

Pengembangan Sistem *Supply-Chain Planning* pada Organisasi Perusahaan Berbasis Web Menggunakan Metode *Agile Scrum*

Christhoper Kloze, Tjahjo Dwi Nurti, Miftah Andriansyah

Universitas Gunadarma, Indonesia

E-mail: christhoperkp@gmail.com

ABSTRAK

Perkembangan teknologi informasi telah memicu transformasi dalam berbagai aspek pekerjaan, termasuk divisi perencanaan di perusahaan. TOPS, sebuah perusahaan produsen trafo, menghadapi tantangan dalam proses perencanaan dan penjadwalan produksi yang mengandalkan platform konvensional seperti *Microsoft Excel*, *Outlook*, dan *WhatsApp Messenger*. Metode ini menyebabkan ketidakefisienan, seperti ketergantungan pada unggahan manual *file Excel* ke server dan penggunaan sistem makro yang memakan waktu lama. Selain itu, kurangnya acuan baku dalam penentuan proses produksi menyebabkan perubahan deadline yang sering terjadi. Untuk mengatasi masalah ini, TOPS membentuk sebuah task force untuk mendigitalisasikan sistem perencanaan melalui pengembangan sistem *supply-chain planning* berbasis *website* menggunakan *framework Laravel* dan *Vue.js*. Sistem ini dirancang untuk meningkatkan efisiensi dan akurasi dalam pembuatan penjadwalan produksi trafo. Proses pengembangan menggunakan metode *Agile Scrum*, yang memungkinkan tim untuk beradaptasi dengan perubahan kebutuhan secara cepat dan iteratif. Hasil penelitian menunjukkan bahwa sistem *supply-chain planning* berhasil dikembangkan dengan menggunakan bahasa pemrograman PHP, *framework Laravel* untuk *backend*, *Vue.js* untuk *frontend*, dan MySQL sebagai *database*. Pengujian menggunakan metode *blackbox testing* menunjukkan bahwa semua skenario pengujian berjalan sesuai harapan. *User acceptance testing* menunjukkan bahwa sistem diterima dengan baik, dengan persentase kepuasan 85,73%. Sistem ini tidak hanya berjalan tanpa error, tetapi juga dianggap efektif dalam membantu tim *supply-chain planning* merencanakan dan mengelola bahan baku secara lebih efisien dan nyaman digunakan. Dengan demikian, sistem ini berhasil memenuhi tujuan untuk mendigitalisasikan dan meningkatkan efisiensi proses perencanaan di TOPS.

Kata Kunci: Rantai Pasokan; PHP; Laravel; MySQL; Situs Web, Vue.js, Agile Scrum.

ABSTRACT

The development of information technology has triggered a transformation in various aspects of work, including the planning division in the company. TOPS, a transformer manufacturing company, faces challenges in the production planning and scheduling process that relies on conventional platforms such as Microsoft Excel, Outlook, and WhatsApp Messenger. This method causes inefficiencies, such as dependence on manual uploads of Excel files to the server and the use of a time-consuming macro system. In addition, the lack of standard references in determining the production process causes frequent changes in deadlines. To overcome this problem, TOPS formed a task force to digitize the planning system by developing a website-based supply-chain planning system using the Laravel and Vue.js frameworks. This system is designed to increase efficiency and accuracy in making transformer production schedules. The development process uses the Agile Scrum method, which allows the team to adapt to changing needs quickly and iteratively. The results of the study showed that the supply-chain planning system was successfully developed using the PHP programming language, the Laravel framework for the backend, Vue.js for the frontend, and MySQL as the database. Testing using the blackbox testing method showed that all test scenarios ran as expected. User acceptance

testing showed that the system was well received, with a satisfaction percentage of 85.73%. This system not only runs without errors, but is also considered effective in helping the supply-chain planning team plan and manage raw materials more efficiently and conveniently. Thus, this system has succeeded in meeting the goal of digitizing and increasing the efficiency of the planning process at TOPS.

Keywords: Supply-Chain, PHP, Laravel, MySQL, Website, Vue.js, Agile Scrum.



This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International

PENDAHULUAN

Perkembangan teknologi informasi dalam beberapa dekade terakhir telah menciptakan peluang baru, termasuk di divisi perencanaan. TOPS, perusahaan produsen trafo, fokus pada produksi trafo berkapasitas lebih besar dengan teknologi manufaktur terkini untuk mendukung elektrifikasi, urbanisasi, dan digitalisasi.

Meskipun telah memperluas produk dan layanan, sistem perencanaan yang menggunakan *Microsoft Excel*, *Outlook*, dan *WhatsApp* menyebabkan ketidakefisienan. Jika jadwal tidak diunggah ke server, divisi produksi tidak dapat mengaksesnya. Penggunaan makro di *Excel* memakan waktu, dan kurangnya acuan baku dalam penentuan proses produksi serta deadline sering mengakibatkan perubahan deadline pembuatan trafo.

Untuk mengatasi ketidakefisienan dalam proses perencanaan di TOPS, pengembangan sistem baru diperlukan. Dua metodologi utama yang dapat digunakan adalah *Waterfall* dan *Agile Scrum*. *Waterfall* menggunakan pendekatan linear, di mana setiap tahap diselesaikan secara berurutan, namun metode ini kurang fleksibel terhadap perubahan. Sebaliknya, *Agile Scrum* menawarkan fleksibilitas dengan pendekatan iteratif melalui sprint, memungkinkan tim untuk beradaptasi cepat terhadap perubahan kebutuhan bisnis dan *feedback* pengguna. Karena fleksibilitas dan kolaborasi yang tinggi, *Agile Scrum* lebih cocok untuk pengembangan sistem di TOPS, memastikan peningkatan efisiensi dan produktivitas di divisi perencanaan dan produksi.

Beberapa penelitian telah dilakukan mengenai penerapan metode *Agile* dalam pengembangan sistem informasi dan manajemen proyek, namun sedikit yang fokus pada implementasi dalam konteks perencanaan rantai pasokan di industri manufaktur. Urgensi penelitian ini terletak pada kebutuhan untuk meningkatkan efisiensi dan produktivitas dalam perencanaan produksi. Dengan memahami dan menerapkan teknologi modern, perusahaan dapat mengoptimalkan proses kerja mereka.

Berdasarkan permasalahan tersebut, TOPS membuat sebuah resolusi bisnis yang dibentuk menjadi sebuah *task force* untuk mendigitalisasikan beberapa sistem, salah satunya dinamakan sistem *supply-chain planning* berbasis *website* menggunakan *framework laravel* serta *vue.js* yang dapat mengefisiensikan waktu pembuatan penjadwalan produksi trafo. Selain itu, pengembangan sistem menggunakan metode *agile scrum* untuk memastikan bahwa proses pengembangan berjalan secara iteratif dan kolaboratif, sehingga memungkinkan tim untuk cepat beradaptasi dengan perubahan kebutuhan dan mempercepat penyampaian fitur yang bernilai bagi pengguna.

Tujuan dari penelitian ini adalah untuk membuat serta mengembangkan sistem *supply-chain planning* berbasis *website* yang dapat membantu perusahaan dalam mendigitalisasikan sistem pada divisi perencanaan yang sudah berjalan. Manfaat dari penelitian ini diharapkan dapat memberikan solusi bagi perusahaan dalam meningkatkan efisiensi dan akurasi proses perencanaan, serta mengurangi ketergantungan pada sistem manual.

METODE

Metode yang digunakan dalam penelitian ini adalah metode *agile scrum*. Metode *agile scrum* merupakan sebuah metode pengembangan *software* dengan berbagai proses kecil berulang. Aktivitas yang dilakukan pada *scrum* yaitu *product backlog*, *sprint planning*, *sprint backlog* dan *sprint review*. Langkah-langkah dalam metode *agile scrum* di dalam pengembangan perangkat lunak yaitu *Product Backlog*, *Sprint*, *Daily Scrum*, dan *Sprint Review*. Metode pengumpulan data dalam penelitian ini meliputi Observasi yaitu pengamatan langsung terhadap sistem yang berjalan di Divisi Perencanaan untuk mengevaluasi dan memperbaiki kelengkapan data. Kuesioner yaitu penyebaran pertanyaan terkait aplikasi kepada Divisi Perencanaan melalui kuesioner online.

Perencanaan

Perencanaan sistem Supply Chain Planning dimulai dengan identifikasi masalah di Divisi Perencanaan TOPS, seperti kesulitan dalam manajemen jadwal produksi akibat penggunaan makro Excel yang memakan waktu lama. Sistem ini akan dikembangkan menggunakan PHP, HTML, CSS, JavaScript, dengan Laravel untuk backend dan Vue.js untuk frontend.

Perancangan

Perancangan sistem *supplychain planning* di TOPS diwujudkan dengan menggunakan struktur navigasi, diagram UML yaitu *use case diagram*, *activity diagram*, *class diagram*, rancangan database dan rancangan tampilan sistem menggunakan *wireframe*.

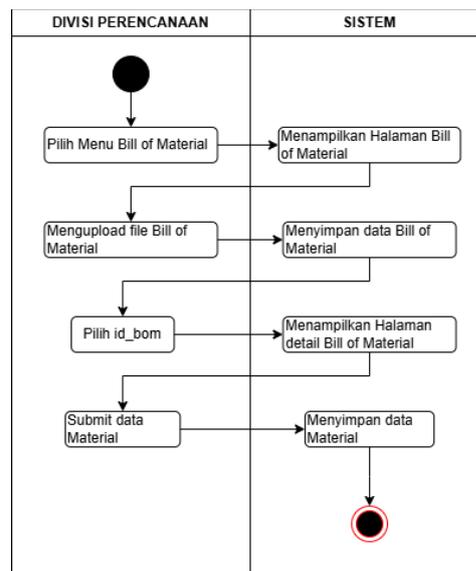
1. Use Case Diagram

Divisi perencanaan dalam pengelolaan data *Bill of Material* dapat melakukan *upload data excel bill of material*, melakukan edit data *bill of material*, serta melakukan *delete data bill of material*.

2. Activity Diagram

Activity diagram pada sistem *supplychain planning* terdapat pada table dibawah ini:

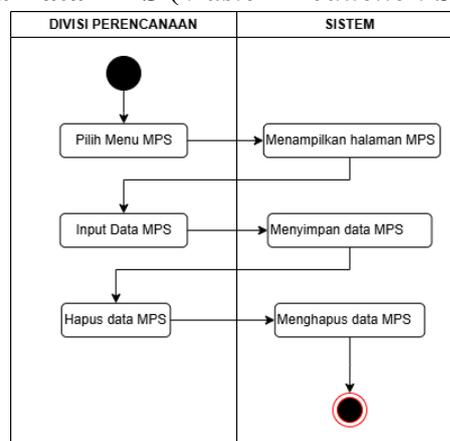
a. Activity Diagram Kelola Data *Bill of Material*



Gambar 1. Activity Diagram Kelola Data *Bill of Material*

Berdasarkan gambar 1, jika material yang dibutuhkan oleh *Bill of Material* belum terpenuhi maka divisi perencanaan tidak dapat menekan tombol *submit*. Apabila material yang dibutuhkan oleh *Bill of Material* telah terpenuhi maka divisi perencanaan dapat menekan tombol *submit* sehingga sistem dapat menyimpan data Material dan akan mengurangi *stock* dari material yang dibutuhkan.

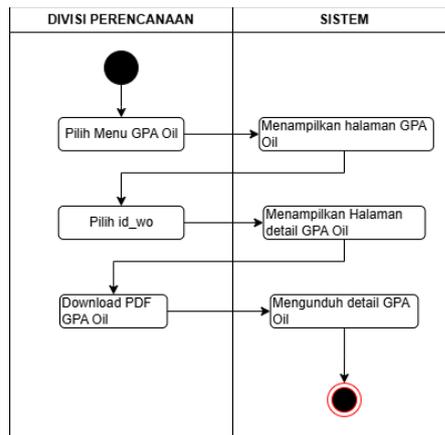
b. Activity Diagram Kelola Data MPS (*Master Production Schedule*)



Gambar 2. Activity Diagram Kelola Data MPS

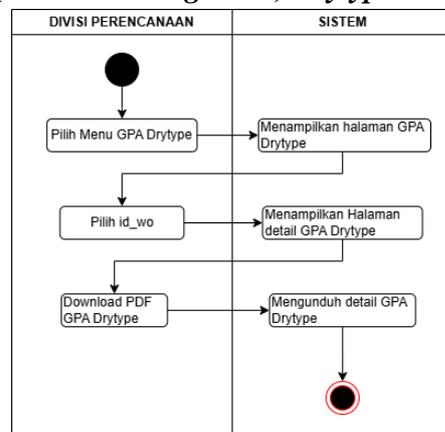
Berdasarkan gambar 2, Divisi perencanaan menginput MPS yang terdiri dari *workorder code*, kode SO, *Line*, *Project Name*, *Due Date*, KVA, Qty, dan *Manhour Code*. Sistem akan menyimpan data MPS yang telah diinput oleh divisi perencanaan. Divisi perencanaan juga dapat menghapus data MPS yang nantinya sistem akan menghapus data MPS yang telah diinput oleh divisi perencanaan.

c. Activity Diagram GPA (*Global Picking Area*) Oil Trafo



Gambar 3.1 Activity Diagram GPA Oil Trafo

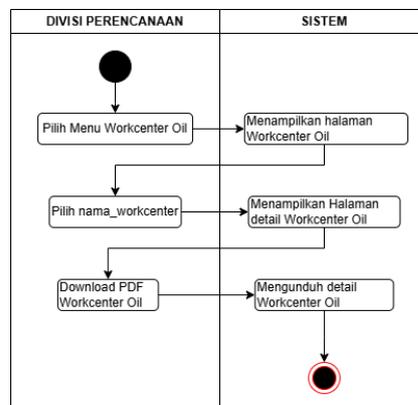
d. Activity Diagram GPA (Global Picking Area) Drytype



Gambar 4. Activity Diagram GPA Drytype

Berdasarkan gambar 4, Divisi perencanaan memilih id_wo untuk memeriksa detail GPA yang dimana sistem akan menampilkan halaman detail GPA Drytype yang berisikan start date serta deadline id_wo yang dipilih pada masing-masing workcenter. Divisi perencanaan download PDF GPA Drytype maka sistem akan mengunduh detail GPA Drytype.

e. Activity Diagram Workcenter Oil Trafo

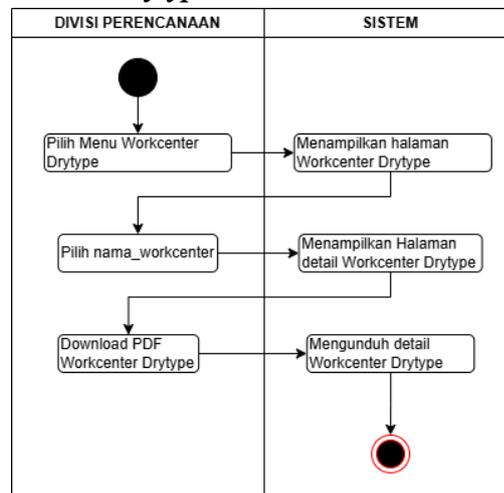


Gambar 5. Activity Diagram Workcenter Oil Trafo

Berdasarkan gambar 5, Divisi perencanaan memilih nama_workcenter untuk memeriksa detail workcenter yang dimana sistem akan menampilkan halaman detail

Workcenter Oil Trafo yang berisikan informasi mengenai *start date* dan *end date* dari masing-masing *workorder* pada *workcenter* tersebut. Divisi perencanaan *download* PDF *Workcenter Oil* maka sistem akan mengunduh *detail workcenter oil* trafo.

f. Activity Diagram *Workcenter Drytype*

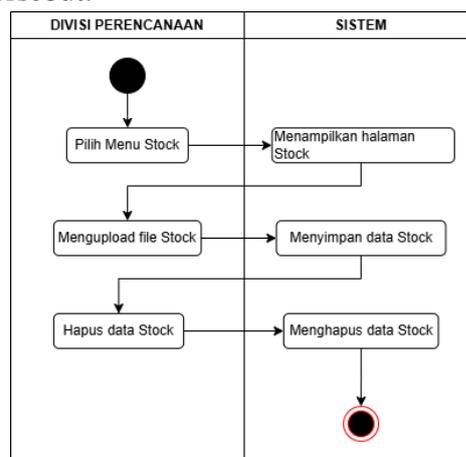


Gambar 6. Activity Diagram *Workcenter Drytype*

Berdasarkan gambar 6, Divisi perencanaan memilih *nama_workcenter* untuk memeriksa *detail workcenter* yang dimana sistem akan menampilkan halaman *detail Workcenter Drytype* yang berisikan informasi mengenai *start date* dan *end date* dari masing-masing *workorder* pada *workcenter* tersebut. Divisi perencanaan *download* PDF *Workcenter Drytype* maka sistem akan mengunduh *detail workcenter drytype*.

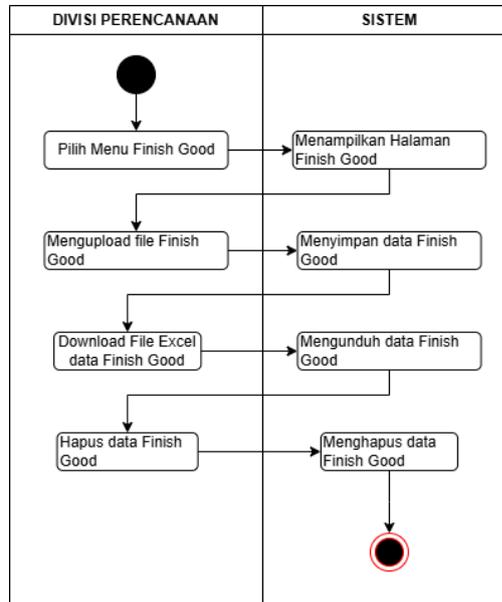
g. Activity Diagram *Kelola Data Stock*

Berdasarkan gambar 7, Divisi perencanaan mengupload *file excel stock* maka sistem akan menyimpan data *stock* yang di-*upload* oleh divisi perencanaan. divisi perencanaan menghapus data *stock* yang telah diinput sebelumnya, maka sistem akan menghapus data *stock* tersebut.



Gambar 7. Activity Diagram *Kelola Data Stock*

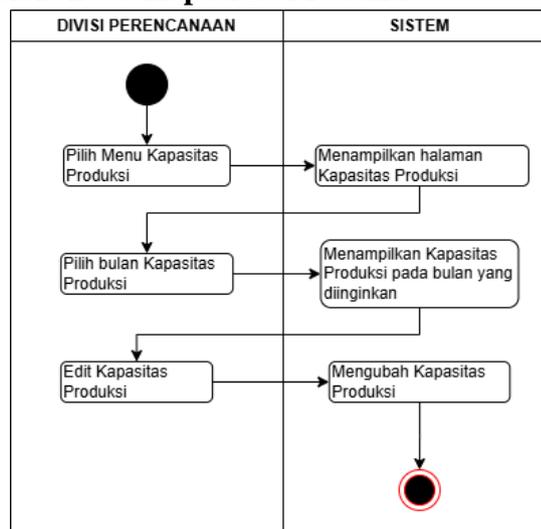
h. Activity Diagram Kelola Data *Finish Good*



Gambar 8. Activity Diagram Kelola Data *Finish Good*

Berdasarkan gambar 8, divisi perencanaan mengupload file excel finish good ke dalam sistem, sistem akan menyimpan data finish good yang telah di upload oleh divisi perencanaan. Divisi perencanaan download file excel data finish good lalu sistem akan mengunduh data finish good tersebut. Divisi perencanaan menghapus data finish good lalu sistem akan menghapus data finish good dari database.

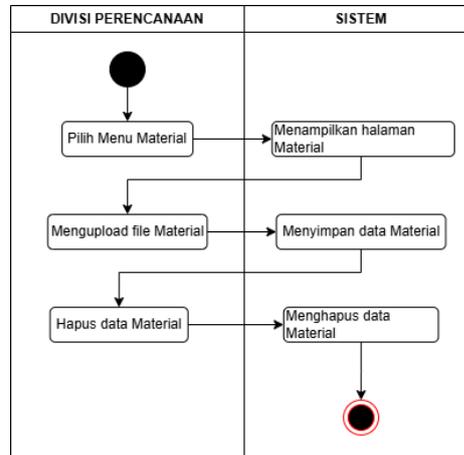
i. Activity Diagram Kelola Data Kapasitas Produksi



Gambar 9. Activity Diagram Kelola Data Kapasitas Produksi

Berdasarkan gambar 9, divisi perencanaan memilih bulan kapasitas produksi, sistem akan menampilkan informasi mengenai kapasitas produksi pada bulan yang dipilih oleh divisi perencanaan. Divisi perencanaan mengedit kapasitas produksi, maka sistem akan mengubah kapasitas produksi yang diedit oleh divisi perencanaan.

j. Activity Diagram Material



Gambar 10. Activity Diagram Material

Berdasarkan gambar 10, divisi perencanaan mengupload file excel material maka sistem akan menyimpan data material tersebut dan menampilkan pada halaman materi yang berisikan informasi seperti *item code*, *item name*, *unit*, dan *quantity*. Divisi perencanaan dapat menghapus data material yang telah di upload dan sistem akan menghapus dari database.

3. Class Diagram

Berdasarkan class diagram, terdapat beberapa kelas yang saling terhubung. *WorkcenterDryType*, *WorkcenterOilTrafo*, *GPADryType*, dan *GPAOilTrafo* terhubung dengan *Mps*. Selanjutnya, *Mps* dan *Wo* terhubung dengan *StandardizeWork*. Kelas *Wo* terhubung dengan *Bom* dan *Fgoods*, sementara *Bom* terhubung dengan *Detailbom*, yang juga terhubung dengan *Stock* dan *Material*. Setiap kelas memiliki atribut dan metode yang spesifik: *Bom* memiliki atribut seperti *id_bom*, *deskripsi*, dan *qty_bom* serta metode *create* dan *update*. *Detailbom* memiliki atribut seperti *id_boms*, *nama_workcenter*, dan *uom_material* dengan metode yang sama. *Fgoods* mencakup atribut seperti *no_transfer*, *id_wo*, dan *qty*, juga dengan metode *create* dan *update*. Sementara itu, *GPADryType* memiliki atribut seperti *id_wo*, *project*, dan *qty_trafo*, dengan metode *create*. *GPAOilTrafo* mirip dengan *GPADryType* dan juga memiliki metode *create*. Kelas *Hitung_Kapasitas* memiliki atribut seperti *nama_pl*, *oeo*, dan *shift_kerja*, serta metode *create* dan *update*. Terakhir, *Kanbanfg* mencakup atribut seperti *kode_fg*, *nama_item*, dan *stock_on_hand*, dengan metode *create* dan *update*.

HASIL DAN PEMBAHASAN

Pengujian

Pada tahap pengujian sistem *supplychain planning* ini dilakukan dengan cara *blackbox testing* serta pengujian pengguna lain.

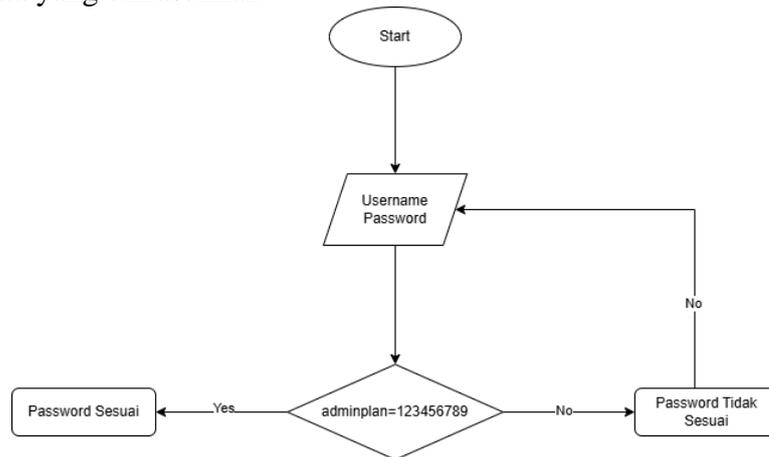
1. Whitebox Testing

Teknik basis *path* adalah metode dalam pengujian perangkat lunak untuk uji *white box* yang bertujuan mengukur kompleksitas program dengan mengidentifikasi semua jalur yang

mungkin. Proses ini meliputi beberapa langkah utama: membuat *Flow Graph*, menghitung jumlah edges dan nodes, menghitung *Cyclomatic Complexity (CC)*, dan menyusun skenario pengujian.

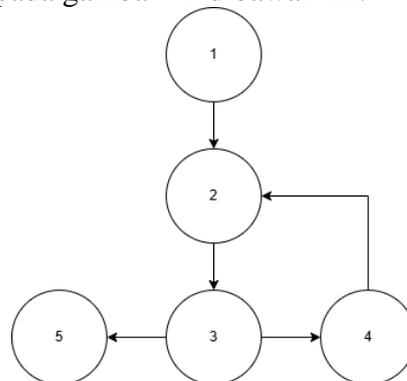
a. Pengujian Fitur Login

Flowchart dan *flowgraph* adalah representasi visual dari alur proses atau logika dalam sebuah program atau aplikasi. Pada *flowchart* basis *path* fitur *login*, terdapat langkah-langkah yang dimana seperti memasukkan *username* dan *password* hingga verifikasi data yang dimasukkan.



Gambar 11. *Flowchart Login*

Berdasarkan gambar 11, yang merupakan *flowchart login* dapat dibuat menjadi sebuah *flowgraph* seperti pada gambar 12 dibawah ini.



Gambar 12. *Flowgraph Login*

Berdasarkan gambar 12, perhitungan basis path pada fitur login dilakukan dengan rumus berikut:

Cyclomatic Complexity:

$$V(G) = E - N + 2$$

$$V(G) = 5 - 5 + 2$$

$$V(G) = 2$$

Berdasarkan perhitungan *Cyclomatic Complexity* didapatkan *independent path* yaitu suatu rangkaian arus yang harus dilalui yang dimana terdapat pada tabel 1.

Tabel 1. *Independent Path Login*

No	Independent Path
1	1-2-3-4-2 (<i>User</i> memasukkan <i>username</i> dan <i>password</i> yang salah)
2	1-2-3-5 (<i>User</i> memasukkan <i>username</i> dan <i>password</i> yang benar)

Berdasarkan tabel 1, merupakan *independent* yang mencakup jalur-jalur independen yaitu 1-2-3-4-2 (skenario ketika pengguna memasukkan username dan password yang salah) serta 1-2-3-5 (skenario ketika pengguna memasukkan username dan password yang benar). Selain itu, dapat menghasilkan *test case login path 1* seperti yang terdapat pada tabel 2.

Tabel 2. Test Case Login path 1

Path	1
Jalur	1-2-3-4-2
Skenario	<ol style="list-style-type: none"> 1. Start 2. Masukkan <i>username</i> dan <i>password</i> 3. Verifikasi <i>username</i> dan <i>password</i> 4. Verifikasi <i>username</i> dan <i>password</i> tidak sesuai 2. Masukkan <i>username</i> dan <i>password</i>
Hasil Pengujian	Berhasil

Selain itu, dari tabel *independent path* terdapat *test case path 2* yang dimana terdapat pada tabel 3.

Tabel 3. Test Case Login path 2

Path	2
Jalur	1-2-3-5
Skenario	<ol style="list-style-type: none"> 1. Start 2. Masukkan <i>username</i> dan <i>password</i> 3. Verifikasi <i>username</i> dan <i>password</i> 5. Verifikasi <i>username</i> dan <i>password</i> sesuai
Hasil Pengujian	Berhasil

Untuk pengujian, dibuat *Test Case Fitur Login* yang mencakup kedua jalur ini. Pada *Path 1*, dilakukan pengujian dengan skenario pengguna memasukkan informasi login yang salah, sedangkan pada *Path 2*, pengujian dilakukan dengan skenario informasi login yang benar. Setiap test case dilengkapi dengan skenario pengujian, hasil yang diharapkan, hasil aktual, dan deskripsi mengenai input yang dimasukkan.

Berdasarkan *test case login* menghasilkan *scenario test login* yang terdapat pada tabel 4.

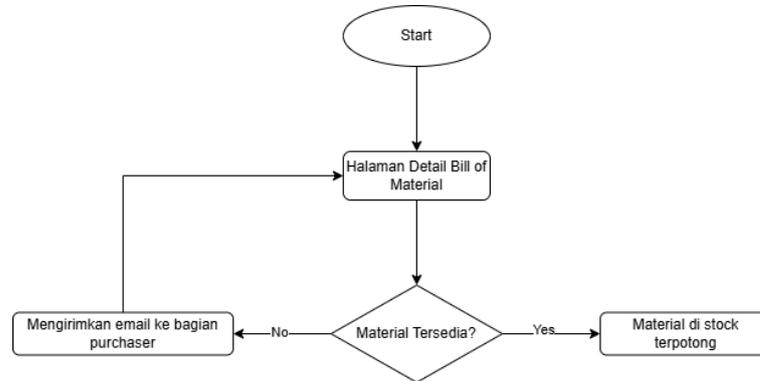
Tabel 4. Scenario Test Login

Path	2
Aktivitas	Memasukkan <i>password</i> benar dan terverifikasi
Expected Result	Menampilkan halaman <i>dashboard supplychain planning</i>
Result	Menampilkan halaman <i>dashboard supplychain planning</i>
Nilai	<i>Username</i> : adminplan <i>Password</i> : 123456789
Keterangan	Valid

Berdasarkan tabel 4 merupakan tabel *scenario test login* yaitu metode pengujian yang mendetail aktivitas selama proses login. Metode ini menyajikan hasil yang diharapkan dan membandingkannya dengan hasil aktual, serta mencatat data yang dimasukkan dan memberikan penjelasan lengkap mengenai validitas hasil pengujian. Fokus utama dalam *scenario test* ini adalah pada kondisi di mana *password* yang dimasukkan benar dan terverifikasi, serta diharapkan hasilnya adalah berhasil login ke halaman *dashboard supply chain planning*.

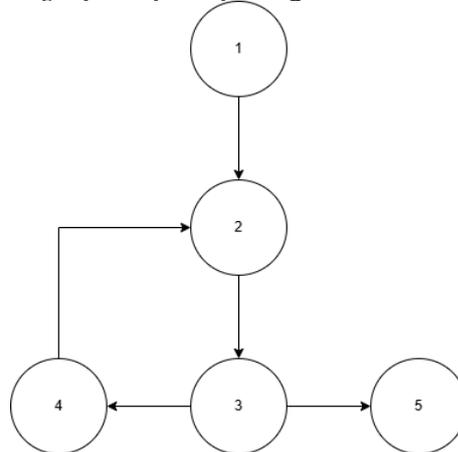
b. Pengujian Fitur Detail Bill of Material

Pada *flowchart* basis *path* fitur *detail bill of material*, terdapat langkah-langkah yang dimana seperti pengecekan data *stock* material hingga pemotongan *stock* material.



Gambar 13. *Flowchart Detail Bill of Material*

Berdasarkan gambar 13, yang merupakan *flowchart detail bill of material* dapat dibuat menjadi sebuah *flowgraph* seperti pada gambar 14 dibawah ini.



Gambar 14. *Flowgraph Detail Bill of Material*

Berdasarkan gambar 14, perhitungan basis *path* pada fitur login dilakukan dengan rumus berikut:

Berdasarkan perhitungan *Cyclometric Complexity* didapatkan *independent path* yaitu suatu rangkaian arus yang harus dilalui yang dimana terdapat pada tabel 5.

Tabel 5. *Independent Path Detail Bill of Material*

No	Independent Path
1	1-2-3-4-2 (<i>Stock</i> material tidak tersedia)
2	1-2-3-5 (<i>Stock</i> material tersedia)

Berdasarkan tabel 5 merupakan *independent* yang mencakup jalur-jalur independen yaitu 1-2-3-4-2 (skenario ketika *stock* material tidak tersedia) serta 1-2-3-5 (skenario ketika *stock* material tersedia). Selain itu, dapat menghasilkan *test case detail bill of material path* 1 seperti yang terdapat pada tabel 6.

Tabel 6. *Test Case Path 1 Detail Bill of Material*

Path	1
Jalur	1-2-3-4-2
Skenario	1. Start 2. Menekan <i>button</i> info pada <i>bill of material</i> 3. Sistem otomatis melakukan pengecekan ketersediaan <i>stock</i> material

	4. <i>Stock material</i> tidak tersedia, sistem mengirimkan email ke bagian <i>purchaser</i>
	2. Menampilkan halaman <i>detail bill of material</i>
Hasil Pengujian	Berhasil

Selain itu, dari tabel *independent path* terdapat *test case path 2* yang dimana terdapat pada tabel 7.

Tabel 7. Test Case Path 2 Detail Bill of Material

Path	2
Jalur	1-2-3-5
Skenario	1. Start 2. Menekan <i>button</i> info pada <i>bill of material</i> 3. Sistem otomatis melakukan pengecekan ketersediaan <i>stock material</i> 5. <i>Stock material</i> tersedia, sistem akan memotong <i>stock material</i>
Hasil Pengujian	Berhasil

Untuk pengujian, dibuat *Test Case* Fitur *Detail Bill of Material* yang mencakup kedua jalur ini. Pada *Path 1*, dilakukan pengujian dengan skenario pengecekan *stock material* yang tidak tersedia, sedangkan pada *Path 2*, pengujian dilakukan dengan skenario pengecekan *stock material* tersedia. Setiap test case dilengkapi dengan skenario pengujian, hasil yang diharapkan, hasil aktual, dan deskripsi yang dimasukkan.

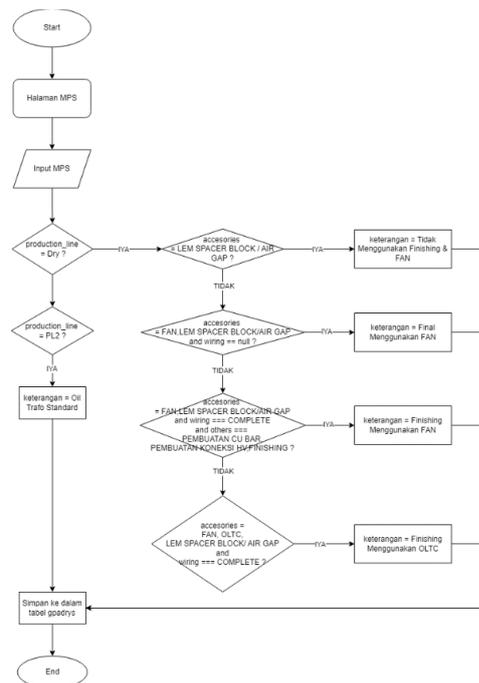
Tabel 8. Scenario Test Detail Bill of Material

Path	2
Aktivitas	Pengecekan <i>stock material</i> tersedia
Expected Result	<i>Stock material</i> yang tersedia akan berkurang
Result	<i>Stock material</i> yang tersedia akan berkurang
Nilai	<i>Stock material</i> berkurang
Keterangan	Valid

Berdasarkan tabel 8, merupakan tabel *scenario test detail bill of material* yaitu metode pengujian yang mendetail aktivitas *detail bill of material*. Metode ini menyajikan hasil yang diharapkan dan membandingkannya dengan hasil aktual, serta mencatat data yang dimasukkan dan memberikan penjelasan lengkap mengenai validitas hasil pengujian. Fokus utama dalam *scenario test* ini adalah pengecekan *stock material* tersedia atau tidak, serta diharapkan hasilnya adalah pemotongan *stock material* yang tersedia.

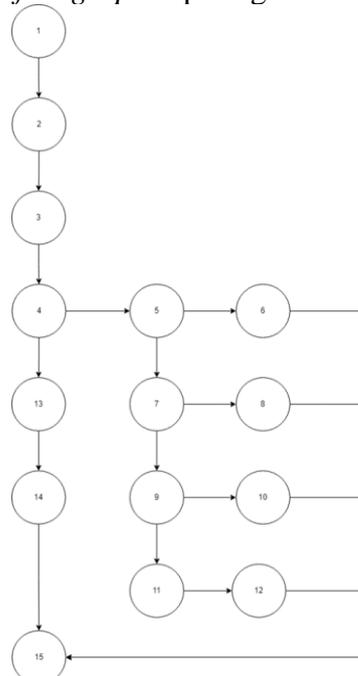
c. Pengujian Fitur *Master Production Schedule*

Flowchart dan *flowgraph* adalah representasi visual dari alur proses atau logika dalam sebuah program atau aplikasi. Pada *flowchart* basis *path* fitur *master production schedule*, terdapat langkah-langkah yang dimana seperti *user* memasukkan inputan hingga sistem menyimpan ke dalam tabel *gpadyrs*.



Gambar 15. Flowchart Master Production Schedule

Berdasarkan gambar 15, yang merupakan *flowchart master production schedule* dapat dibuat menjadi sebuah *flowgraph* seperti gambar 16 dibawah ini.



Gambar 16. Flowgraph Master Production Schedule

Berdasarkan gambar 16, perhitungan basis path pada fitur *master production schedule* dilakukan dengan rumus berikut:

Berdasarkan perhitungan *Cyclometric Complexity* didapatkan *independent path* yaitu suatu rangkaian arus yang harus dilalui yang dimana terdapat pada tabel 9.

Tabel 9. Independent Path Master Production Schedule

No	Independent Path
1	1-2-3-4-5-6-15 (Menggunakan Finishing & FAN)

No	Independent Path
2	1-2-3-4-5-7-8-15 (Final Menggunakan FAN)
3	1-2-3-4-5-7-9-10-15 (Finishing Menggunakan FAN)
4	1-2-3-4-5-7-9-11-12-15 (Finishing Menggunakan OLTC)
5	1-2-3-4-13-14-15 (Oil Trafo Standard)

Berdasarkan tabel 9 merupakan *independent* yang mencakup jalur-jalur independen yaitu 1-2-3-4-5-6-15 (skenario ketika trafo memiliki keterangan menggunakan *finishing & FAN*), 1-2-3-4-5-7-8-15 (skenario ketika trafo memiliki keterangan *final assembly* menggunakan FAN), 1-2-3-4-5-7-9-10-15 (skenario ketika trafo memiliki keterangan *finishing* menggunakan FAN), 1-2-3-4-5-7-9-11-12-15 (skenario ketika trafo memiliki keterangan *finishing* menggunakan OLTC), serta 1-2-3-4-13-14-15 (skenario ketika trafo memiliki keterangan *oil trafo standard*). Selain itu, dapat menghasilkan *test case master production schedule path 1* seperti yang terdapat pada tabel 10.

Tabel 10. Test Case Path 1 Master Production Schedule

Path	1
Jalur	1-2-3-4-5-6-15
Skenario	<ol style="list-style-type: none"> 1. Start 2. Membuka halaman MPS 3. <i>User</i> menginput inputan 4. Sistem akan mengecek apakah <i>production_line</i> adalah <i>dry</i> ? 5. Jika <i>production_line</i> adalah <i>dry</i> sistem akan mengecek apakah accessories adalah LEM SPACER BLOCK / AIR GAP? 6. Jika iya maka keterangan yang tampil di GPA adalah Tidak Menggunakan <i>Finishing & FAN</i> 15. Data akan disimpan di tabel <i>gpadrys</i>
Hasil Pengujian	Berhasil

Selain itu, dari tabel *independent path* terdapat *test case path 2* yang dimana terdapat pada tabel 11.

Tabel 11. Test Case Path 2 Master Production Schedule

Path	2
Jalur	1-2-3-4-5-7-8-15
Skenario	<ol style="list-style-type: none"> 1. Start 2. Membuka halaman MPS 3. <i>User</i> menginput inputan 4. Sistem akan mengecek apakah <i>production_line</i> adalah <i>dry</i> ? 5. Jika <i>production_line</i> adalah <i>dry</i> sistem akan mengecek apakah accessories adalah LEM SPACER BLOCK / AIR GAP? 7. Jika tidak maka sistem akan mengecek apakah accessories adalah FAN, LEM SPACER BLOCK/AIR GAP dan wiring == null ? 8. Jika iya maka keterangan yang tampil di GPA adalah <i>Final Assembly</i> menggunakan FAN 15. Data akan disimpan di tabel <i>gpadrys</i>
Hasil Pengujian	Berhasil

Selain itu, dari tabel *independent path* terdapat *test case path 3* yang dimana terdapat pada tabel 12.

Tabel 12. Test Case Path 3 Master Production Schedule

Path	3
Jalur	1-2-3-4-5-7-9-10-15
Skenario	<ol style="list-style-type: none"> 1. Start 2. Membuka halaman MPS 3. <i>User</i> menginput inputan 4. Sistem akan mengecek apakah <i>production_line</i> adalah <i>dry</i> ? 5. Jika <i>production_line</i> adalah <i>dry</i> sistem akan mengecek apakah accessories adalah LEM SPACER BLOCK / AIR GAP? 7. Jika tidak maka sistem akan mengecek apakah accessories adalah FAN,LEM SPACER BLOCK/AIR GAP dan wiring == null ? 8. Jika tidak maka sistem akan mengecek apakah accessories adalah FAN,LEM SPACER BLOCK/AIR GAP dan wiring adalah COMPLETE dan others adalah PEMBUATAN CU BAR, PEMBUATAN KONEKSI HV, FINISHING ? 9. Jika iya maka keterangan yang tampil di GPA adalah Finishing menggunakan FAN 15. Data akan disimpan di tabel <i>gpadrys</i>
Hasil Pengujian	Berhasil

Selain itu, dari tabel *independent path* terdapat *test case path 4* yang dimana terdapat pada tabel 13.

Tabel 13. Test Case Path 4 Master Production Schedule

Path	4
Jalur	1-2-3-4-5-7-9-11-12-15
Skenario	<ol style="list-style-type: none"> 1. Start 2. Membuka halaman MPS 3. <i>User</i> menginput inputan 4. Sistem akan mengecek apakah <i>production_line</i> adalah <i>dry</i> ? 5. Jika <i>production_line</i> adalah <i>dry</i> sistem akan mengecek apakah accessories adalah LEM SPACER BLOCK / AIR GAP? 7. Jika tidak maka sistem akan mengecek apakah accessories adalah FAN,LEM SPACER BLOCK/AIR GAP dan wiring adalah null? 9. Jika tidak maka sistem akan mengecek apakah accessories adalah FAN,LEM SPACER BLOCK/AIR GAP dan wiring adalah COMPLETE dan others adalah PEMBUATAN CU BAR, PEMBUATAN KONEKSI HV, FINISHING ? 11. Jika tidak maka sistem akan mengecek apakah accessories adalah FAN, OLTC, LEM SPACER BLOCK/ AIR GAP dan <i>wiring</i> adalah COMPLETE? 12. Jika iya maka keterangan yang tampil di GPA adalah Finishing Menggunakan OLTC

	15. Data akan disimpan di tabel <i>gpadrys</i>
Hasil Pengujian	Berhasil

Selain itu, dari tabel *independent path* terdapat *test case path 5* yang dimana terdapat pada tabel 14.

Tabel 14. Test Case Path 5 Master Production Schedule

Path	5
Jalur	1-2-3-4-13-14-15
Skenario	<ol style="list-style-type: none"> 1. Start 2. Membuka halaman MPS 3. <i>User</i> menginput inputan 4. Sistem akan mengecek apakah <i>production_line</i> adalah <i>dry</i> ? 13. Jika <i>production_line</i> bukan <i>dry</i> maka sistem akan mengecek apakah <i>production_line</i> adalah PL2? 14. Jika iya maka keterangan yang tampil di GPA adalah <i>Oil Trafo Standard</i> 15. Data akan disimpan di tabel <i>gpadrys</i>
Hasil Pengujian	Berhasil

Untuk pengujian, dibuat *Test Case* Fitur *Master Production Schedule* yang mencakup kelima jalur ini. Pada *Path 1*, dilakukan pengujian dengan skenario ketika trafo memiliki keterangan menggunakan *finishing & FAN*, kemudian pada *Path 2*, dilakukan pengujian dengan skenario ketika trafo memiliki keterangan *final assembly* menggunakan FAN, kemudian pada *Path 3* dilakukan pengujian dengan skenario ketika trafo memiliki keterangan *finishing* menggunakan FAN, kemudian pada *Path 4* dilakukan pengujian dengan skenario ketika trafo memiliki keterangan *finishing* menggunakan OLTC, serta pada *Path 5* dilakukan pengujian dengan skenario ketika trafo memiliki keterangan *oil trafo standard*. Setiap test case dilengkapi dengan skenario pengujian, hasil yang diharapkan, hasil aktual, dan deskripsi mengenai input yang dimasukkan.

Tabel 15. Scenario Test Master Production Schedule

Path	1 & 2
Aktivitas	Pengecekan <i>production_line</i> dan mengisi kolom keterangan
Expected Result	Jika <i>production_line</i> adalah <i>dry</i> maka keterangan yang digunakan adalah Menggunakan <i>Finishing & FAN</i> , <i>Final</i> menggunakan FAN, <i>Finishing</i> menggunakan FAN, dan <i>finishing</i> menggunakan OLTC. Lalu jika <i>production_line</i> adalah PL2 maka keterangan yang digunakan adalah <i>oil trafo standard</i>
Result	Keterangan yang digunakan menyesuaikan <i>production_line</i> yang dipilih oleh <i>user</i>
Nilai	Keterangan yang digunakan menyesuaikan <i>production_line</i> yang dipilih oleh <i>user</i>
Keterangan	Valid

Berdasarkan tabel 15 merupakan tabel *scenario test master production schedule* yaitu metode pengujian yang mendetail aktivitas *master production schedule*. Metode ini menyajikan hasil yang diharapkan dan membandingkannya dengan hasil aktual, serta mencatat data yang dimasukkan dan memberikan penjelasan lengkap mengenai validitas hasil pengujian. Fokus utama dalam *scenario test* ini adalah pengecekan keterangan yang digunakan menyesuaikan dengan *production_line* yang diinput oleh *user*.

2. Blackbox Testing

Pada blackbox testing, pengujian dilakukan berdasarkan *input* dan *output* yang diharapkan dari sistem, tanpa mengetahui detail implementasi atau kode program di baliknya. Pengujian *blackbox testing* dilakukan oleh *scrum master* yaitu seorang *supervisor* divisi perencanaan.

Hasil pengujian *blackbox testing* menunjukkan sebanyak 26 skenario pengujian mendapatkan hasil sesuai dengan yang diharapkan. Pada 26 skenario tersebut, setiap kasus pengujian aplikasi berhasil menjalankan fungsi-fungsi yang telah ditentukan dan memberikan *output* yang sesuai.

a. User Acceptance Testing

Pada pengujian *user acceptance testing*, pengujian dilakukan menggunakan kuesioner *online* yang terdiri dari lima pertanyaan kepada 30 responden. Pengujian pengguna lain bertujuan untuk mengetahui pengalaman pengguna dan seberapa suksesnya sistem berhasil dibuat. Adapun hasil dari pengujian pengguna lain terdapat pada tabel 16.

Tabel 16. Hasil Pengujian User Acceptance Test

No	Pertanyaan	1	2	3	4	5
1	Apakah tampilan dari sistem tersebut menarik?	0	0	2	1	1
					5	3
2	Apakah sistem ini mudah digunakan navigasinya jelas untuk pengguna baru dan yang sudah berpengalaman?	0	0	1	4	1
				1		5
3	Menurut Anda, apakah dengan adanya sistem ini dapat membantu tim <i>supply chain planning</i> dalam merencanakan dan mengelola bahan baku secara lebih efektif?	0	0	7	1	1
					1	2
4	Apakah anda setuju selama menggunakan menu yang terdapat pada sistem ini tidak mengalami <i>error</i> secara tiba-tiba?	0	0	6	1	1
					1	3
5	Apakah selama penggunaan website ini Anda merasa nyaman dan tidak mengalami kesulitan dalam pembuatan jadwal serta pengiriman permintaan bahan baku yang kurang?	0	0	2	1	1
					0	8

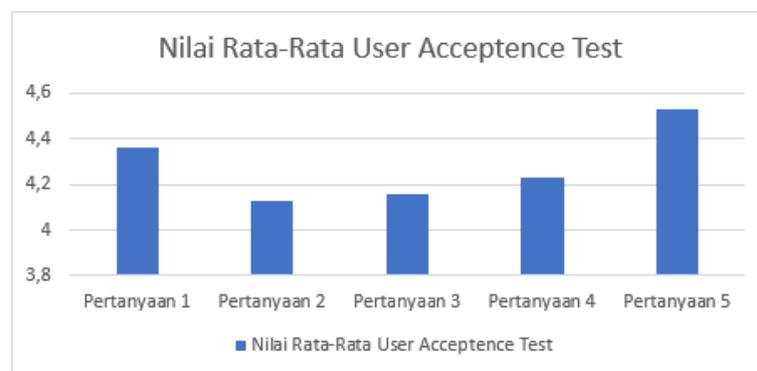
Berdasarkan tabel 16, data yang telah didapatkan dikalikan dengan jumlah bobot nilai agar mendapatkan skor pengujian. Hasil dari skor pengujian tersebut terdapat pada tabel 17.

Tabel 17. Hasil Jumlah Skor User Acceptance Test

No	Pertanyaan	1	2	3	4	5	Jumlah
1	Apakah tampilan dari sistem tersebut menarik?	0	0	2	15	13	131
2	Apakah sistem ini mudah digunakan navigasinya jelas untuk pengguna baru dan yang sudah berpengalaman?	0	0	11	4	15	124
3	Menurut Anda, apakah dengan adanya sistem ini dapat membantu tim <i>supply chain</i>	0	0	7	11	12	125

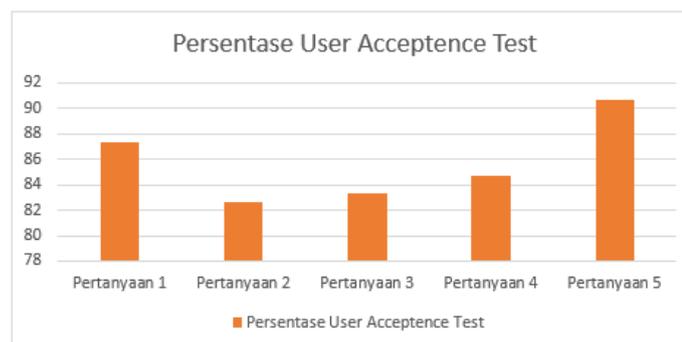
No	Pertanyaan	1	2	3	4	5	Jumlah
	<i>planning</i> dalam merencanakan dan mengelola bahan baku secara lebih efektif?						
4	Apakah selama menggunakan menu yang terdapat pada sistem ini mengalami <i>error</i> tiba-tiba?	0	0	6	11	13	127
5	Apakah selama penggunaan website ini Anda merasa nyaman dan tidak mengalami kesulitan dalam pembuatan jadwal serta pengiriman permintaan bahan baku yang kurang?	0	0	2	10	18	136

Berdasarkan tabel 17, telah didapatkan hasil jumlah skor pengujian pengguna lain.



Gambar 17. Nilai Rata-Rata *User Acceptance Test*

Berdasarkan gambar 17, terlihat bahwa pertanyaan lima yang mempunyai nilai rata-rata tertinggi dengan nilai 4.53. Disusul dengan pertanyaan 1 dengan nilai 4.36, kemudian pertanyaan 4 dengan nilai 4.23, kemudian pertanyaan 3 dengan nilai 4.16 lalu pertanyaan 2 dengan nilai 4.13.



Gambar 18. Persentase *User Acceptance Test*

Berdasarkan gambar 18, pertanyaan lima memiliki persentase tertinggi (90,67%), diikuti pertanyaan 1 (87,33%), pertanyaan 4 (84,66%), pertanyaan 3 (83,33%), dan pertanyaan 2 (82,66%). Total persentase adalah 428,65%, yang jika dibagi dengan jumlah pertanyaan (5) menghasilkan rata-rata 85,73%. Hasil analisis dari kuesioner online terhadap 30 responden menunjukkan bahwa dengan persentase 85,73% (Sangat Kuat), sistem dapat berjalan tanpa error, memiliki tampilan menarik, efektif dalam membantu tim supply chain planning, dan nyaman digunakan untuk pembuatan jadwal serta pengiriman permintaan bahan baku.

KESIMPULAN

Dapat disimpulkan bahwa sistem *supply-chain planning* berhasil dibuat menggunakan bahasa pemrograman PHP, *framework laravel* dan *framework Vue.js* pada *user interface* sistem. Sistem *supply-chain planning* berhasil dibuat menggunakan *database SQL* yaitu MySQL untuk penyimpanan data. Sistem penentuan *deadline* masing-masing *workcenter* memanfaatkan data *manhour* dari divisi produksi telah berhasil dibuat pada sistem *supply-chain planning*. Sistem *supply-chain planning* telah dilakukan pengujian menggunakan *blackbox testing* dan pengujian pengguna lain. Pada pengujian menggunakan *blackbox testing* semua skenario pengujian sesuai dengan yang diharapkan. Pada pengujian pengguna lain dengan persentase 85,73% dapat disimpulkan bahwa sistem dapat berjalan baik tanpa adanya *error* pada sistem, tampilan sistem menarik, sistem dapat membantu tim *supply chain planning* dalam merencanakan dan mengelola bahan baku secara lebih efektif, serta sistem nyaman untuk digunakan dan pengguna tidak mengalami kesulitan dalam pembuatan jadwal serta pengiriman permintaan bahan baku yang kurang.

REFERENSI

- Anis, Y., Mukti, A. B., & Rosyid, A. N. (2023, Oktober). Penerapan Model Waterfall Dalam Pengembangan Sistem Informasi Aset Destinasi Wisata Berbasis Website. *Klik : Kajian Ilmiah Informatika Dan Komputer*, 4(2), 1134-1142.
- Ariesta, A., Dewi, Y. N., Sariasih, F. A., & Fibriany, F. W. (2021, Juni). Penerapan Metode Agile Dalam Pengembangan Application Programming Interface System Pada Pt Xyz. *Jurnal Coreit*, 7(1), 38-42.
- Christian, Y., & Bisma, R. (2021). Studi Perbandingan Performa Aplikasi Web Monolitik Dan Microservice Berbasis Apache Kafka. *Journal Of Informatics And Computer Science*, 3(1), 79-88.
- Dzaky, F. A., & Kurniawan, D. (2023). Implementasi Metode Agile Framework Scrum Dalam Pengembangan Sistem Informasi Manajemen Aset Terpadu Universitas Diponegoro Modul Inventarisasi. *Jurnal Masyarakat Informatika*, 14(1), 53-69.
- Etrariadi, N., & A'inunisyah, E. S. (2023, Mei 18). Pengembangan Website Manajemen Proyek Menggunakan Metode Agile Scrum (Studi Kasus Diskopindag Kota Malang). *Jurnal Nasional Teknologi Dan Sistem Informasi*, 9(1).
- Ismanto, Hidayah, F., & Kristinanti. (2020, Februari). Pemodelan Proses Bisnis Menggunakan Business Process Modelling Notation (Bpmn) (Studi Kasus Unit Penelitian Dan Pengabdian Kepada Masyarakat (P2km) Akademi Komunitas Negeri Putra Sang Fajar Blitar). *Jurnal Riset Dan Konseptual*, 5(1), 69-76.
- Kurniawan, T., & Syarifuddin. (2020, Juli 2). Perancangan Sistem Aplikasi Pemesanan Makanan Dan Minuman Pada Cafeteria No Caffe Di Tanjung Balai Karimun Menggunakan Bahasa Pemrograman Php Dan Mysql. *Jurnal Tikar*, 1(2), 192-206.
- Maulana, T., Firdaus, & Guslendra. (2024). Perancangan Sistem Informasi Pembokongan Dan Keuangan Berbasis Web Pada Pict Story Wedding Fotografer Dengan Menggunakan Bahasa Pemrograman Php Dan Database Mysql. *Jurnal Sains Informatika Terapan*, 3(1), 20-25.
- Nurhayati, L., & Setiadi, D. (2017). Pemodelan Proses Bisnis (Studi Kasus Pd. Simpati Sumedang). *Jurnal Ilmu-Ilmu Informatika Dan Manajemen*, 11(1), 40-50.

- Pamungkas, R. P., & Khalida, R. (2019). Manajemen Proyek Agile Dengan Pendekatan Metode Scrum Sebagai Peningkatan Layanan Berkelanjutan Perusahaan. *Prosiding Seminar Nasional Sisfotek (Sistem Informasi Dan Teknologi Informasi)*, 3(1), 187-194.
- Renata, A., Zahrani, M. R., & Melandri, M. B. (2022, Maret). Analisis Efisiensi Tahapan Penggunaan Metode Manajemen Proyek Sti (Agile Scrum Dan Waterfall). *Jurnal Sistem Dan Teknologi Informasi Indonesia*, 1(2).
- Roilan, R. I., Yulianto, P. A., & Astuti, Y. (2023). Metode Agile Scrum Dalam Pembuatan Aplikasi Permohonan Informasi E-Ppid Bawaslu. *Journal Of Information System Management (Joism)*, 5(1), 64-69.
- Sansprayada, A., & Suteja, I. N. (2019). Implementasi Aplikasi Framework Laravel Studi Kasus Pt. Xyz. *Jurnal Teknik Informatika Stmik Antar Bangsa*, 18-24.
- Shidqi, M., & Riqky, M. A. (2021). Pengembangan Aplikasi Dan Website Manajemen Proyek Pt Santai Berkualitas Syberindo Menggunakan Metode Agile. *3rd Seminastika*, 8-15.
- Siallagan, T. F., & Wisnu, D. (2020). Rancang Bangun Sistem Pengidentifikasi Travel Bag Pada Kelompok Biro Perjalanan Umroh/Haji Berbasis Web. *Jurnal Teknologi Informasi Dan Komunikasi*, 15(1).
- Sidiq, M. A., Anshori, M. I., & Yaqin, R. A. (2024, Mei). Penerapan Arsitektur Monolitik Pada Aplikasi Jasa Service Online Tekku Berbasis Web. *Jurnal Komputer Dan Informatika*, 6(1), 27-36.
- Sie, J. B., Musdar, I. A., & Bahri, S. (2022, September). Pengujian White Box Testing Terhadap Website Room Menggunakan Teknik Basis Path. *Kharisma Tech*, 17(2), 45-57.
- Suhaimi, R., Santoso, N., & Siregar, R. A. (2020, Juni). Pengembangan Sistem Manajemen Proyek Menggunakan Metode Scrum Berbasis Android. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 4(6), 1897-1905.
- Suryantara, I. N., Raymond, R., & Sulaiman, D. C. (2021, November). Pengembangan Aplikasi Penjualan Mobil Dengan Framework Scrum Pada Pt Xyz. *Jurnal Teknik Sistem Dan Industri*, 2(2), 70-85.