

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai salah satu dasar dalam penyusunan penelitian ini untuk melihat penelitian yang telah dilakukan berkaitan dengan sistem manajemen proyek dan sistem manajemen *ticket*. Penelitian yang digunakan sebagai referensi dipilih berdasarkan kesesuaian topik, permasalahan, serta metode atau pendekatan yang digunakan dalam pengembangan sistem. Setiap penelitian ditinjau berdasarkan permasalahan yang dibahas, metode yang digunakan, hasil yang diperoleh, serta keterbatasan yang masih terdapat pada penelitian tersebut. Pembahasan tersebut digunakan untuk melihat bagian yang telah dibahas pada penelitian sebelumnya dan bagian yang belum dibahas secara khusus, terutama yang berkaitan dengan pengelolaan *ticket* pada proyek *IT*. Hasil kajian kemudian digunakan untuk menentukan posisi penelitian ini serta menjadi dasar dalam menjelaskan perbedaan penelitian yang dilakukan dengan penelitian terdahulu. Beberapa penelitian yang menjadi referensi dalam penelitian ini disajikan pada pembahasan berikut.

Penelitian Mawaddah & Devitra (2023), menggunakan metode *Unified Modelling Language* (UML) dalam perancangan sistem untuk pengelolaan proyek pada perusahaan konstruksi. Penelitian tersebut membahas pengelolaan data proyek yang sebelumnya dilakukan secara manual dan kemudian dituangkan ke dalam sistem. Namun, penelitian ini belum membahas pengelolaan *ticket* pada proyek *IT* yang mencakup proses pengajuan, persetujuan, penugasan, pembaruan status pekerjaan, dan pencatatan riwayat aktivitas dalam satu sistem. Hal tersebut menjadi salah satu perbedaan dengan penelitian yang dilakukan karena penelitian ini berfokus pada pengelolaan *ticket* yang digunakan dalam proses pengelolaan proyek pada tim *IT*.

Penelitian Nugraha Purnawan dkk. (2022), mengembangkan sistem manajemen proyek berbasis *web* menggunakan metode *Waterfall* dan metode *Weighted Product* untuk proses penilaian proyek. Sistem yang dikembangkan digunakan untuk proses *monitoring* dan evaluasi proyek berdasarkan kebutuhan

pengguna. Namun, penelitian tersebut masih membahas manajemen proyek secara umum dan belum menjadikan *ticket* sebagai bagian utama dalam pengelolaan proyek. Selain itu, metode *Waterfall* yang digunakan memiliki tahapan pengembangan yang berbeda dengan penelitian ini yang menggunakan *Prototyping* dan melibatkan pengguna dalam proses evaluasi rancangan.

Penelitian Suwandana & Utami (2022), membahas pengembangan sistem informasi berbasis web dengan pendekatan PMBOK untuk sistem manajemen proyek. Sistem tersebut digunakan dalam proses pengelolaan dan pemantauan proyek serta penyajian informasi mengenai perkembangan proyek. Namun, penelitian tersebut belum menjelaskan pengelolaan *ticket* yang mencakup pengajuan, persetujuan, penugasan, pembaruan status, dan pemberian umpan balik pada proyek *IT*. Oleh karena itu, penelitian ini memiliki fokus yang berbeda karena pengelolaan proyek dilakukan melalui *ticket* sebagai bentuk pengajuan pekerjaan dari pengguna kepada tim *IT*.

Penelitian Auliansyah dkk., (2022), mengembangkan sistem *monitoring* proyek dengan menggunakan metode Kurva-S untuk melihat hubungan antara rencana dan realisasi pelaksanaan proyek. Pembahasan dalam penelitian tersebut mencakup perkembangan proyek berdasarkan waktu, biaya, dan kinerja. Namun, fokus penelitian berada pada *monitoring* proyek konstruksi sehingga belum membahas proses pengelolaan pekerjaan dalam proyek pengembangan perangkat lunak, khususnya melalui mekanisme *ticket*. Penelitian ini menggunakan *ticket* sebagai bagian dari proses pengajuan dan pengelolaan pekerjaan yang dilakukan oleh tim *IT*.

Penelitian Noer Arifin & Latifah Dwi Mutiara Sari (2024), mengembangkan sistem manajemen proyek berbasis *web* menggunakan metode *Rapid Application Development* (RAD). Sistem tersebut digunakan untuk pengelolaan proyek dan proses kolaborasi antar pengguna. Namun, penelitian tersebut belum menguraikan pengelolaan *ticket* secara khusus dan belum menggunakan proses evaluasi *prototype* secara berulang bersama pengguna dalam pengembangan sistem. Penelitian ini menggunakan metode *Prototyping* dengan melibatkan pengguna pada tahap evaluasi rancangan sebelum sistem dikembangkan lebih lanjut.

Tabel 2.1 Perbandingan Penelitian Terdahulu

Peneliti	(Mawaddah & Devitra, 2023)	(Nugraha Purnawan et al., 2022)	(Suwandana & Utami, 2022)	(Auliansyah et al., 2022)	(Noer Arifin & Latifah Dwi Mutiara Sari, 2024)
Masalah	Pengelolaan data proyek masih dilakukan secara manual dan belum ada digitalisasi, sehingga tidak efektif.	Tidak adanya sistem penilaian dan <i>monitoring</i> proyek yang efektif, serta masih menggunakan pencatatan manual.	Kurangnya sistem yang membantu <i>project manager</i> dan pimpinan dalam mengelola dan memantau proyek dari awal hingga selesai, serta pendokumentasian yang masih berantakan.	Ketidaksesuaian efisiensi dalam pengelolaan proyek dan kebutuhan akan sistem informasi manajemen proyek yang lebih efektif dan kolaboratif.	Kurangnya efisiensi dalam pengelolaan proyek dan kebutuhan akan sistem informasi manajemen proyek yang lebih efektif dan kolaboratif.
Metode	<i>Unified Modelling Language</i> (UML)	<i>Weighted Product Method</i>	<i>Project Management Body of Knowledge</i> (PMBOK)	Kurva-S	-
Pengembangan Sistem	<i>Prototype</i>	<i>Waterfall</i>	<i>Agile</i>	-	<i>Rapid Application Development</i> (RAD)
Pengujian	Metode pengujian yang digunakan tidak disebutkan	Pengujian fungsi program setelah <i>coding</i> (uji <i>logic & fungsional</i>).	Uji coba dengan data proyek nyata (peningkatan jalan Wadungasri – Tambaksu mur).	Pengujian prototipe berbasis <i>feedback</i> pengguna.	Tahap Testing setelah <i>coding</i> (uji keseluruhan sistem)
Hasil	Sistem informasi berbasis web yang dibuat dapat menyimpan data	Sistem yang dibuat dapat berjalan sesuai fungsi dan desain,	Sistem yang dibuat berhasil diterapkan dan dapat memfasilitasi komunikasi serta <i>monitoring</i>	Sistem yang dibuat berhasil menghasilkan <i>Gantt Chart</i> dan Kurva-S untuk	Sistem yang dibuat berhasil meningkatkan efisiensi pengelolaan proyek, mempermu

	proyek secara terorganisir, sehingga meningkatkan efisiensi pengelolaan proyek konstruksi.	sehingga kebutuhan pengguna dapat dipenuhi.	proyek. Hasil pengujian menunjukkan bahwa sistem dapat digunakan dengan baik tanpa ada error.	<i>monitoring</i> proyek. Sehingga deviasi proyek dapat terdeteksi dengan nilai -0.17% hingga -1.19%. Sedangkan rata-rata error dalam pengujian BRK adalah 51.07% dan BAK adalah 53.65%	dah dalam pemantauan kemajuan tugas, dan memberikan wawasan lebih baik terhadap produktivitas karyawan.
--	--	---	---	---	---

Berdasarkan penelitian terdahulu, terdapat berbagai penelitian yang membahas sistem manajemen proyek dengan metode dan pendekatan yang disesuaikan dengan kebutuhan masing-masing organisasi. Beberapa penelitian membahas pengelolaan data proyek, pemantauan perkembangan proyek, serta proses pengelolaan proyek dalam lingkungan kerja yang berbeda. Namun, penelitian yang dibahas belum secara khusus mengangkat pengelolaan *ticket* pada proyek *IT* yang mencakup proses pengajuan, persetujuan, penugasan anggota tim, pembaruan status pekerjaan, pemberian umpan balik, dan pencatatan riwayat aktivitas dalam satu sistem. Selain itu, penggunaan metode *Prototyping* dalam penelitian ini dilakukan dengan melibatkan pengguna pada proses evaluasi rancangan melalui beberapa iterasi. Berdasarkan hal tersebut, penelitian ini membahas perancangan dan pembangunan Sistem Manajemen *Ticket* berbasis *web* menggunakan metode *Prototyping* pada PT. Primanusa Globalindo dengan fokus pada proses pengelolaan dan pemantauan *ticket* dalam proyek *IT*.

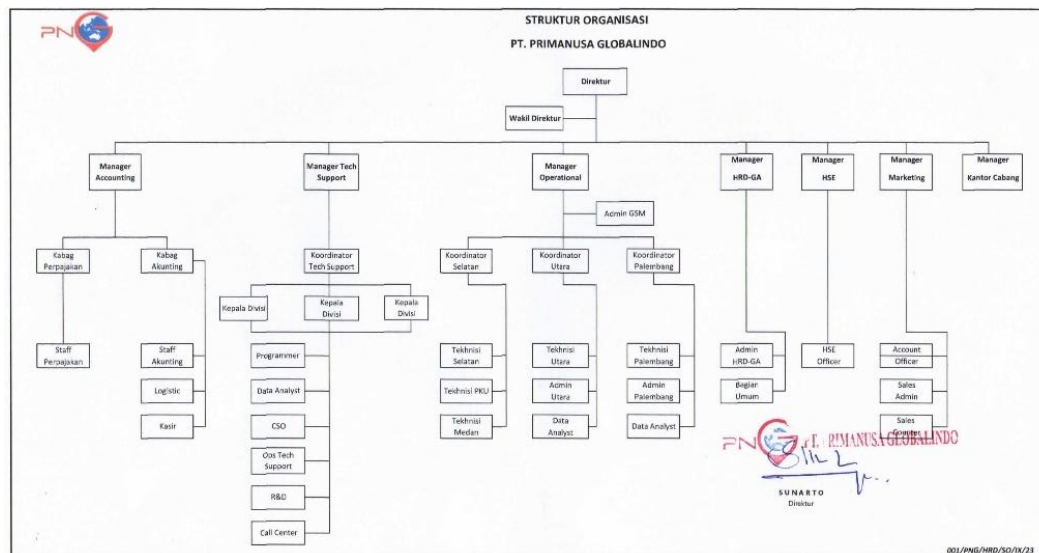
2.2 Landasan Teori

2.2.1 PT. Primanusa Globalindo

PT. Primanusa Globalindo merupakan perusahaan nasional yang menyediakan perangkat GPS *Tracker* dan CCTV dengan merek Gtrack. Perusahaan ini didirikan pada 31 Mei 2002 dan bergerak dalam penyediaan solusi teknologi yang berkaitan dengan pelacakan kendaraan serta pengelolaan armada. Produk

yang disediakan digunakan untuk kebutuhan pemantauan kendaraan dan keamanan sesuai dengan kebutuhan pelanggan (PT. Primanusa Globalindo, 2024).

PT. Primanusa Globalindo Cabang Pekanbaru berlokasi di Jl. Riau Ujung No. 349, Pekanbaru 28292, Riau. Cabang ini merupakan salah satu bagian dari kegiatan operasional perusahaan yang menangani pemasangan, pemeliharaan, dan perbaikan perangkat GPS Tracker dan CCTV untuk wilayah Sumatera, khususnya Provinsi Riau dan sekitarnya. PT. Primanusa Globalindo Cabang Pekanbaru memiliki 97 orang karyawan, dengan 7 orang di antaranya berada pada Divisi IT. Struktur organisasi PT. Primanusa Globalindo dapat dilihat pada gambar 2.1 (PT. Primanusa Globalindo, 2024).



Gambar 2.1 Struktur Organisasi PT. Primanusa Globalindo

Dalam menjalankan kegiatan operasionalnya, Divisi *IT* PT. Primanusa Globalindo menangani berbagai permintaan pekerjaan yang berkaitan dengan kebutuhan teknologi informasi perusahaan. Sebelum adanya Sistem Manajemen *Ticket*, proses pengajuan dan pengelolaan pekerjaan masih dilakukan melalui media komunikasi dan pencatatan yang digunakan oleh tim dalam kegiatan sehari-hari. Informasi mengenai permintaan pekerjaan, penugasan, perkembangan pekerjaan, dan penyelesaian pekerjaan disampaikan oleh pihak yang terlibat sesuai dengan kebutuhan pekerjaan yang sedang berlangsung.

Pihak yang terlibat dalam proses pengelolaan pekerjaan tersebut terdiri dari karyawan perusahaan sebagai pihak yang mengajukan kebutuhan, Manajer Proyek

sebagai pihak yang melakukan peninjauan dan menentukan penugasan, serta tim IT sebagai pihak yang melaksanakan pekerjaan. Karyawan dari berbagai bagian di perusahaan dapat mengajukan kebutuhan pekerjaan sesuai dengan kebutuhan masing-masing, sehingga pihak yang mengajukan permintaan tidak terbatas pada anggota Divisi IT. Manajer Proyek memiliki peran dalam meninjau permintaan yang masuk, menentukan pekerjaan yang perlu ditindaklanjuti, serta menentukan anggota tim IT yang akan mengerjakan pekerjaan tersebut.

Tim IT kemudian melaksanakan pekerjaan sesuai dengan penugasan yang diberikan dan menyampaikan perkembangan pekerjaan kepada pihak yang berkepentingan. Selain itu, terdapat anggota Divisi IT yang ditunjuk untuk menjalankan tugas administrasi, seperti mengelola data dan kebutuhan administrasi yang berkaitan dengan kegiatan pengelolaan pekerjaan. Peran tersebut merupakan pembagian tugas dalam kegiatan operasional sebelum adanya sistem dan tidak menunjukkan hak akses pada sistem yang dikembangkan.

Berdasarkan kondisi tersebut, proses pengelolaan pekerjaan pada Divisi IT melibatkan pihak pengaju dari berbagai bagian perusahaan, Manajer Proyek, tim IT, serta anggota tim IT yang menjalankan tugas administrasi. Kondisi dan pembagian peran tersebut menjadi dasar dalam mengidentifikasi kebutuhan pengguna dan menentukan aktor yang akan terlibat dalam perancangan Sistem Manajemen *Ticket* pada penelitian ini.

2.2.2 Manajemen Proyek

Berdasarkan PMBOK® *Guide* edisi ke-7 (Institute, 2021), manajemen proyek merupakan penerapan pengetahuan, keterampilan, alat, dan teknik pada aktivitas proyek untuk memenuhi kebutuhan yang telah ditetapkan. Panduan ini menekankan pentingnya pencapaian nilai bagi pemangku kepentingan, seperti efisiensi operasional, profitabilitas, peningkatan citra, atau inovasi, serta memastikan hasil akhir sesuai tujuan strategis organisasi. Dengan pendekatan adaptif, PMBOK memberikan fleksibilitas dalam mengelola proyek di tengah perubahan dan ketidakpastian, sambil memperhatikan risiko, peluang, dan kolaborasi lintas fungsi demi keberhasilan jangka panjang.

Menurut Arifin (2021), proses manajemen proyek terdiri dari 5 proses yang saling berkaitan untuk memastikan proyek dapat berjalan lebih terstruktur, tepat

waktu, dan sesuai dengan tujuan yang ingin dicapai. Berikut adalah 5 proses manajemen proyek:

- 1) Proses inisiasi (*Initiating Process*): Merupakan tahapan awal dalam manajemen proyek yang memiliki tujuan untuk menentukan kebutuhan dan tujuan proyek. Di tahap ini akan dilakukan identifikasi ruang lingkup proyek, *stakeholder* yang akan terlibat, serta permasalahan yang ingin diselesaikan. Tahapan ini menjadi penting karena tahapan ini akan menjadi dasar dari pengembangan proyek agar tepat sasaran.
- 2) Proses perencanaan (*Planning Process*): merupakan tahap penyusunan rencana kerja proyek yang lebih rinci. Pada tahap ini akan dilakukan penyusunan jadwal pekerjaan, estimasi biaya, pembagian tugas, penentuan sumber daya, serta mengidentifikasi risiko yang mungkin dapat terjadi pada proses pengembangan sistem. Perencanaan yang baik akan membantu proses selanjutnya lebih terarah dan dapat meminimalisir kesalahan pada pengerjaan proyek.
- 3) Pelaksanaan (*Executing Process*): Merupakan proses pelaksanaan atas seluruh rencana yang telah disusun pada tahap sebelumnya. Pada tahap ini tim pengembang akan mulai melakukan perancangan *UI* (antarmuka), lalu pembuatan sistem, pengembangan fitur-fitur, dan terakhir implementasi *database*. Selain itu, pada tahapan ini koordinasi antar anggota tim menjadi peran penting dalam proses pengembangan sistem agar berjalan sesuai dengan kebutuhan pengguna.
- 4) Proses pengawasan (*Controlling Process*): Merupakan tahapan pemantauan jalannya proyek, proses ini memastikan seluruh pengerjaan berjalan sesuai rencana. Selain itu, pada tahapan ini akan dilakukan evaluasi terhadap progres pekerjaan, pengujian sistem, serta perbaikan apabila terdapat kesalahan atau ketidaksesuaian pada sistem yang telah dibangun. Tahap controlling ini menjadi peran yang sangat penting untuk menjaga kualitas sistem yang dikembangkan agar sesuai dengan kebutuhan pengguna yang telah ditentukan sebelumnya.
- 5) Proses penutupan (*Closing Process*): Merupakan tahapan akhir dalam manajemen proyek. Pada tahap ini akan dilakukan penyelesaian seluruh

aktivitas proyek, dokumentasi hasil pengembangan, evaluasi sistem, serta penyerahan sistem kepada pengguna. Dalam penelitian ini, tahapan closing dilakukan setelah sistem selesai dilakukan pengujian dan dinyatakan dapat digunakan oleh pengguna.

2.2.3 Manajemen Ticket

Manajemen *ticket* merupakan bagian dari pengelolaan layanan *IT* yang digunakan untuk mencatat dan menangani permintaan atau permasalahan yang disampaikan oleh pengguna kepada tim *IT*. (Al-Hawari & Barham, 2021) menjelaskan bahwa sistem *help desk* berperan sebagai titik penghubung antara pengguna dan staf *IT* dalam penyampaian permintaan layanan maupun pelaporan permasalahan. Melalui *ticket*, pengguna dapat menyampaikan kebutuhan kepada tim *IT*, sedangkan pihak *IT* dapat mengelola permintaan tersebut dan melakukan tindak lanjut berdasarkan informasi yang terdapat pada *ticket*.

Dalam pengelolaan *ticket*, terdapat beberapa aktivitas yang dilakukan sejak *ticket* dibuat sampai pekerjaan diselesaikan. (Al-Hawari & Barham, 2021) menjelaskan aktivitas dalam proses penanganan *ticket* yang meliputi *opening*, *classifying*, *prioritizing*, *assigning*, *resolving*, *documenting*, *archiving*, dan *closing*. Proses tersebut menunjukkan bahwa *ticket* tidak hanya digunakan sebagai tempat pencatatan permintaan, tetapi juga menjadi bagian dari alur pekerjaan yang mencatat pihak yang menangani, perkembangan pekerjaan, serta hasil dari penanganan *ticket*.

Berdasarkan aktivitas tersebut, pengelolaan *ticket* dapat dijelaskan melalui beberapa tahapan. Tahapan pada penelitian ini tidak diterapkan secara keseluruhan sama seperti sistem *help desk* pada penelitian Al-Hawari dan Barham, tetapi disesuaikan dengan proses pengelolaan pekerjaan pada Divisi *IT* PT. Primanusa Globalindo.

1) Pengajuan Ticket

Tahap pertama dimulai ketika pengguna mengajukan kebutuhan pekerjaan melalui sistem. Pada tahap ini, pengguna mengisi informasi yang diperlukan untuk menjelaskan pekerjaan yang ingin diajukan. Proses tersebut berkaitan dengan aktivitas *opening* pada pengelolaan *ticket*, yaitu ketika permintaan atau permasalahan mulai dicatat ke dalam sistem (Al-Hawari & Barham, 2021).

Pada penelitian ini, pengguna yang mengajukan *ticket* disebut sebagai *requestor*. *Requestor* dapat berasal dari berbagai bagian di PT. Primanusa Globalindo yang memiliki kebutuhan pekerjaan kepada tim IT. Data yang telah dimasukkan kemudian menjadi dasar bagi proses peninjauan dan penanganan *ticket* pada tahap berikutnya.

2) Peninjauan dan Penentuan Prioritas

Setelah *ticket* dibuat, informasi yang terdapat di dalamnya perlu diperiksa sebelum pekerjaan dilanjutkan. Dalam proses *help desk*, *classifying* dan *prioritizing* merupakan bagian dari aktivitas penanganan *ticket* untuk menentukan karakteristik permintaan serta tingkat prioritas pekerjaan (Al-Hawari & Barham, 2021).

Pada penelitian ini, proses peninjauan dilakukan oleh Manajer Proyek. Manajer Proyek memeriksa permintaan yang diajukan oleh *requestor* dan menentukan tindakan yang perlu dilakukan terhadap *ticket*. Penyesuaian ini dilakukan berdasarkan proses yang diterapkan pada PT. Primanusa Globalindo, sehingga fungsi peninjauan dan penentuan prioritas berada pada pihak yang memiliki kewenangan dalam pengelolaan pekerjaan IT.

3) Penugasan

Setelah *ticket* ditinjau dan dapat dilanjutkan ke tahap pengerjaan, dilakukan penugasan kepada anggota tim IT yang bertanggung jawab terhadap pekerjaan tersebut. Aktivitas *assigning* merupakan salah satu bagian dari proses penanganan *ticket* yang dijelaskan oleh (Al-Hawari & Barham, 2021), yaitu menentukan pihak yang akan menangani *ticket* sesuai dengan kebutuhan layanan yang diberikan.

Pada sistem yang dikembangkan, Manager menentukan *Person in Charge* (PIC) yang akan mengerjakan *ticket*. Selain PIC, sistem juga dapat mencatat anggota tim lain yang terlibat sebagai *assist*. Informasi tersebut digunakan untuk menunjukkan pihak yang bertanggung jawab dan terlibat dalam pengerjaan suatu *ticket*.

4) Pengerjaan dan Pembaruan *Ticket*

Setelah PIC ditentukan, anggota tim IT melakukan pekerjaan berdasarkan informasi dan kebutuhan yang tercantum pada *ticket*. Selama proses pengerjaan

berlangsung, informasi mengenai kondisi pekerjaan dapat diperbarui sesuai dengan perkembangan yang terjadi. Dalam sistem *help desk*, proses penanganan *ticket* mencakup aktivitas *resolving* yang berkaitan dengan penyelesaian permintaan atau permasalahan yang dilaporkan oleh pengguna (Al-Hawari & Barham, 2021).

Pada penelitian ini, anggota tim *IT* dapat memperbarui status *ticket* dan mencatat perkembangan pekerjaan selama proses pengerjaan. Pembaruan tersebut menjadi bagian dari informasi *ticket* sehingga kondisi pekerjaan dapat diketahui oleh pihak yang memiliki hak akses terhadap *ticket*.

5) Penyelesaian dan Dokumentasi *Ticket*

Setelah pekerjaan selesai dilakukan, hasil pekerjaan dicatat pada sistem sebagai bagian dari penyelesaian *ticket*. Dokumentasi merupakan salah satu aktivitas yang terdapat dalam proses pengelolaan *ticket*, karena informasi mengenai pekerjaan yang telah dilakukan perlu disimpan sebagai bagian dari data penanganan *ticket* (Al-Hawari & Barham, 2021).

Pada sistem yang dikembangkan, hasil pekerjaan dapat diberikan kepada *requestor* untuk ditinjau. *Requestor* kemudian dapat memberikan persetujuan atau *feedback* terhadap hasil pekerjaan yang diberikan. Proses tersebut disesuaikan dengan kebutuhan PT. Primanusa Globalindo dalam pengelolaan pekerjaan *IT* yang dikembangkan pada penelitian ini.

6) Penutupan dan Riwayat Aktivitas

Tahap akhir dilakukan setelah pekerjaan dan proses pemeriksaan hasil telah selesai. *Ticket* kemudian dapat ditutup sesuai dengan kondisi penyelesaian pekerjaan. (Al-Hawari & Barham, 2021) memasukkan aktivitas *archiving* dan *closing* sebagai bagian dari proses penanganan *ticket*, yang menunjukkan bahwa informasi mengenai *ticket* tetap menjadi bagian dari catatan sistem setelah proses penanganannya selesai.

Selain status penyelesaian, sistem yang dikembangkan juga mencatat aktivitas yang terjadi selama proses pengelolaan *ticket*. Riwayat tersebut dapat mencakup perubahan status, penugasan, maupun aktivitas lain yang dilakukan oleh pengguna. Pencatatan ini digunakan untuk menyimpan informasi

mengenai proses yang telah dilakukan terhadap suatu *ticket* sejak dibuat sampai selesai.

Berdasarkan proses tersebut, pengelolaan *ticket* pada penelitian ini disesuaikan dengan alur pekerjaan pada Divisi IT PT. Primanusa Globalindo. Proses dimulai dari pengajuan *ticket* oleh *requestor*, kemudian dilakukan peninjauan dan penentuan prioritas oleh Manajer Proyek. Setelah *ticket* dapat ditindaklanjuti, Manajer Proyek menentukan *Person in Charge* (PIC) dan anggota *assist* apabila diperlukan. Selanjutnya, tim IT mengerjakan pekerjaan, memperbarui perkembangan *ticket*, dan mencatat hasil pekerjaan pada sistem hingga *ticket* selesai dan ditutup.

Dalam menentukan urutan penanganan pekerjaan, prioritas dapat digunakan sebagai dasar untuk menentukan pekerjaan yang perlu didahulukan. (Afrianto et al., 2024) membahas kombinasi *Priority Scheduling* dan *Earliest Due Date* (EDD), dengan *Priority Scheduling* digunakan terlebih dahulu dan EDD digunakan ketika terdapat beberapa pekerjaan dengan tingkat prioritas yang sama. EDD menentukan urutan berdasarkan batas waktu penyelesaian, sehingga pekerjaan dengan tanggal penyelesaian yang lebih dekat ditempatkan pada urutan yang lebih awal (Afrianto et al., 2024)

Konsep tersebut diterapkan pada penelitian ini dalam bentuk pengurutan *ticket* berdasarkan tingkat prioritas dan *estimated finish date*. *Ticket* dengan prioritas lebih tinggi ditempatkan lebih dahulu, sedangkan apabila terdapat *ticket* dengan tingkat prioritas yang sama, urutannya ditentukan berdasarkan *estimated finish date* yang paling dekat. Dengan demikian, konsep *Priority Scheduling* dan EDD digunakan sebagai dasar dalam menentukan urutan penanganan *ticket* pada sistem yang dikembangkan.

2.2.4 Proses Pengembangan Perangkat Lunak

Proses pengembangan perangkat lunak (*Software Development Process*) merupakan serangkaian tahapan yang dilakukan secara sistematis untuk menghasilkan perangkat lunak yang sesuai dengan kebutuhan pengguna. Proses ini mencakup kegiatan mulai dari identifikasi kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan perangkat lunak. Setiap tahapan

saling berkaitan sehingga hasil yang diperoleh pada satu tahap menjadi dasar pelaksanaan tahap berikutnya.

Terdapat berbagai model yang dapat digunakan dalam proses pengembangan perangkat lunak, di antaranya *Waterfall*, *Prototype*, *Spiral*, *Rapid Application Development* (RAD), *Agile*, dan *Scrum*. Masing-masing model memiliki karakteristik, tahapan, serta keunggulan yang berbeda sesuai dengan kebutuhan proyek dan tingkat perubahan kebutuhan pengguna. Oleh karena itu, pemilihan model pengembangan perlu disesuaikan dengan karakteristik sistem yang akan dibangun.

Pada penelitian ini, model pengembangan yang digunakan adalah *Prototyping* karena kebutuhan pengguna berkembang selama proses perancangan sistem. Melalui pendekatan ini, pengguna dapat memberikan masukan pada setiap iterasi sehingga rancangan sistem dapat disesuaikan dengan kebutuhan yang diharapkan sebelum diimplementasikan secara penuh.

2.2.5 Metode *Prototyping*

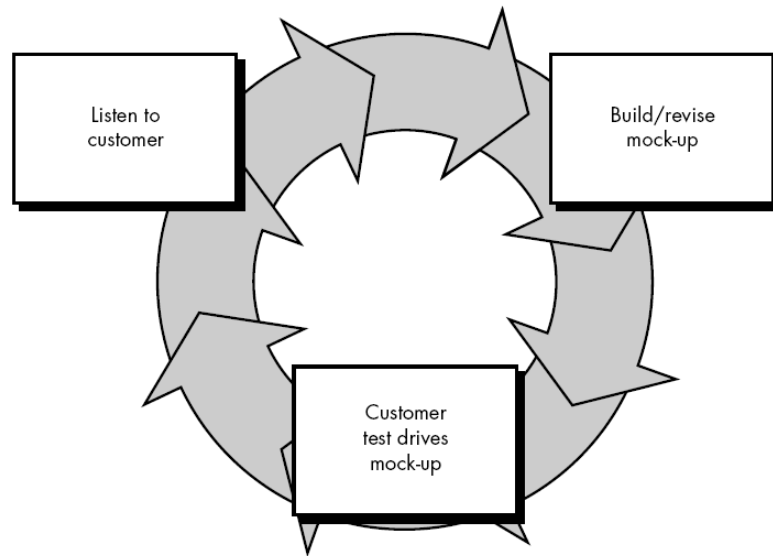
Metode *Prototyping* merupakan salah satu model dalam proses pengembangan perangkat lunak yang menekankan pembuatan rancangan awal (*prototype*) sebagai media komunikasi antara pengembang dan pengguna. Menurut Pressman (2010), *prototype* merupakan representasi awal dari sistem yang dikembangkan untuk membantu pengguna memberikan gambaran mengenai kebutuhan sistem sebelum implementasi dilakukan. Melalui pendekatan ini, pengguna dapat mengevaluasi rancangan yang telah dibuat dan memberikan masukan sehingga pengembang memperoleh pemahaman yang lebih baik terhadap kebutuhan sistem.

Berbeda dengan model pengembangan yang menetapkan kebutuhan sistem secara lengkap sejak awal, metode *Prototyping* memungkinkan perubahan kebutuhan dilakukan selama proses pengembangan berlangsung. Setiap masukan dari pengguna digunakan sebagai dasar untuk memperbaiki atau menyempurnakan *prototype* pada iterasi berikutnya hingga diperoleh rancangan yang sesuai dengan kebutuhan pengguna. Pendekatan ini menjadikan komunikasi antara pengguna dan pengembang berlangsung secara berkelanjutan selama proses pengembangan sistem.

Menurut Pressman (2010), metode *Prototyping* terdiri atas tiga tahapan utama, yaitu *Listen to Customer*, *Build/Revise Mock-up*, dan *Customer Test-Drives Mock-up*. Tahap *Listen to Customer* dilakukan untuk mengidentifikasi kebutuhan pengguna melalui kegiatan seperti wawancara dan observasi. Selanjutnya, pada tahap *Build/Revise Mock-up*, pengembang menyusun rancangan awal sistem berdasarkan kebutuhan yang telah diperoleh. Rancangan tersebut kemudian dievaluasi oleh pengguna pada tahap *Customer Test-Drives Mock-up*. Apabila masih terdapat kebutuhan yang belum sesuai, maka proses kembali ke tahap awal hingga *prototype* dinilai telah memenuhi kebutuhan pengguna.

Metode *Prototyping* memiliki beberapa kelebihan, di antaranya memberikan kesempatan kepada pengguna untuk berpartisipasi secara aktif selama proses pengembangan, memudahkan pengembang dalam memahami kebutuhan pengguna, serta memungkinkan perubahan kebutuhan dilakukan sebelum sistem diimplementasikan secara penuh. Namun demikian, metode ini juga memiliki keterbatasan, seperti kemungkinan terjadinya perubahan kebutuhan secara berulang yang dapat memengaruhi waktu pengembangan apabila ruang lingkup proyek tidak didefinisikan dengan baik.

Pada penelitian ini, metode *Prototyping* dipilih karena pengembangan Sistem Manajemen *Ticket* memerlukan keterlibatan pengguna dalam proses penyusunan kebutuhan dan evaluasi rancangan sistem. Melalui proses iterasi, pengguna dapat memberikan masukan terhadap *mock-up* yang dikembangkan sehingga rancangan sistem yang dihasilkan disesuaikan dengan kebutuhan operasional divisi *IT* PT. Pramanusa Globalindo.



Gambar 2.2 Metode Prototyping
(Sumber: ilmurplkitabersama.blogspot.com)

Metode ini dipilih karena mampu membantu proses pengembangan sistem secara iteratif. Berikut adalah penjelasan mengenai gambar 2.2:

1) *Listen to customer*

Pada tahapan ini akan dilakukan penggalan informasi dari *user* untuk mendapatkan informasi mengenai kebutuhan yang diperlukan pada sistem. Pengumpulan data dilakukan melalui tahapan wawancara dan observasi terkait proses bisnis yang berjalan saat ini. Hasil dari tahapan ini akan digunakan sebagai dasar dalam menentukan fitur dan kebutuhan sistem di proses selanjutnya.

2) *Build/revise mock-up*

Pada tahapan ini pembuatan rancangan awal sistem berupa *mock-up* atau prototipe berdasarkan kebutuhan pengguna yang telah didapatkan dari tahapan sebelumnya. *Mock-up* atau prototipe yang dibuat akan dapat direvisi kembali apabila terdapat masukan atau perubahan dari pengguna saat dilakukan pengujian.

3) *Customer test-drives mock-up*

Pada tahapan ini prototipe diserahkan, prototipe akan diserahkan kepada *user* untuk diuji dan dinilai. Jika *user* memberikan masukan atau revisi maka tahapan akan dimulai lagi dari *listen to customer*. Jika tidak ada perubahan maka akan dilanjutkan ke tahapan implementasi dan pengujian.

2.2.6 Functional Testing

Pengujian perangkat lunak (*software testing*) adalah tahap yang sangat penting dalam proses rekayasa perangkat lunak, karena menjadi langkah untuk memastikan bahwa sistem yang dibangun mampu memenuhi kebutuhan pengguna serta berjalan sesuai dengan spesifikasi yang telah ditetapkan. Salah satu pendekatan yang umum digunakan adalah *functional testing*. Menurut (Pressman & Maxim, 2020), *functional testing* adalah proses verifikasi fungsi perangkat lunak yang dilakukan dengan memeriksa perilaku eksternal sistem. Pengujian ini berfokus pada apa yang dilakukan sistem, bukan bagaimana sistem tersebut melakukannya. Dengan kata lain, *functional testing* menilai kesesuaian antara input yang diberikan dan output yang dihasilkan tanpa mempertimbangkan struktur internal atau logika program yang digunakan.

Salah satu metode yang termasuk dalam *functional testing* adalah *black-box testing*. Menurut (Wicaksono, 2021), *black-box testing* adalah teknik pengujian yang memeriksa fungsi sistem dari sisi eksternal tanpa memerlukan pengetahuan tentang struktur internal atau kode program. Dalam metode ini, penguji bertindak layaknya pengguna akhir (*end-user*) yang hanya berinteraksi dengan antarmuka sistem, memberikan input tertentu, dan kemudian memeriksa apakah output yang dihasilkan sudah sesuai dengan yang diharapkan. Keunggulan pendekatan ini adalah kemampuannya dalam mendeteksi kesalahan fungsi, permasalahan pada antarmuka, maupun kegagalan integrasi antar modul, tanpa terjebak pada kompleksitas teknis di dalam sistem.

2.2.7 Usability Testing

Usability Testing merupakan salah satu metode evaluasi yang digunakan untuk mengetahui sejauh mana suatu sistem dapat digunakan oleh pengguna dalam menyelesaikan tugas yang diberikan. Pengujian ini dilakukan dengan melibatkan pengguna secara langsung untuk mengamati interaksi pengguna dengan sistem serta mengidentifikasi kendala yang muncul selama proses penggunaan. Menurut (Hertzum, 2020), *Usability Testing* merupakan bagian dari pendekatan *user-centered design* yang bertujuan memperoleh umpan balik pengguna sebagai dasar dalam melakukan penyempurnaan terhadap antarmuka maupun fungsi sistem.

Menurut (Rahman Deluma et al., 2023), evaluasi *usability* mengacu pada standar **ISO 9241-11** yang terdiri atas tiga aspek utama, yaitu *effectiveness*, *efficiency*, dan *satisfaction*. *Effectiveness* menunjukkan kemampuan pengguna dalam menyelesaikan tugas sesuai tujuan yang telah ditetapkan. *Efficiency* menggambarkan sumber daya yang digunakan selama pengguna menyelesaikan tugas tersebut, sedangkan *satisfaction* menunjukkan tingkat kenyamanan dan penerimaan pengguna ketika menggunakan sistem. Ketiga aspek tersebut digunakan sebagai dasar dalam mengevaluasi kualitas penggunaan suatu sistem sehingga dapat diketahui apakah sistem telah sesuai dengan kebutuhan pengguna.

Pelaksanaan *Usability Testing* umumnya terdiri atas tiga tahapan, yaitu persiapan, pelaksanaan, dan analisis hasil. Tahap persiapan dilakukan dengan menentukan tujuan pengujian, menyusun skenario penggunaan, menentukan responden, serta menyiapkan instrumen pengujian. Selanjutnya, pada tahap pelaksanaan, pengguna diminta menjalankan tugas sesuai skenario yang telah ditentukan, sedangkan peneliti mengamati proses penggunaan sistem dan mencatat berbagai kendala maupun masukan yang diberikan oleh pengguna. Tahap terakhir adalah analisis hasil, yaitu mengidentifikasi temuan selama pengujian untuk mengetahui bagian sistem yang masih memerlukan perbaikan maupun bagian yang telah sesuai dengan kebutuhan pengguna (Hertzum, 2020).

Salah satu instrumen yang banyak digunakan dalam *Usability Testing* adalah *System Usability Scale* (SUS). Menurut (Kesuma, 2021), SUS merupakan instrumen evaluasi yang digunakan untuk mengukur persepsi pengguna terhadap kemudahan penggunaan suatu sistem melalui sepuluh pernyataan yang disusun secara bergantian antara pernyataan positif dan negatif. Pengisian kuesioner SUS menggunakan Skala *Likert* lima tingkat yang terdiri atas Sangat Tidak Setuju (STS), Tidak Setuju (TS), Netral (N), Setuju (S), dan Sangat Setuju (SS) dengan skor 1 sampai 5. Jawaban responden selanjutnya diolah menggunakan aturan perhitungan SUS sehingga menghasilkan skor pada rentang 0–100. Nilai tersebut kemudian diinterpretasikan untuk mengetahui tingkat *usability* sistem berdasarkan persepsi pengguna. Instrumen SUS banyak digunakan karena memiliki proses pengisian yang sederhana, mudah dipahami oleh responden, serta dapat diterapkan pada berbagai jenis perangkat lunak.

2.2.8 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) adalah tahap akhir dalam proses pengujian perangkat lunak, di mana sistem yang telah dikembangkan diuji langsung oleh pengguna akhir atau perwakilan bisnis. Menurut Hambling & Goethem (2013) UAT adalah “*formal testing with respect to user needs, requirements, and business processes, conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system*”. Dengan demikian, UAT dilakukan untuk memastikan bahwa sistem telah sesuai dengan kebutuhan pengguna, persyaratan yang telah ditetapkan, serta proses bisnis yang berlaku sehingga dapat diterima untuk digunakan.

Pelaksanaan UAT dilakukan berdasarkan skenario pengujian yang disusun sesuai dengan kebutuhan fungsional sistem. Pengguna menjalankan setiap fungsi sesuai dengan proses bisnis yang telah ditentukan, kemudian memberikan penilaian terhadap kesesuaian hasil yang diperoleh. Apabila fungsi sistem telah berjalan sesuai dengan kebutuhan pengguna dan memenuhi *acceptance criteria*, maka fungsi tersebut dinyatakan diterima (*accepted*). Sebaliknya, apabila masih ditemukan ketidaksesuaian terhadap kebutuhan pengguna, maka sistem memerlukan perbaikan sebelum digunakan pada lingkungan operasional.

Menurut (Hambling & Goethem, 2013), keberhasilan pelaksanaan UAT sangat bergantung pada keterlibatan pengguna selama proses pengujian. Partisipasi pengguna memungkinkan setiap fungsi sistem dievaluasi berdasarkan pengalaman penggunaan secara langsung sehingga ketidaksesuaian terhadap kebutuhan operasional dapat diidentifikasi sebelum sistem diimplementasikan. Selain itu, masukan yang diberikan pengguna dapat menjadi dasar dalam melakukan penyempurnaan terhadap fungsi sistem agar lebih sesuai dengan proses bisnis yang berlaku.

Pada penelitian ini, *User Acceptance Testing (UAT)* dilakukan dengan melibatkan empat jenis pengguna, yaitu Admin, Manajer Proyek, Anggota Tim IT, dan *Requestor*. Setiap pengguna melakukan pengujian berdasarkan skenario yang disusun sesuai dengan hak akses dan fungsi yang dimiliki. Hasil pengujian digunakan untuk mengetahui apakah setiap fungsi sistem diterima oleh pengguna

dan telah sesuai dengan kebutuhan operasional Divisi *IT* PT. Primanusa Globalindo.