

Penerapan Metode Extreme Programming dalam Pengembangan Sistem Inventaris Barang Habis Pakai Berbasis Web pada Instansi XYZ

1st Aisyah Nabila Zahra
 Sistem Informasi
 Telkom University Surabaya
 Surabaya, Indonesia
aisyahnabilaz@student.telkomuniversity.ac.id

2nd Adznil Rachmadhi Putra
 Sistem Informasi
 Telkom University Surabaya
 Surabaya, Indonesia
adzrachmadhip@telkomuniversity.ac.id

3rd Purnama Anaking
 Sistem Informasi
 Telkom University Surabaya
 Surabaya, Indonesia
purnamaanaking@telkomuniversity.ac.id

Abstrak — Instansi pemerintah tingkat provinsi yang menangani distribusi barang habis pakai pada berbagai unit kerja sering menghadapi kendala dalam proses pencatatan dan pengelolaan inventaris barang pakai habis yang belum terintegrasi. Prosedur permintaan barang yang masih dilakukan secara manual mengakibatkan keterlambatan distribusi, kesalahan pencatatan, dan minimnya transparansi data secara *real-time*. Penelitian ini bertujuan untuk mengembangkan sistem informasi pengelolaan inventaris berbasis web dengan menerapkan metode *Extreme Programming* (XP). Sistem ini dibangun melalui tahapan perumusan *user story*, *release planning*, perancangan sistem dengan iterative, pengujian fungsional dan terstruktur, serta *usability testing*. Teknologi yang digunakan dalam pengembangan meliputi Laravel sebagai kerangka kerja *backend*, MySQL untuk pengelolaan basis data, serta Tailwind CSS pada antarmuka pengguna. Evaluasi sistem dilakukan melalui pengujian *blackbox*, *whitebox*, serta pengujian kegunaan berdasarkan lima aspek dari Nielsen. Hasil menunjukkan bahwa sistem dapat mempercepat proses permintaan barang, meningkatkan akurasi pencatatan, serta memberikan transparansi terhadap data persediaan. Temuan ini menunjukkan bahwa metode XP efektif untuk diterapkan dalam pengembangan sistem informasi pada lingkungan instansi yang belum mengetahui kepastian kebutuhan sistemnya.

Kata kunci— *extreme programming*, *pencatatan inventaris*, *persediaan barang habis pakai*, *sistem informasi*, *usability testing*

I. PENDAHULUAN

Instansi pemerintah tingkat provinsi yang menangani urusan sosial memiliki tanggung jawab terhadap kelancaran operasional berbagai unit kerja dan Unit Pelaksanaan Teknis (UPT) yang tersebar di wilayah administratifnya. Salah satu aspek penting dalam menunjang layanan tersebut adalah pengelolaan inventaris barang habis pakai, seperti alat tulis kantor, bahan komputer, obat-obatan, hingga peralatan pendukung kegiatan. Barang-barang ini dikelola oleh unit administrasi internal dengan tanggung jawab terhadap dokumentasi dan distribusi, yang sekaligus memastikan kesesuaian dengan prinsip pengelolaan Barang Milik Negara (BMN) [1].

Saat ini, instansi terkait telah menggunakan sistem informasi berbasis web untuk mendukung pencatatan barang masuk dan keluar, pelaporan bulanan, serta perhitungan saldo persediaan. Namun, masih terdapat sejumlah keterbatasan yang menghambat pengguna sistem secara optimal. Salah satu masalah utama adalah proses pengajuan permintaan barang yang masih dilakukan secara manual. Pegawai dari unit kerja dan UPT tidak memiliki akses langsung ke informasi stok gudang, sehingga seluruh permintaan harus disampaikan melalui Pengelola Gudang. Ketergantungan terhadap satu peran menyebabkan pengajuan tidak dapat dilakukan jika pegawai tersebut tidak tersedia,, dan semua catatan permintaan harus diinput ulang secara manual ke dalam sistem. Proses ini beresiko menimbulkan kesalahan pencatatan dimasa mendatang.

Selain itu, pencatatan kartu barang masih dilakukan secara manual dan terpisah dari sistem utama. Hal ini menyebabkan ketidaksesuaian antara data sistem dan kondisi fisik barang saat dilakukan pemeriksaan stok (*stock opname*), serta menyulitkan pelacakan riwayat transaksi untuk setiap unit barang. Sistem juga belum memiliki alur digital untuk validasi dan persetujuan permintaan, sehingga proses verifikasi dilakukan secara informal dan tidak terdokumentasi secara sistematis.

Penelitian terdahulu menunjukkan bahwa metode pengembangan perangkat lunak berbasis *Agile*, seperti *extreme programming* (XP), cocok untuk diterapkan dalam konteks organisasi yang memiliki kebutuhan dinamis dan sumber daya terbatas [2], [3], [4]. Pendekatan XP telah berhasil digunakan dalam pengembangan sistem layanan public, dan terbukti memberikan kemudahan dalam menyesuaikan sistem dengan kebutuhan pengguna pada setiap iterasi [5]. Teknologi pendukung seperti QR Code juga telah dimanfaatkan dalam sistem informasi inventaris berbasis *mobile* untuk mempercepat proses pelacakan dan dokumentasi barang [6].

Berdasarkan indentifikasi permasalahan tersebut, penelitian ini bertujuan untuk mengembangkan sistem informasi pengelolaan inventaris barang habis pakai berbasis web dengan menambahkan tiga fitur utama. Pertama, fitur *self-request submission* yang memungkinkan pegawai dari berbagai unit kerja dan UPT untuk mengajukan permintaan

barang secara mandiri dan melihat status stok secara *real-time*. Kedua, penggantian kartu barang manual dengan kartu digital yang terintegrasi melalui QR *Code* untuk memudahkan pelacakan dan dokumentasi transaksi barang. Ketiga, digitalisasi alur validasi dan persetujuan permintaan barang melalui sistem, yang melibatkan tiga level peran administratif internal. Sistem dikembangkan dengan menggunakan *framework* Laravel, berbasis data MySQL, dan antarmuka pengguna berbasis Tailwind CSS. Pendekatan XP digunakan untuk membangun sistem secara iterative dengan keterlibatan pengguna langsung dalam setiap tahapan pengembangan. Evaluasi sistem melalui pengujian, *blackbox*, *whitebox*, serta *usability testing* berdasarkan lima aspek dari Nielsen. Sistem ini dirancang untuk menggantikan proses manual yang masih dominan, serta memperbaiki akses pengguna terhadap informasi stok, alur permintaan barang, dan dokumentasi kartu barang melalui sistem yang lebih terstruktur.

II. KAJIAN TEORI

A. Sistem informasi dan sistem inventaris

Sistem informasi merupakan integrasi antara teknologi informasi dan aktivitas manusia yang digunakan untuk mengumpulkan, memproses, menyimpan, dan menyebarkan data menjadi informasi yang berguna [7]. Dalam suatu organisasi sistem informasi berperan penting untuk mendukung proses operasional, membantu pengambilan keputusan, serta mengurangi beban kerja manual yang berisiko menimbulkan kesalahan. Komponen utama dalam sistem informasi meliputi data, perangkat keras, perangkat lunak, jaringan komunikasi, pengguna, dan prosedur [7]. Masing-masing komponen tersebut bekerja secara terintegrasi untuk memastikan bahwa informasi dapat dihasilkan dan digunakan sesuai kebutuhan.

Salah satu penerapan sistem informasi yang umum digunakan dalam organisasi adalah sistem informasi inventaris. Sistem ini dirancang untuk mencatat dan mengelola barang, baik habis pakai maupun tidak habis pakai, guna memastikan ketersediaan, akurasi data, dan kelancaran distribusi [8], [9], [10]. Sistem informasi inventaris menyediakan fungsi pencatatan mutasi barang, pemantauan stok secara *real-time*, serta pelaporan yang mendukung pelacakan barang dan pengambilan keputusan berbasis data. Tujuan utama dari sistem ini adalah mengurangi kesalahan pencatatan, mempercepat proses kerja yang sebelumnya dilakukan secara manual, serta meningkatkan transparansi dalam pengelolaan logistik internal organisasi [11].

B. Teknologi Pengembangan Web

Pengembangan sistem informasi berbasis web pada penelitian ini menggunakan Laravel, MySQL, dan Tailwind CSS sebagai fondasi teknologinya. Pemilihan teknologi dilakukan berdasarkan kebutuhan untuk membangun sistem yang dapat dikembangkan secara bertahap atau modular, mudah dikembangkan, serta memiliki dokumentasi dan komunitas yang kuat.

Laravel merupakan kerangka kerja berbasis PHP yang menggunakan arsitektur *Model-View-Controller* (MVP). Arsitektur ini memisahkan logika data (*model*), antarmuka pengguna (*view*), dan pengontrol alur (*controller*), sehingga

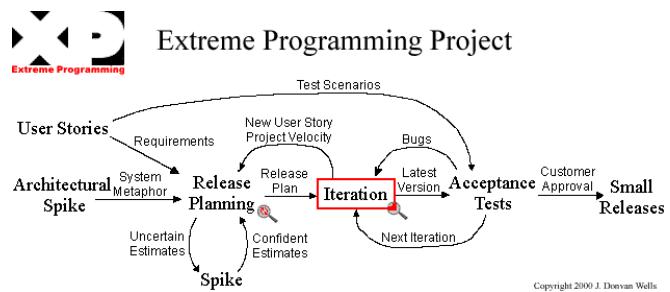
memudahkan pengelolaan kode dan pemeliharaan sistem [12]. Laravel juga menyediakan fitur seperti *Eloquent ORM* untuk interaksi basis data, *Blade templating* untuk penyusunan tampilan dinamis, serta *Artisan CLI* untuk mengotomatisasi tugas pengembangan.

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang mendukung transaksi *multi-user* dan bersifat *open source*. MySQL dipilih karena kemampuannya dalam menangani data berskala besar secara cepat, stabil, dan terintegrasi langsung dengan Laravel melalui *Eloquent* [13].

Tailwind CSS digunakan sebagai kerangka kerja antarmuka pengguna berbasis utilitas. Berbeda dari *framework* berbasis komponen seperti Bootstrap, Tailwind memberikan fleksibilitas kepada pengembang untuk menyusun desain berdasarkan kombinasi kelas utilitas sesuai kebutuhan desain antarmuka [14].

C. Metode Pengembangan Perangkat Lunak

Metode pengembangan yang digunakan dalam penelitian ini adalah *Extreme Programming* (XP), salah satu pendekatan dalam metode Agile yang dikembangkan oleh Kent Beck. XP dirancang untuk menangani perubahan kebutuhan sistem dengan pendekatan iterative, kolaboratif, dan fokus pada kualitas perangkat lunak [2], [15], [16].



GAMBAR 1
(TAHAPAN DALAM METODE EXTREME PEROGRAMMING (XP))

XP memiliki lima prinsip utama: komunikasi, kesederhanaan, umpan balik, keberanian, dan saling menghargai. Kelima prinsip tersebut diterapkan dalam empat tahap inti:

1. *Planning*, yaitu tahap ini dimulai dengan menyusun *user stories* dari hasil observasi dan wawancara dengan pengguna. *User stories* menggambarkan fitur yang dibutuhkan dari sudut pandang pengguna, dan digunakan untuk menyusun prioritas pengembangan. Informasi yang diperoleh dari *user stories* juga digunakan dalam penyusunan *release planning*, guna menentukan urutan rilis fitur secara bertahap dalam tiap iterasi pengembangan.
2. *Design*, yaitu tahap perancangan sistem berdasarkan prinsip *Keep It Simple*, dengan fokus pada kebutuhan saat ini. Desain diagram UML seperti *use case diagram*, *class diagram*, dan *activity diagram* disusun untuk menggambarkan fungsionalitas sistem secara jelas dan terstruktur.
3. *Coding*, yaitu tahap penulisan kode berdasarkan urutan prioritas *user stories*. Kode dikembangkan secara bertahap menggunakan *framework* Laravel dan Tailwind CSS, dengan pengelolaan data menggunakan MySQL.

4. *Testing*, yaitu tahap pengujian otomatis (*unit testing*) dan pengujian penerimaan (*acceptance test*) berdasarkan *user stories*. Pengujian dilakukan pada setiap iterasi untuk memastikan fungsionalitas sistem sesuai kebutuhan pengguna. Selain itu, dilakukan juga *usability testing* untuk mengevaluasi kemudahan penggunaan sistem dari sudut pandang berbagai peran pengguna yang terlibat.

XP digunakan karena fleksibel terhadap dinamika kebutuhan pengguna di lapangan dan sesuai untuk tim dengan sumber daya pengembangan terbatas.

D. Metode Pengujian Sistem

Pengujian perangkat lunak diperlukan untuk memastikan bahwa sistem yang dikembangkan telah berjalan sesuai dengan spesifikasi fungsional dan kebutuhan pengguna. Tiga pendekatan utama yang digunakan dalam penelitian ini adalah *blackbox testing*, *whitebox testing*, dan *usability testing*.

Blackbox testing merupakan metode pengujian perangkat lunak yang bertujuan untuk menguji fungsiionalitas sistem tanpa memeriksa struktur internal kode program. Pendekatan ini dilakukan berdasarkan spesifikasi sistem, dengan tujuan memverifikasi bahwa semua fungsi berjalan sesuai harapan [17], [18]. Pengujian dilakukan dengan merancang dan mengeksekusi kasus uji berdasarkan skenario pengguna, mencatat output aktual, membandingkannya dengan output yang diharapkan, dan mencatat temuan jika terjadi ketidaksesuaian ulang untuk kasus yang gagal, termasuk *regression testing* atau pengujian berulang terhadap sistem setelah dilakukan perubahan kode guna memastikan bahwa perbaikan tidak menyebabkan gangguan pada fungsi lain dalam sistem.

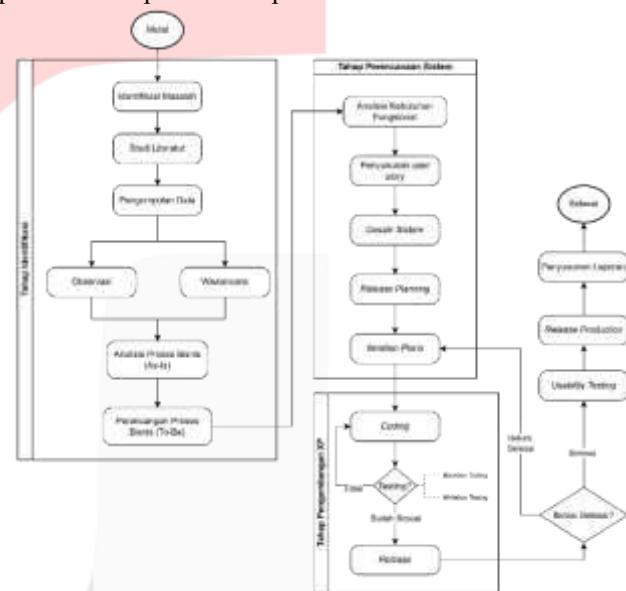
Whitebox testing adalah metode pengujian yang dilakukan dengan menganalisis logika program dan alur kontrol dalam kode untuk memastikan semua jalur eksekusi telah diuji [17], [19]. Pengujian ini cocok untuk mengidentifikasi kesalahan logika, struktur yang tidak efisien, atau cabang kode yang tidak pernah dieksekusi. Pengujian *whitebox* dilakukan pada tingkat unit, menggunakan alat bantu seperti PHPUnit untuk memverifikasi setiap fungsi dalam kode secara langsung.

Usability testing adalah metode yang digunakan untuk menilai seberapa mudah dan nyaman sistem digunakan oleh pengguna akhir. Menurut Nielsen [20], lima aspek utama yang diukur dalam *usability testing* adalah *learnability*, *efficiency*, *memorability*, *error handling*, dan *satisfaction*. *Usability testing* penting dilakukan untuk memastikan antarmuka pengguna sesuai dengan ekspektasi dan kebutuhan operasional. Pengujian ini tidak selalu membutuhkan jumlah responden yang besar, sebagian besar isu kegunaan dapat ditemukan dengan melibatkan beberapa partisipan dari masing-masing kategori pengguna [21], [22].

III. METODE

Penelitian ini menggunakan pendekatan Extreme Programming (XP), yaitu salah satu metode dalam pengembangan perangkat lunak berbasis Agile. XP dipilih karena mampu merespons perubahan kebutuhan pengguna secara cepat, dengan iterasi pengembangan pendek dan proses validasi berkelanjutan.

Alur lengkap proses pengembangan sistem ditunjukkan pada Gambar 2, yang mencakup empat tahap utama: identifikasi, perencanaan, pengembangan berbasis XP, dan evaluasi. Tahap identifikasi mencakup observasi dan wawancara, serta analisis proses bisnis *as-is* dan perancangan proses *to-be*. Tahap perencanaan sistem meliputi analisis kebutuhan fungsional, penyusunan *user stories*, desain sistem, dan penyusunan *release planning*. Tahap pengembangan dilakukan melalui iterasi, yang melibatkan pengkodean, pengujian, dan rilis bertahap, terakhir evaluasi akhir untuk *usability testing* sebelum masuk ke tahap produksi dan penulisan laporan.



GAMBAR 2

SISTEMATIKA PENYELESAIAN MASALAH

A. Identifikasi (*Planning*)

Pada tahap ini, kebutuhan pengguna dikumpulkan melalui observasi dan wawancara. Informasi yang diperoleh digunakan untuk menganalisis proses bisnis *as-is*, yaitu alur kerja manual yang sedang berlangsung. Hasil analisis ini kemudian digunakan untuk merancang proses bisnis *to-be*, sebagai dasar sistem yang akan dikembangkan.

B. Perancangan (*Design*)

Setelah kebutuhan pengguna dipetakan, dilakukan perumusan *user stories* dalam format:

"Sebagai [peran], saya ingin [tujuan] agar dapat [manfaat]."

Setiap *user story* diprioritaskan untuk membentuk *release plan*, yaitu rencana rilis bertahap yang dijalankan dalam iterasi pengembangan. Diagram yang digunakan dalam tahap ini mencakup *use case diagram* dan *activity diagram* untuk memetakan interaksi pengguna dengan sistem.

Desain sistem dilakukan dengan prinsip *Keep It Simple*, hanya mencakup fitur yang dibutuhkan dalam fase iterasi

berjalan. Beberapa jenis diagram digunakan untuk memvisualisasikan struktur dan perilaku sistem, di antaranya *use case* diagram, *entity relationship* diagram (ERD), *class* diagram, *activity* diagram, dan *robustness* diagram. Diagram ini membantu tim pengembang memahami konteks dan interaksi sistem secara lebih sistematis. Perancangan database dilakukan dengan menyusun skema relasional berdasarkan hasil ERD.

C. Pengkodean (*Coding*)

Pengembangan kode dilakukan secara bertahap berdasarkan fitur yang telah diprioritaskan dalam setiap iterasi. Bahasa pemrograman yang digunakan adalah PHP, dengan framework Laravel. Laravel dipilih karena mendukung arsitektur *Model-View-Controller* (MVC) dan menyediakan fitur *middleware*, *route*, *migration*, serta *unit testing* yang mempercepat proses pengembangan dan pengujian. Basis data yang digunakan adalah MySQL, sementara antarmuka pengguna dikembangkan menggunakan Tailwind CSS untuk menjaga konsistensi tampilan dan efisiensi pemrograman antarmuka.

D. Pengujian (*Testing*)

Pengujian dilakukan secara berlapis pada setiap iterasi, menggunakan tiga pendekatan.

1. *Blackbox testing* digunakan untuk menguji fungsionalitas sistem tanpa memperhatikan struktur kode, dengan mengacu pada spesifikasi kebutuhan pengguna [17].
2. *Whitebox testing* diterapkan menggunakan PHPUnit untuk menguji alur logika kode dan memastikan setiap fungsi berjalan sesuai alur program [19].
3. *Usability testing* dilakukan pada akhir iterasi untuk mengevaluasi kemudahan penggunaan sistem. Pengujian mencakup lima aspek Nielsen: *learnability*, *efficiency*, *memorability*, *error recovery*, dan *satisfaction* [20]. Pada penelitian ini pengujian *usability* dilakukan dengan menggunakan data kualitatif melalui wawancara terhadap perwakilan pengguna dari setiap peran. Pertanyaan disusun berdasarkan prinsip evaluasi antarmuka dan dikembangkan secara tematik untuk menangkap persepsi dan pengalaman pengguna setelah mencoba sistem.

E. Rilis (*Release*)

Setiap iterasi menghasilkan rilis internal (*small release*) yang mencakup satu atau beberapa fitur fungsional. Rilis diberikan kepada pengguna untuk diuji secara langsung, sehingga masukan dapat diperoleh sebelum fitur tersebut masuk ke tahap produksi. Proses ini memungkinkan penyesuaian sistem secara berkelanjutan dan mengurangi resiko pengembangan yang tidak sesuai kebutuhan pengguna.

F. *Release Product* dan Penyusunan Laporan

Setelah seluruh iterasi selesai dan sistem diuji menggunakan *usability testing*, sistem dirilis ke lingkungan produksi (*release production*). Sistem kemudian di-deploy pada server internal organisasi dan mulai digunakan oleh pengguna atau pegawai. Tahap akhir dari penelitian ini

adalah penyusunan laporan yang mendokumentasikan seluruh proses pengembangan, hasil pengujian, serta evaluasi sistem yang telah diimplementasikan.

IV. HASIL DAN PEMBAHASAN

Penelitian ini mengikuti pendekatan *Extreme Programming* (XP) dalam mengembangkan sistem informasi pengelolaan inventaris barang habis pakai. Proses pengembangan dibagi ke dalam empat tahapan utama, yaitu identifikasi, perencanaan, pengembangan sistem, dan evaluasi sistem. Berikut uraian dari masing-masing tahapan.

A. Identifikasi (*Planning*)

Pada tahap ini peneliti melakukan kegiatan observasi dan waancara yang dilakukan secara langsung kepada pihak-pihak yang terlibat dalam pengelolaan barang habis pakai di lingkungan kerja. Tujuan dari proses ini adalah untuk mengidentifikasi alur kerja yang sedang berjalan dan mengumpulkan informasi terkait kendala dan kebutuhan pengguna.

Informasi yang diperoleh dari hasil wawancara dan observasi dilakukan analisis untuk menyusun proses bisnis eksisting (*As-Is*), yang menggambarkan alur permintaan barang, distribusi, pencatatan, hingga pelaporan. Berdasarkan temuan tersebut, dilakukan perancangan proses bisnis yang diusulkan (*To-Be*), dengan pendekatan digital dan berbasis sistem informasi terintegrasi.

B. Perancangan (*Design*)

1. *User Story* dan *Release Plan*

Berdasarkan wawancara dengan *stakeholder* dan analisis proses bisnis, dilanjutkan dengan membuat *user stories* untuk membuat pemetaan fitur yang akan dikembangkan. Berikut *user stories* yang menjadi dasar pengembangan sistem antara lain:

TABEL 1
(DAFTAR USER STORY)

Aktor	User Story
Semua pengguna	Sebagai pengguna, saya ingin login ke dalam sistem agar dapat mengakses fitur sesuai peran saya.
Semua pengguna	Sebagai pengguna, saya ingin mengajukan permintaan barang secara digital agar proses pengadaan lebih terdokumentasi.
Semua pengguna	Sebagai penanggung jawab gudang, saya ingin memverifikasi permintaan dan melihat riwayat distribusi agar proses penyaluran barang dapat dikendalikan.
Pengurus Barang	Sebagai pengurus barang, saya ingin mencetak dokumen penyaluran agar proses distribusi tercatat secara administratif.
Administrator	Sebagai administrator, saya ingin menambah akun pengguna dan mengatur hak akses agar sistem dapat digunakan oleh seluruh pengguna sesuai perannya.

Aktor	User Story
Semua pengguna	Sebagai pengguna, saya ingin mencetak nota permintaan atau dokumen pendukung lainnya agar memiliki bukti administratif atas transaksi yang dilakukan.
Penanggung Jawab Gudang	Sebagai penanggung jawab gudang, saya ingin melihat dan memverifikasi daftar permintaan barang agar proses distribusi dapat dilakukan secara akurat.
Pengurus barang	Sebagai pengurus barang, saya ingin memvalidasi permintaan yang telah diverifikasi agar proses distribusi sesuai dengan alur yang ditentukan.
Semua pengguna	Sebagai pengguna, saya ingin memindai QR Code pada barang untuk melihat riwayat transaksi agar informasi inventaris dapat diakses secara cepat.
Administrator	Sebagai administrator, saya ingin mengunggah data barang dan akun secara massal agar proses input data menjadi lebih efisien.

Seluruh *user stories* tersebut digunakan sebagai dasar dalam menyusun *release plan* dan menentukan iterasi pengembangan selama proses pengembangan sistem berbasis metode Extreme Programming (XP). Berikut hasil penyusunan contoh hasil *iteration plan*.

TABEL 2
(PENENTUAN ITERASI)

Iterasi	User Story ID	Story Point (SP)	Total SP
1	US-01	2	15
	US-24	3	
	US-25	3	
	US-26	5	
	US-22	1	
	US-31	1	
2	US-21	5	16
	US-23	3	
	US-27	3	
	US-28	5	
3	US-11	5	17
	US-12	5	
	US-30	2	
	US-13	5	
4	US-03	5	16
	US-04	5	
	US-05	3	
	US-02	3	

Iterasi	User Story ID	Story Point (SP)	Total SP
5	US-07	5	16
	US-08	3	
	US-09	3	
	US-06	2	
	US-10	3	
6	US-16	5	18
	US-17	3	
	US-14	5	
	US-29	5	
7	US-18	5	12
	US-19	2	
	US-20	3	
	US-15	2	

Penentuan *story point* (SP) dilakukan untuk mengukur kompleksitas dan beban kerja dari setiap *user story*. Proses estimasi ini menggunakan pendekatan angka Fibonacci (1, 2, 3, 5, 8, dst) yang umum digunakan dalam metode Extreme Programming (XP). Angka Fibonacci dipilih karena memberikan toleransi terhadap ketidakpastian yang semakin besar seiring bertambahnya kompleksitas fitur.

Setiap iterasi dirancang untuk menghasilkan fitur yang dapat langsung diuji oleh pengguna. Misalnya, iterasi pertama berfokus pada pembangunan fitur dasar sistem seperti login, pengelolaan akun pengguna, dan manajemen data entitas pokok. Iterasi berikutnya mulai membangun fitur-fitur pengajuan, pengelolaan, hingga pencetakan dokumen dan integrasi QR Code.

2. Perancangan Sistem

Tahap perancangan sistem dilakukan setelah *user story* dan *iteration plan* disusun. Fokus utama dalam tahap ini adalah merancang struktur dan alur sistem berdasarkan kebutuhan fungsional yang sudah berhasil diidentifikasi dalam setiap iterasi.

Setelah *iteration plan* ditentukan, langkah selanjutnya adalah merancang sistem berdasarkan kebutuhan pada tiap iterasi. Tahap ini mencakup pemodelan menggunakan diagram UML untuk memvisualisasikan alur kerja sistem dan arsitektur sistemnya. Diagram yang digunakan meliputi:

- Use case diagram, untuk menggambarkan relasi antara aktor dalam sistem pengelolaan inventaris barang habis pakai dengan fungsionalitas utama sistem.
- Activity diagram, untuk menggambarkan alur aktivitas dari berbagai proses dalam sistem, seperti proses login, pengajuan permintaan, dan verifikasi permintaan.
- Entity Relationship Diagram (ERD) dan Class Diagram, untuk mendefinisikan struktur basis data serta hubungan antar entitas atau kelas.

- Robustness Diagram, untuk menjembatani antara *use case* dan *class diagram*, menekankan pada peran *boundary*, *control*, dan *entity*.
- *Sequance Diagram*, untuk menggambarkan urutan iteraksi antara objek dalam sistem dengan konteks proses tertentu, seperti alur pengajuan permintaan barang dari awal hingga selesai.

Pemodelan ini dirancang secara modular untuk memastikan setiap bagian sistem dapat dikembangkan secara bertahap dan independen sesuai prioritas pada *iteration plan*. Dengan demikian, perancangan sistem tidak hanya mendukung kelancaran proses implementasi pada setiap iterasi, tetapi juga menjaga fleksibilitas sistem untuk perubahan yang mungkin terjadi selama pengembangan.

3. Pengkodean

Tahap pengkodean dilakukan berdasarkan prioritas fitur yang telah ditentukan pada perancangan iterasi. Pada setiap iterasi, tim pengembang mulai menuliskan kode sesuai *user story* yang telah dirumuskan dan dirancang pada tahap sebelumnya. Proses pengkodean mengikuti prinsip *Keep It Simple*, dengan fokus pada penyelesaian kebutuhan utama pengguna tanpa menambahkan kompleksitas yang tidak diperlukan.

Sebagai contoh, pada iterasi pertama, fitur yang dikembangkan mencakup kebutuhan dasar sistem seperti autentikasi pengguna, pengelolaan kategori barang, pengelolaan unit kerja, pengelolaan data barang persediaan, serta pengaturan akun pengguna. Tabel 3 merangkum implementasi fitur berdasarkan user story yang telah dirumuskan pada tahap perencanaan.

TABEL 3
(CONTOH FITUR YANG DIKEMBANGKAN PADA ITERASI 1)

US-ID	Kebutuhan	Fitur yang Dikembangkan
US-01	Pengguna dapat masuk ke sistem dengan aman	Halaman login dengan autentikasi peran
US-24	Admin dapat mengelola kategori barang	Daftar kategori, tambah, dan hapus kategori
US-25	Admin dapat mengelola data unit kerja	Daftar, tambah, dan hapus unit kerja
US-26	Admin dapat mengelola data barang persediaan	Menampilkan daftar barang, tambah barang, koreksi stok, hapus barang

US-ID	Kebutuhan	Fitur yang Dikembangkan
US-22	Pengguna dapat mengganti password akun secara mandiri	Form ubah password untuk masing-masing pengguna setelah login
US-31	Pengguna dapat keluar dari sistem untuk mengakhiri sesi dengan aman	Tombol <i>logout</i> di header sistem, menghapus sesi pengguna aktif

4. Pengujian

Pengujian sistem dilakukan secara bertahap pada setiap akhir iterasi untuk memastikan bahwa fitur yang dikembangkan telah memenuhi kebutuhan pengguna sesuai user story. Pada tahap ini digunakan dua pendekatan utama, yaitu white-box testing dan black-box testing, sedangkan *usability testing* dilakukan setelah seluruh siklus iterasi selesai.

a) White-box dan Black-box Testing

Pada akhir setiap iterasi, pengujian *white-box* dilakukan oleh tim pengembang untuk memverifikasi alur logika, struktur kendali program, serta validasi kondisi dan cabang dari kode yang ditulis. Pengujian ini bertujuan untuk memastikan bahwa seluruh bagian kode berfungsi sesuai yang diharapkan dan bebas dari *error* fungsional.

Setelah pengujian internal selesai, sistem diuji menggunakan pendekatan *black-box* oleh penguji non-pengembang. *Black-box testing* digunakan untuk memverifikasi fungsionalitas dari sistem berdasarkan input dan output yang diharapkan, tanpa memeriksa struktur kode secara langsung. Setiap fitur diuji berdasarkan skenario kasus uji yang telah dirancang sebelumnya untuk memastikan bahwa sistem merespons sesuai spesifikasi.

b) Usability Testing

Setelah seluruh iterasi pengembangan diselesaikan dan seluruh fitur utama telah diimplementasikan, dilakukan *usability testing* untuk mengukur pengalaman pengguna saat berinteraksi dengan sistem. Pengujian ini mencakup aspek kemudahan penggunaan, efisiensi navigasi, serta kepuasan pengguna terhadap fungsionalitas sistem.

Pengujian dilakukan dengan melibatkan perwakilan pengguna dari berbagai peran yang terlibat dalam sistem. Responden diminta mencoba sistem sesuai tugas mereka, kemudian diwawancara untuk mengevaluasi pengalaman penggunaan mereka. Pendekatan ini memungkinkan identifikasi masalah *usability* yang tidak terdeteksi selama proses pengembangan, serta memberi masukan untuk perbaikan pada versi rilis selanjutnya.

V. KESIMPULAN

Sistem pengelolaan inventaris barang habis pakai yang dikembangkan dalam penelitian ini berhasil mengatasi sejumlah permasalahan yang sebelumnya dihadapi dalam pengelolaan manual, seperti keterbatasan informasi, proses

pengajuan yang belum terdigitalisasi, serta kurangnya dokumentasi proses administrasi. Sistem berbasis web yang dibangun memungkinkan pengguna untuk melakukan pengajuan, verifikasi, dan validasi permintaan barang secara digital.

Metode penegmbangan *Extreme Programming* (XP) diterapkan secara iterative melalui tahapan utama perencanaan, perancangan, pengkodean, dan pengujian yang berulang. Penyusunan *user story* dan *iteration plan* membantu prioritas pengembangan, sementara tahapan pengujian seperti *blackbox*, *whitebox*, dan *usability testing* digunakan untuk memastikan bahwa setiap fitur yang dikembangkan telah sesuai dengan kebutuhan pengguna. Pendekatan ini efektif untuk membantu pengembang dalam membangun sistem yang adaptif terhadap perubahan kebutuhan pengguna.

REFERENSI

- [1] D. Wahyuningsih, "Inventarisasi BMN Sebagai Upaya Tertib Administrasi dan Tertib Pengelolaan BMN." Accessed: Dec. 05, 2024. [Online]. Available: <https://www.djkn.kemenkeu.go.id/kpknl-metro/baca-artikel/13812/Inventarisasi-BMN-Sebagai-Upaya-Tertib-Administrasi-dan-Tertib-Pengelolaan-BMN.html>
- [2] Ian. Sommerville, *Software engineering*. Pearson, 2011.
- [3] Kent. Beck and Martin. Fowler, *Planning extreme programming*, First Edition. Addison-Wesley, 2001.
- [4] M. Siahaan, "PERANCANGAN APLIKASI TRACKING PESANAN BERBASIS WEB DENGAN METODE SDLC DAN EXTREME PROGRAMMING," 2023.
- [5] J. B. Satya, L. Suhery, A. A. J. Sinlae, and U. Uliyatunisa, "Pengembangan Sistem Pelayanan Publik Melalui Sistem Administrasi Kependudukan Menggunakan Metode Extreme Programming," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 3, no. 2, pp. 87–93, 2021, doi: 10.30865/json.v3i2.3592.
- [6] K. Setemen, I. G. Sudirtha, C. I. R. Marsiti, G. R. Dantes, and P. H. Suputra, "Developing inventory information system using mobile computing with quick response (2d-barcode) and geotagging," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2020. doi: 10.1088/1742-6596/1516/1/012011.
- [7] K. J. . Sousa and Effy. Oz, *Management information systems*. Cengage Learning, 2015.
- [8] A. T. Prastowo and S. Sanusi, "Implementasi Metode Pengembangan Extreme Programming Pada Sistem Informasi Pengelolaan Inventaris Aset Kantor," *Insearch: Information System Research Journal*, vol. 3, no. 01, 2023, doi: 10.15548/isrj.v3i01.5861.
- [9] N. A. Putri, P. D. Larasati, M. F. Mulya, and S. Anwar, "Sistem Informasi Inventaris Barang Berbasis Web menggunakan Codeigniter pada Pusat Pendidikan dan Pelatihan Pajak (PPPP)," *Jurnal Sistem Komputer dan Kecerdasan Buatan*, no. Vol. 7 No. 1 (2023): Volume VII-Nomor 1-September 2023, Sep. 2023, doi: <https://doi.org/10.47970/siskom-kb.v7i1.475>.
- [10] M. Ridwan Nawawi *et al.*, "RANCANG BANGUN SISTEM INFORMASI INVENTARIS FASILITAS PONDOK PESANTREN NURUL ULUM DENGAN MENGGUNAKAN METODE XP (EXTREME PROGRAMMING)," 2022.
- [11] L. Ratna Kinashih, K. Syaqilla Cut Yuna Haryono, W. Syafiyah Inayah Aulia, T. Rahayu, M. Bayu Wibisono, and P. Hari Saputro, "ANALISIS DAN PERANCANGAN SISTEM INVENTARIS BERBASIS WEB," 2024.
- [12] M. Stauffer, "Laravel Up & Running A Framework for Building Modern PHP Apps," 2019. [Online]. Available: www.EBooksWorld.ir
- [13] A. Hidayat, A. Yani, P. Studi Sistem Informasi, and S. Mahakarya, "MEMBANGUN WEBSITE SMA PGRI GUNUNG RAYA RANAU MENGGUNAKAN PHP DAN MYSQL," 2019.
- [14] E. Syifaudi, "Apa Itu Tailwind CSS? Keunggulan dan Cara Menggunakannya untuk Proyek Web." Accessed: Dec. 09, 2024. [Online]. Available: <https://www.exabytes.co.id/blog/apa-itu-tailwind-css-adalah/>
- [15] K. Beck, "Praise for Extreme Programming Explained, Second Edition," 2012.
- [16] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 7th ed. 2010. [Online]. Available: www.mhhe.com/pressman.
- [17] S. R. Wicaksono, *Blackbox Testing Teori dan Studi Kasus*, 1st ed. CV. Seribu Bintang, 2021. [Online]. Available: www.fb.com/cv.seribu.bintang
- [18] A. Hasanuddin, "Rancang Bangun Sistem Informasi Persediaan Bahan Habis Pakai (BHP)," *Applied Information System and Management (AISM)*, vol. 4, no. 2, pp. 83–88, Oct. 2021, doi: 10.15408/aism.v4i2.19501.
- [19] A. Andriyanto, A. Sekar Pratiwi, I. Nurdiansyah, and A. Saifudin, "Otomatisasi Pengujian Aplikasi POS (Point Of Sale) Menggunakan Metode White Box," *OKTAL : Jurnal Ilmu Komputer dan Science*, vol. 2, no. 6, 2023.
- [20] S. R. Wicaksono and V. G. Ananta, "Usability Testing Situs Daring Sebagai Evaluasi User Experience (Studi Kasus Situs PERUMDAM Among Tirto)," *Jurnal Sistem dan Informatika (JSI)*, vol. 17, no. 1, pp. 13–17, Apr. 2023, doi: 10.30864/JSI.V17I1.452.
- [21] M. Mader, "Sample Size in Usability Tests and User Interviews - Less is More? • Centigrade GmbH." Accessed: Jul. 17, 2025. [Online]. Available: <https://www.centigrade.de/en/blog/sample-size-in-usability-tests-and-user-interviews-less-is-more/>
- [22] S. R. Wicaksono, *Usability Testing*. CV. Seribu Bintang, 2023. Accessed: Dec. 18, 2024. [Online]. Available: https://www.researchgate.net/publication/369040195_Usability_Testing