pada validasi fungsionalitas perangkat lunak dari sudut pandang pengguna, memastikan bahwa aplikasi tidak hanya memenuhi spesifikasi teknis tetapi juga efektif, efisien, dan dapat diandalkan untuk digunakan sehari-hari.

Menurut (Ganesh, 2014) *User Acceptance Testing* (UAT) adalah tahap akhir dalam pengujian perangkat lunak, di mana pengguna atau klien memverifikasi bahwa solusi yang dikembangkan memenuhi kebutuhan bisnis dan spesifikasi yang telah ditentukan. UAT dilakukan oleh pengguna akhir setelah komponen perangkat lunak lulus uji integrasi oleh konsultan, dan bertujuan untuk memastikan aplikasi dapat berfungsi dengan baik dalam lingkungan operasional yang sebenarnya. Selama UAT, segala *bug* atau cacat yang ditemukan harus diperbaiki sebelum klien memberikan sertifikat penerimaan. Pengujian ini melibatkan persiapan rencana uji, penjadwalan tim penguji, dan konfigurasi lingkungan UAT, dengan fokus pada memastikan bahwa seluruh fungsi sistem telah diuji.

2.7 *Agile*

Menurut (Murugaiyan, 2012), *Agile* adalah metodologi pengembangan perangkat lunak yang didasarkan pada pendekatan iteratif dan inkremental. Dalam metodologi ini, kebutuhan dan solusi berkembang melalui kolaborasi antara tim lintas fungsional yang mengorganisir diri sendiri. *Agile* bertujuan untuk memberikan perangkat lunak yang berfungsi dengan cepat dan terus-menerus, sehingga pelanggan dapat merasakan manfaatnya dalam waktu singkat. Metodologi ini sangat menekankan fleksibilitas dan adaptabilitas terhadap perubahan, bahkan pada tahap akhir pengembangan. Karakteristik utama dari

Agile adalah kemampuannya untuk beradaptasi dengan perubahan yang sering terjadi selama proses pengembangan. *Agile* memungkinkan tim untuk bekerja secara lebih responsif terhadap perubahan kebutuhan yang diajukan oleh pelanggan. Salah satu aspek penting dari *Agile* adalah kolaborasi yang intensif antara pengembang dan pengguna akhir. Komunikasi tatap muka antara kedua pihak dianggap krusial untuk memastikan keselarasan ekspektasi dan percepatan pengiriman produk yang sesuai dengan kebutuhan.

Disebutkan juga dalam , bahwa *Agile* adalah metodologi pengembangan perangkat lunak yang mengandalkan pengembangan iteratif dan inkremental. Dalam *Agile*, fase pengembangan akan terus diulang dan disempurnakan berdasarkan umpan balik pelanggan. Pendekatan ini memungkinkan fleksibilitas tinggi karena *Agile* menghindari pengumpulan kebutuhan secara menyeluruh di awal proyek, yang sering kali sulit dilakukan. Pelanggan memiliki kesempatan untuk berinteraksi dengan produk di setiap iterasi, memberikan umpan balik yang bisa langsung diimplementasikan dalam siklus berikutnya.

Empat prinsip utama dari *Agile* adalah:

1. Keterlibatan pelanggan sejak awal.
2. Pengembangan secara iteratif.
3. Tim yang mampu mengatur dirinya sendiri.
4. Adaptasi terhadap perubahan(*agile-traditional*).

Agile sangat efektif dalam proyek-proyek yang kompleks, di mana kebutuhan tidak dapat sepenuhnya ditentukan di awal. Dengan iterasi yang cepat, tim pengembang dapat dengan cepat merespons perubahan kebutuhan atau keinginan pelanggan tanpa harus melakukan perubahan besar pada sistem yang sudah dibangun. Ini sangat kontras dengan metode SDLC tradisional seperti *Waterfall* atau *V-Model*, di mana perubahan setelah fase awal bisa menjadi sulit dan mahal.

2.8 Penelitian Terdahulu

No	Judul	Penulis	Isi Tulisan	Fokus pengetahuan yg diadopsi
1	An overview on how to develop a low-code application using OutSystems	Ricardo Martins, Filipe Caldeira, Filipe S'a, Maryam Abbasi, Pedro Martins	Jurnal ini membahas pengembangan platform layanan mandiri untuk karyawan menggunakan OutSystems, yang memungkinkan pengelolaan informasi pribadi, penjadwalan, dan dokumentasi. Fitur yang disediakan mencakup pengeditan profil, permintaan cuti, dan pengajuan pengeluaran, dengan rencana kerja masa depan untuk notifikasi dan integrasi kalender.	Fokus pengetahuan yang diadopsi pada jurnal ini adalah pengembangan platform layanan mandiri untuk karyawan menggunakan aplikasi low-code OutSystems, yang mencakup pengelolaan informasi pribadi, penjadwalan, dan dokumentasi, serta fitur-fitur seperti pengeditan profil, permintaan cuti, dan pengajuan pengeluaran.
2	OutSystems as a Rapid Application Development Platform for Mobile and Web Applications	Dmitry Golovin	Tulisan ini membahas studi tentang penggunaan platform pengembangan aplikasi low-code OutSystems di Lahti University of Applied Sciences untuk menciptakan aplikasi mobile	Fokus pengetahuan yang diadopsi pada jurnal ini adalah penggunaan platform pengembangan aplikasi low-code OutSystems untuk menciptakan aplikasi mobile dan web yang bertujuan meningkatkan layanan bagi mahasiswa di Lahti University of Applied Sciences, serta tantangan yang dihadapi mahasiswa dalam