
Perbandingan Metode *Agile* dan *Waterfall* Berdasarkan Analisis Waktu Pengembangan Sistem

Zulham Ghinafikar¹, Muhammad Muzakky Mu'thy², Muhammad Ainul Yaqin³

220605110184@student.uin-malang.ac.id¹, 220605110175@student.uin-malang.ac.id²,

yaqinov@ti.uin-malang.ac.id³

^{1,2,3}Universitas Islam Negeri Maulana Malik Ibrahim Malang

Informasi Artikel

Diterima : 17-02-2025

Direview : 07-03-2025

Disetujui : 25-04-2025

Kata Kunci

Perpustakaan, PHP,
MySQL, Sistem,
Informasi

Abstrak

Perpustakaan merupakan bagian dari sumber belajar yang harus dimiliki oleh setiap perguruan tinggi. Karena mahasiswa/i dengan mudah mencari informasi atau ilmu pengetahuan melalui perpustakaan. Dengan adanya perkembangan teknologi membuat manusia berfikir untuk dapat bekerja lebih efektif dan efisien. Salah satunya yaitu membuat sistem konvensional menjadi sistem yang terkomputerisasi. Dengan berbasis web, perpustakaan dapat lebih efektif dan efisien dalam pencarian dan pemesanan buku. Pada penelitian ini dirancang suatu sistem informasi berbasis web menggunakan bahasa pemrograman PHP MySQL dengan Metode Waterfall. Dengan sistem ini, diharapkan mampu mengatasi berbagai kebutuhan dari mahasiswa/i untuk mencari buku dan melakukan peminjaman serta memudahkan administrasi kampus dalam sirkulasi peminjaman buku dan pembuatan laporan. Dari hasil penelitian yang telah dilaksanakan penulis, program aplikasi ini dapat berjalan dengan baik, dan dapat menggantikan sistem lama yang masih di tuliskan ke dalam buku.

Keywords

Library, PHP, MySQL, System, Information

Abstrak

Libraries are part of the learning resources that every university must have. Because students can easily find information or knowledge through the library. With the development of technology, humans think to be able to work more effectively and efficiently. One of them is making a conventional system into a computerized system. With web-based, libraries can be more effective and efficient in searching and ordering books. In this study, a web-based information system was designed using the PHP MySQL programming language with the Waterfall Method. With this system, it is expected to be able to overcome the various needs of students to find books and make loans and facilitate campus administration in circulating book loans and making reports. From the results of research that has been carried out by the author, this application program can run well, and can replace the old system that is still written in the book.

A. Pendahuluan

Pemilihan metode pengembangan yang tepat menentukan keberhasilan proyek [1]. Dengan beragamnya metode yang tersedia, pengembang sering kali dihadapkan pada pertanyaan yang menentukan: "Metode mana yang paling efektif untuk mencapai tujuan pengembangan dengan waktu yang optimal?".

Penelitian ini bertujuan untuk menjawab pertanyaan tersebut dengan membandingkan dua metode pengembangan perangkat lunak yang dipilih: *Agile* dan *Waterfall*. Kedua metode ini memiliki pendekatan yang berbeda dalam mengelola siklus pengembangan, dengan *Agile* menekankan pada fleksibilitas, adaptabilitas dan iteratif, sementara *Waterfall* mengadopsi pendekatan yang lebih linear dan terstruktur [2].

Perbandingan efektivitas waktu pengembangan antara metode *Agile* dan *Waterfall* menjadi fokus utama penelitian ini. Untuk mencapai tujuan tersebut, metode pengukuran berbasis *Function Points* dipilih sebagai alat pengukuran yang tepat. *Function Points* adalah metode yang telah terbukti efektif dalam mengukur kompleksitas fungsional sebuah sistem informasi, yang kemudian dapat digunakan untuk memperkirakan waktu yang dibutuhkan dalam pengembangan.

Hasil penelitian ini diharapkan akan memberikan pemahaman yang lebih mendalam tentang perbedaan signifikan dalam waktu pengembangan antara metode *Agile* dan *Waterfall*. Dengan demikian, pengembang akan memiliki wawasan yang lebih baik dalam memilih metode yang paling sesuai dan efektif sesuai dengan kebutuhan proyek mereka.

B. Tinjauan Pustaka

B.1. Sistem Informasi

Dalam suatu sistem, beberapa elemen bekerja sama untuk mencapai suatu tujuan tertentu. Suatu sistem memiliki berbagai atribut seperti komponen, batasan, lingkungan eksternal, antarmuka, input, output, proses manajemen, dan tujuan. Ada dua pendekatan untuk menyoroti sistem dan alur kerja. Program, memproses data untuk menghasilkan informasi yang mendukung pengambilan keputusan. Informasi, merupakan hasil pengolahan data yang memberikan arti dan nilai bagi penerimanya. Informasi lebih dari sekedar kumpulan konsep; informasi mengontrol perilaku suatu sistem dan mengevaluasi kinerjanya [3].

Saat ini, sistem informasi mempunyai dampak yang signifikan terhadap berbagai organisasi, termasuk sektor swasta dan pemerintah. Oleh karena itu, penting untuk lebih mengembangkan proses bisnis, termasuk pengenalan sistem informasi [4].

B.2. Metode *Function Points Analysis*

Function Points adalah metode yang digunakan untuk mengukur ukuran fungsionalitas perangkat lunak dengan cara mengidentifikasi dan mengevaluasi berbagai elemen fungsional, seperti input, output, dan external query. Teknik ini, dikenal sebagai *Function Points Analysis (FPA)*, mengurai sistem menjadi komponen-komponen yang lebih kecil untuk memudahkan pemahaman dan analisis [5].

Penggunaan FPA memberikan beberapa manfaat, termasuk kemampuan untuk mengukur ukuran perangkat lunak, memahami biaya unit, serta membandingkan

produktivitas antara alat, bahasa, dan platform yang berbeda. Elemen-elemen fungsional yang dinilai dalam FPA meliputi transaksi eksternal seperti *External Inputs (EI)*, *External Output (EO)*, dan *External Inquiries (EQ)*, *Internal Logical Files (ILF)*, serta *External Interface Files (EIF)*.

B.3. Metode Agile

Metode *Agile* adalah pendekatan dalam pengembangan perangkat lunak yang menekankan pada perbaikan siklus, kolaborasi, dan adaptasi. Metode ini memungkinkan tim untuk merespons perubahan dengan cepat dan efisien, serta memprioritaskan kepuasan pelanggan. Metode *Agile* memiliki keunggulan terhadap durasi waktu yang cepat dalam pengembangannya sehingga menjadi lebih efisien dan fleksibel serta meningkatkan rasio kepuasan pelanggan [6].

Berikut ini merupakan tahapan-tahapan dalam pengembangan suatu sistem informasi menggunakan metodologi *Agile*:

- a. Analisis Kebutuhan: Pada tahap awal pengembangan sistem informasi, dilakukan identifikasi kebutuhan pengguna dan sistem. Ini melibatkan pemahaman terhadap keinginan pengguna terhadap sistem dan bagaimana sistem tersebut akan digunakan. Analisis ini mencakup kebutuhan sistem, fungsional, non-fungsional, dan aspek keamanan [7].
- b. Perancangan Database: Proses pembuatan struktur database yang akan digunakan dalam sistem informasi. Struktur ini dibuat berdasarkan kebutuhan sistem dan pengguna. Pendekatan yang sering digunakan adalah System Development Life Cycle (SDLC) [8].
- c. Pembuatan Interface Pengguna Dasar: Langkah ini melibatkan pengembangan interface pengguna yang dirancang untuk memudahkan pengguna dalam berinteraksi dengan sistem.
- d. Pengembangan Modul Inventori: Modul ini bertugas mengelola data inventori dalam sistem. Fungsi-fungsi seperti penambahan, penghapusan, dan pembaruan data inventori dimasukkan ke dalam modul ini [9].
- e. Integrasi dengan Database: Tahap ini mencakup menghubungkan sistem informasi dengan database.
- f. Pengembangan Modul Laporan: Modul ini dikembangkan untuk menghasilkan laporan berdasarkan data yang tersedia dalam sistem. Laporan tersebut digunakan untuk analisis dan pengambilan keputusan [9].
- g. Fitur Pencarian/Filter: Dirancang untuk memudahkan pengguna dalam mencari dan menyaring data dalam sistem.
- h. Pengujian: Tahap ini melibatkan pengujian sistem untuk memastikan bahwa sistem berfungsi dengan baik.
- i. Perbaikan Bug: Identifikasi dan perbaikan bug atau kesalahan dalam sistem dilakukan untuk memastikan bahwa sistem berfungsi tanpa masalah dan sesuai dengan kebutuhan [10].
- j. Optimalisasi Performa: Langkah ini bertujuan untuk meningkatkan kinerja sistem agar dapat beroperasi secara efisien dan efektif [11].
- k. Penerapan: Sistem diluncurkan ke lingkungan produksi agar dapat digunakan oleh pengguna [7].
- l. Dokumentasi: Pembuatan dokumentasi sistem yang menjelaskan cara kerja sistem dan cara penggunaan sistem [7].

- m. Pelatihan Pengguna: Pengguna diberikan pelatihan tentang cara menggunakan sistem untuk memastikan bahwa mereka dapat menggunakan sistem dengan baik dan efisien [7].

B.4. Metode *Waterfall*

Metodologi *Waterfall* menerapkan pendekatan linear dan berurutan, di mana setiap tahapan harus diselesaikan sebelum memulai tahapan berikutnya. Setiap tahapan menghasilkan dokumen dan artefak yang jelas, yang menjadi masukan untuk tahapan berikutnya. Pendekatan ini cocok untuk proyek dengan persyaratan yang stabil dan jelas, namun kurang fleksibel terhadap perubahan yang terjadi di tengah jalan [12].

Berikut ini merupakan tahapan-tahapan dalam pengembangan suatu sistem informasi menggunakan metodologi *Waterfall*:

- Requirement Analysis, menganalisis kebutuhan sistem secara menyeluruh. Ini melibatkan interaksi dengan pemangku kepentingan untuk memahami kebutuhan bisnis dan fungsional sistem secara detail [12].
- Design, perancangan ini mencakup rancangan database, desain antarmuka pengguna, serta perencanaan struktur program [12].
- Development, tim pengembangan mulai menerjemahkan desain menjadi kode program yang berfungsi. Ini melibatkan penulisan kode, pengujian unit, dan integrasi komponen-komponen sistem [12].
- Testing, sistem diuji secara menyeluruh untuk memastikan bahwa semua fungsi berjalan sesuai dengan spesifikasi yang ditetapkan. Pengujian meliputi pengujian fungsionalitas, pengujian integrasi, dan pengujian performa [12].
- Deployment, tahap implementasi dimulai. Ini melibatkan peluncuran sistem ke lingkungan produksi dan mengintegrasikannya dengan infrastruktur yang ada [12].
- Maintenance, perawatan dan pemeliharaan sistem menjadi fokus utama. Ini melibatkan pemantauan kinerja sistem, penanganan bug, serta peningkatan dan perbaikan sistem secara berkala [12].

C. Metode Penelitian

Pada bagian ini dijelaskan tahapan metode *Function Points Analysis* terdiri atas tahap Identifikasi EI, EO, EQ, EIF, ILF, tahap Hitung Jumlah dan Tipe Tiap Element, tahap Hitung *Function Points* Total dan tahap Perhitungan Tambahan.



Gambar 1. Metode *Function Points Analysis*

C.1. Menentukan DFD Sistem Informasi

Tahap menentukan DFD dilakukan dengan mengumpulkan beberapa DFD dari literatur pengembangan sistem informasi. DFD akan dipilih dan dilakukan FPA.

C.2. Identifikasi EI, EO, EQ, EIF, ILF

C.2.1. External Inputs

External Inputs (EI) dinilai dan diberi nilai. Penilaian ini didasarkan pada jumlah jenis elemen data (DET) dan jenis file yang dirujuk (FTR). DET mengacu pada jumlah jenis elemen data yang digunakan atau dimasukkan oleh EI [5]. Contohnya, dalam sebuah sistem informasi pemesanan online, DET mungkin mencakup data seperti nama pelanggan, alamat pengiriman, dan jumlah produk yang dipesan. FTR mengacu pada jumlah jenis file yang digunakan atau dirujuk oleh EI. Contohnya, dalam sebuah sistem informasi pemesanan online, DET mungkin mencakup data seperti nama pelanggan, alamat pengiriman, dan jumlah produk yang dipesan. FTR mengacu pada jumlah jenis file yang digunakan atau dirujuk oleh EI. Misalnya, dalam sistem yang sama, FTR mungkin mencakup file-file seperti database pelanggan, database produk, dan database pesanan.

Tabel 1. Penilaian EI

<i>Files Type Referenced (FTR)</i>	<i>Data Elements</i>		
	<i>1-4</i>	<i>5-15</i>	<i>Greater than 5</i>
<i>Less than 2</i>	<i>Low (3)</i>	<i>Low (3)</i>	<i>Average (4)</i>
<i>2</i>	<i>Low (3)</i>	<i>Average (4)</i>	<i>High (6)</i>
<i>Greater than 2</i>	<i>Average (4)</i>	<i>High (6)</i>	<i>High (6)</i>

Tabel 1.1 tersebut digunakan untuk menentukan tingkat dan nilai yang sesuai berdasarkan jumlah DET dan FTR yang terlibat dalam EI. Misalkan sebuah EI memiliki 6 DET dan merujuk (atau memperbarui) dua FTR. Dari tabel, karena jumlah DET-nya berada di antara 5-15 dan jumlah FTR-nya kurang dari 2, maka tingkatnya adalah "Low". Sesuai dengan tabel, nilai untuk tingkat "Low" adalah 3 unadjusted *Function Points*.

C.2.2. External Output

External Outputs (EO) dinilai dan diberi nilai. Penilaian ini didasarkan pada jumlah elemen data (DET) dan jenis file yang dirujuk (FTR). DET mengacu pada jumlah elemen data yang digunakan atau dihasilkan oleh EO [5].

Tabel 2. Penilaian EO

<i>Files Type Referenced (FTR)</i>	<i>Data Elements</i>		
	<i>1-5</i>	<i>6-19</i>	<i>Greater than 19</i>
<i>Less than 2</i>	<i>Low (4)</i>	<i>Low (4)</i>	<i>Average (5)</i>
<i>2 or 3</i>	<i>Low (4)</i>	<i>Average (5)</i>	<i>High (7)</i>
<i>Greater than 3</i>	<i>Average (5)</i>	<i>High (7)</i>	<i>High (7)</i>

Tabel 1.2 tersebut digunakan untuk menentukan tingkat dan nilai yang sesuai berdasarkan jumlah DET dan FTR yang terlibat dalam EO.

C.2.3. External Inquiries

Seperti komponen lainnya, External Inquiries (EQ) dinilai dan diberi nilai. Penilaian ini didasarkan pada jumlah elemen data (DET) dan jenis file yang dirujuk (FTR). DET mengacu pada jumlah elemen data yang digunakan dalam EQ [5].

Tabel 3. Penilaian EQ

<i>Files Type Referenced (FTR)</i>	<i>Data Elements</i>		
	<i>1-5</i>	<i>6-19</i>	<i>Greater than 19</i>
<i>Less than 2</i>	<i>Low (3)</i>	<i>Low (3)</i>	<i>Average (4)</i>
<i>2 or 3</i>	<i>Low (3)</i>	<i>Average (4)</i>	<i>High (6)</i>
<i>Greater than 3</i>	<i>Average (4)</i>	<i>High (6)</i>	<i>High (6)</i>

Tabel 1.3 tersebut digunakan untuk menentukan tingkat dan nilai yang sesuai berdasarkan jumlah DET dan FTR yang terlibat dalam EQ.

C.2.4. Internal Logical Files

Internal Logical Files (ILF) dinilai dan diberi nilai. Penilaian ini didasarkan pada jumlah elemen data (DET) dan jenis record (RET). DET mengacu pada jumlah elemen data yang disimpan atau dimanipulasi oleh ILF [5]. Misalnya, dalam sebuah sistem manajemen inventaris, DET mungkin mencakup data seperti nama produk, jumlah stok, dan harga. RET mengacu pada jumlah jenis record atau kelompok data yang disimpan dalam ILF.

Tabel 4. Penilaian ILF

<i>Record Element Types (RET)</i>	<i>Data Elements</i>		
	<i>1-19</i>	<i>20-50</i>	<i>Greater than 50</i>
<i>Less than 2</i>	<i>Low (7)</i>	<i>Low (7)</i>	<i>Average (10)</i>
<i>2-5</i>	<i>Low (7)</i>	<i>Average (10)</i>	<i>High (15)</i>
<i>Greater than 5</i>	<i>Average (10)</i>	<i>High (15)</i>	<i>High (15)</i>

Tabel 1.4 tersebut digunakan untuk menentukan tingkat dan nilai yang sesuai berdasarkan jumlah DET dan RET yang terlibat dalam ILF. Misalkan sebuah ILF memiliki 30 DET dan 3 RET. Dari tabel, karena jumlah DET-nya berada di antara 20-50 dan jumlah RET-nya di antara 2-5, maka tingkatnya adalah "Average". Sesuai dengan tabel, nilai untuk tingkat "Average" adalah 10.

C.2.5. External Interface Files

External Interface Files (EIF) dinilai dan diberi nilai. Penilaian ini didasarkan pada jumlah elemen data (DET) dan jenis record (RET). DET mengacu pada jumlah elemen data yang diakses atau digunakan oleh EIF[5]. Misalnya, dalam sebuah sistem manajemen inventaris, DET mungkin mencakup data seperti kode produk, harga, dan deskripsi. RET mengacu pada jumlah jenis record atau kelompok data yang disimpan dalam EIF.

Tabel 5. Penilaian EIF

<i>Record Element Types (RET)</i>	<i>Data Elements</i>		
	<i>1-19</i>	<i>20-50</i>	<i>Greater than 50</i>
<i>Less than 2</i>	<i>Low (5)</i>	<i>Low (5)</i>	<i>Average (7)</i>
<i>2-5</i>	<i>Low (5)</i>	<i>Average (7)</i>	<i>High (10)</i>
<i>Greater than 5</i>	<i>Average (7)</i>	<i>High (10)</i>	<i>High (10)</i>

Tabel 1.5 tersebut digunakan untuk menentukan tingkat dan nilai yang sesuai berdasarkan jumlah DET dan RET yang terlibat dalam EIF. Misalkan sebuah EIF memiliki 25 DET dan 4 RET. Dari tabel, karena jumlah DET-nya berada di antara 20-50 dan jumlah RET-nya di antara 2-5, maka tingkatnya adalah "Average". Sesuai dengan tabel, nilai untuk tingkat "Average" adalah 7.

C.3. Hitung Jumlah dan Tipe Tiap Element

C.3.1. Menghitung Unadjusted Function Points DFD

Memasukkan nilai dari EI, EO, EQ, EIF, ILF kedalam Tabel 2.

Tabel 6. Rumus Hitung UFP

<i>Type of Component</i>	<i>Complexity of Components</i>			<i>Total</i>
	<i>Low</i>	<i>Average</i>	<i>High</i>	
<i>External Inputs</i>	<i>× 3</i>	<i>× 4</i>	<i>× 6</i>	
<i>External Outputs</i>	<i>× 4</i>	<i>× 5</i>	<i>× 7</i>	

<i>External Inquiries</i>	× 3	× 4	× 6
<i>Internal Logical Files</i>	× 7	× 10	× 15
<i>External Interface Files</i>	× 5	× 7	× 10
Total Number of Unadjusted Function Points			

C.3.2. Hitung *Value Adjustment Factor*

Nilai dari setiap karakteristik diasumsikan dan dijumlahkan maka didapatkan TDI. VAF didapatkan dengan rumus:

$$VAF = 0.65 + (0.01 \times TDI)$$

C.4. Hitung *Function Points Total*

Rumus *Function Point*

$$FP = UFP \times VAF$$

C.5. Perhitungan Tambahan

Perhitungan tambahan ini akan mencakup perubahan *Function Points* menjadi sebuah *Schedule* dari metode *Agile* dan *Waterfall*. Untuk menyamakan kemampuan pengembang maka SDA(orang/anggota) akan disamakan antara pegebanan sistem dari 5 orang, 10 orang dan 15 orang.

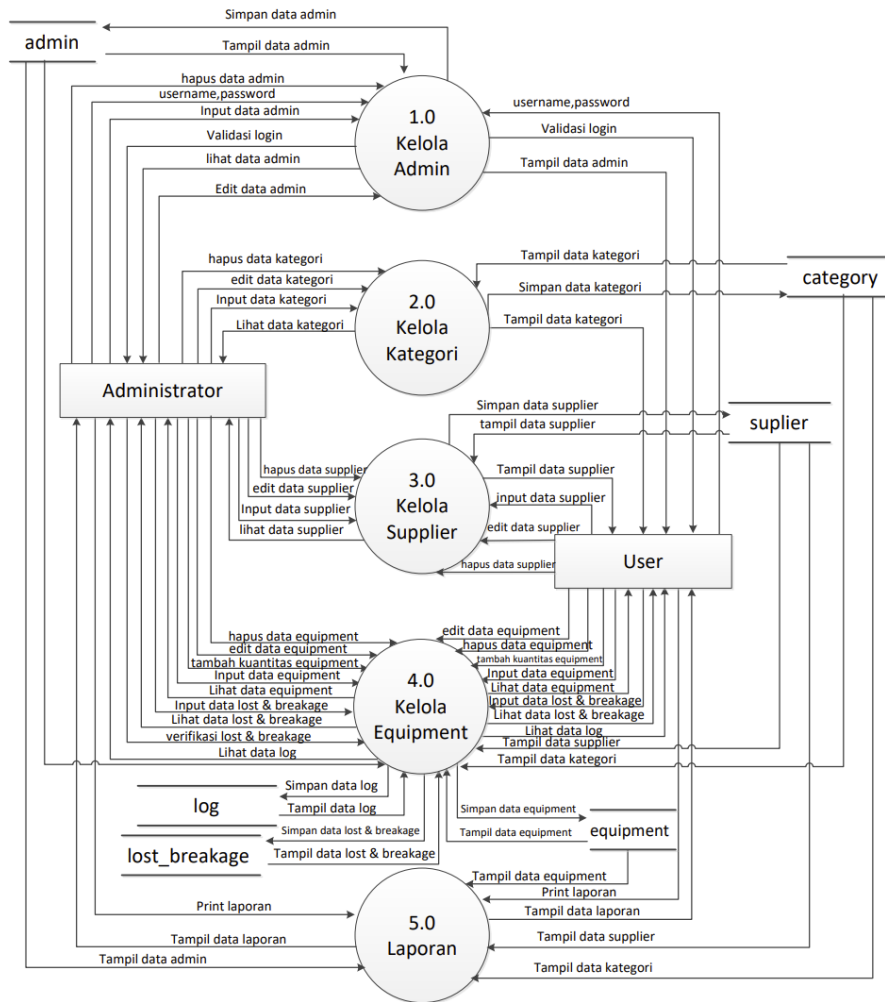
D. Hasil dan Pembahasan

D.1. Menentukan DFD Sistem Informasi

Berikut ini merupakan daftar Sistem Informasi yang kami gunakan:

- *Inventory* Aset Berbasis Web Pada Harris Hotel Denpasar [13]
- *Inventory* Data barang pada PT. Andalas Berlian Motors [14]
- Kasir Berbasis Web pada Toko XYZ [15]
- Penjualan Obat Berbasis Web [16]
- Administrasi Persuratan Berbasis Web pada Program Studi Teknik Electro [17]
- Penjadwalan Produksi pada Toko Perabot [18]
- Persediaan Barang [19]

DFD yang digunakan didapatkan dari literatur pengembangan sistem informasi.



Gambar 2. DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar”

D.2. Identifikasi EI, EO, EQ, EIF, ILF

Dari DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” didapatkan Unadjusted *Function Points* pada tabel 3.1 dan tabel 3.2:

Tabel 7. Komponen Data

<i>External Interface File (EIF)</i>	<i>Internal Logical File (ILF)</i>
None	Tabel admin
	Tabel <i>category</i>
	Tabel <i>supplier</i>
	Tabel <i>equipment</i>
	Tabel <i>log</i>
	Tabel <i>lost & breakage</i>

Tabel 8. Komponen Transaksional

<i>External Input (EI)</i>	<i>External Output (EO)</i>	<i>External Inquiry (EQ)</i>
Input data admin	Tampil data admin	Lihat data admin
Input data kategori	Tampil data kategori	Lihat data kategori
Input data supplier	Tampil data supplier	Lihat data supplier

Input data equipment	Tampil data equipment	Lihat data equipment
Input data lost & breakage	Tampil data lost & breakage	Lihat data lost & breakage
Verifikasi lost & breakage	Tampil data log	Lihat data log
	Tampil data laporan	
	Print laporan	

D.3. Hitung Jumlah dan Tipe Tiap Element

D.3.1. Hitung Unadjusted *Function Point*

Dari rumus Tabel 2, maka didapatkan UFP sebagai berikut:

Tabel 9. Hitung UFP dari DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar”

<i>Type of Component</i>	<i>Complexity of Components</i>			<i>Total</i>
	<i>Low</i>	<i>Average</i>	<i>High</i>	
<i>External Inputs</i>	6			18
<i>External Outputs</i>		8		40
<i>External Inquiries</i>	6			18
<i>Internal Logical Files</i>	6			42
<i>External Interface Files</i>				0
<i>Total Number of Unadjusted Function Points</i>				118

D.3.2. Hitung Value Adjustment Factor

Asumsikan nilai dari setiap karakteristik adalah sebagai berikut:

Tabel 10. Hitung VAF dari DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar”

No.	Karakteristik	Nilai
1	<i>Data communications</i>	4
2	<i>Distributed data processing</i>	2
3	<i>Performance</i>	3
4	<i>Heavily used configuration</i>	3
5	<i>Transaction rate</i>	3
6	<i>Online data entry</i>	4
7	<i>End-user efficiency</i>	4
8	<i>Online update</i>	4
9	<i>Complex processing</i>	3
10	<i>Reusability</i>	3
11	<i>Installation ease</i>	1
12	<i>Operational ease</i>	2
13	<i>Multiple sites</i>	1
14	<i>Facilitate change</i>	1
<i>Total Degree of Influence (TDI)</i>		38

$$VAF = 0.65 + (0.01 \times TDI)$$

$$VAF = 0.65 + (0.01 \times 38)$$

$$VAF = 0.65 + 0.38$$

$$VAF = 1.03$$

D.4. Hitung *Function Points* Total

Rumus *Function Point*

$$FP = UFP \times VAF$$

Maka untuk menghitung *Function Points* dari DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” adalah:

$$FP = UFP \times VAF$$

$$FP = 118 \times 1.03$$

$$FP = 121.54$$

Dari Cara di atas dan dengan nilai VAF yang sama didapatkan beberapa *Function Points* dari literatur pengembangan sistem informasi dengan seperti tabel 5 di bawah:

Tabel 11. *Function Points* dari beberapa literatur pengembangan sistem informasi

No.	Sistem Informasi	Nilai Point	Function
1	<i>Inventory</i> Data barang pada PT. Andalas Berlian Motors	123.6	
2	Kasir Berbasis Web pada Toko XYZ	94.76	
3	<i>Inventory</i> Aset Berbasis Web Pada Harris Hotel Denpasar	121.54	
4	Penjualan Obat Berbasis Web	108.15	
5	Administrasi Persuratan Berbasis Web pada Program Studi Teknik Electro	127.72	
6	Penjadwalan Produksi pada Toko Perabot	149.35	
7	Persediaan Barang	192.61	

D.5. Perhitungan Tambahan

Perhitungan Tambahan akan digunakan untuk menghitung waktu yang akan dihasilkan *Function Point*.

D.5.1. *Converting Function Points into a Schedule*

Dalam pembuatan jadwal dari *Function Points* ini, kami menggunakan AI bernama Claude untuk membuatkan jadwal dan lama pengerjaan dari setiap metode. Dari DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” didapatkan *Schedule* dari metode :

a. *Agile*

Dengan jumlah 5 pekerja didapatkan diagram gantt di bawah ini:

Tabel 12. *Schedule* metode *Agile* DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” dengan 5 pekerja

Sprint	M 1	M 2	M 3	M 4	M 5	M 6	M 7	M 8	M 9
	S S R K J	S S R K J	S S R K J	S S R K J	S S R K J	S S R K J	S S R K J	S S R K J	S S R K J
1	■								
2		■							
3			■						
4				■					
5					■				
6							■		

Dengan variasi jumlah pekerja didapatkan *Schedule* di bawah:

Tabel 13. *Schedule* metode *Waterfall* DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” dengan variasi 5, 10, dan 15 pekerja

Sprint	Aktivitas	Banyak Pekerja		
		5	10	15
1	Analisis kebutuhan, perancangan <i>database</i>	160 jam	80 jam	56 jam
2	Membangun antarmuka pengguna dasar, mengembangkan modul inventori	160 jam	88 jam	56 jam
3	Mengintegrasikan dengan <i>database</i> , mengembangkan modul laporan	160 jam	80 jam	56 jam
4	Fitur pencarian/filter, pengujian	160 jam	80 jam	52 jam
5	Perbaikan <i>bug</i> , optimalisasi performa	160 jam	80 jam	52 jam
6	<i>Deployment</i> , dokumentasi, pelatihan pengguna	176 jam	80 jam	56 jam

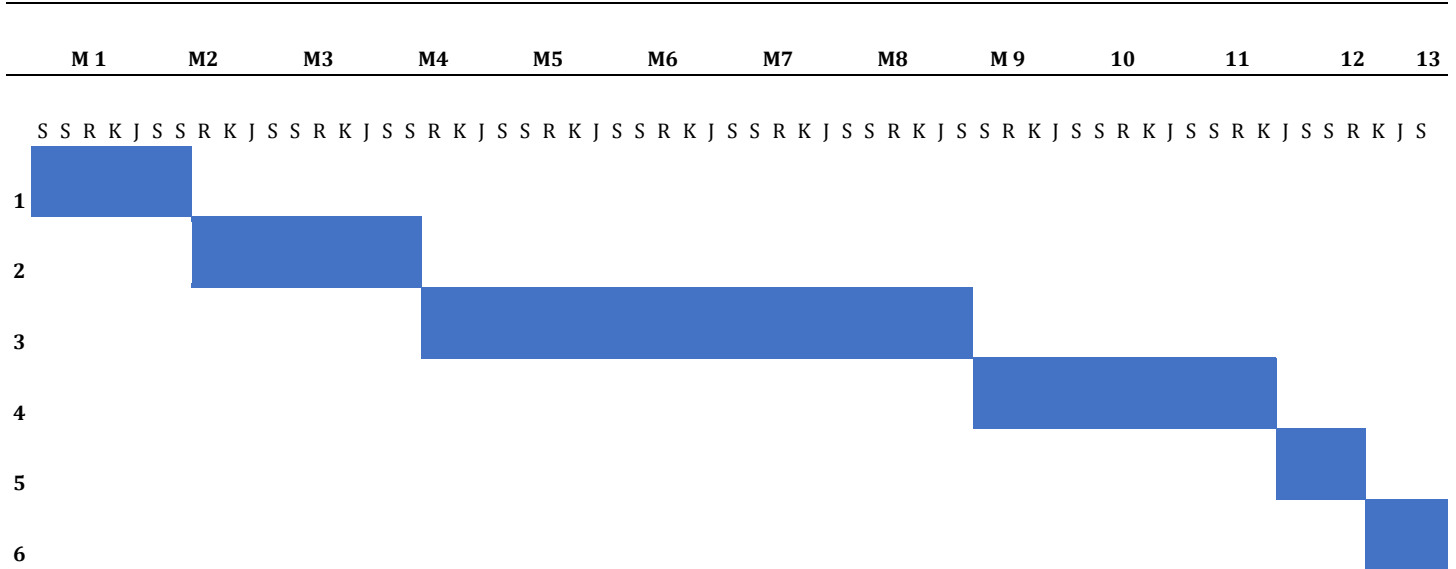
Catatan:

- Jadwal dibuat berdasarkan estimasi 976 jam untuk 5 orang, 488 jam untuk 10 orang, dan 328 jam untuk 15 orang pekerja.
- Setiap sprint berlangsung 2 minggu (80 jam) untuk 5-10 orang, atau 1 minggu (56 jam) untuk 15 orang pekerja.
- Pembagian tugas dan paralel pengerjaan dilakukan untuk memanfaatkan jumlah pekerja yang lebih banyak.

b. Waterfall

Dengan variasi jumlah pekerja didapatkan *Schedule* di bawah:

Tabel 14. *Schedule* metode *Waterfall* DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” dengan 5 pekerja



Dengan variasi jumlah pekerja didapatkan *Schedule* di bawah:

Tabel 15. *Schedule* metode *Waterfall* DFD Sistem Informasi “Inventori Aset Berbasis Web Pada Harris Hotel Denpasar” dengan variasi 5, 10, dan 15 pekerja

No.	Aktivitas	Banyak Pekerja		
		5	10	15
1	Analisis Kebutuhan	160 jam	80 jam	56 jam
2	Desain Sistem	240 jam	120 jam	80 jam
3	Pengkodean	560 jam	280 jam	192 jam
4	Pengujian	320 jam	160 jam	104 jam
5	<i>Deployment</i>	96 jam	48 jam	32 jam
6	Pemeliharaan	80 jam	40 jam	24 jam

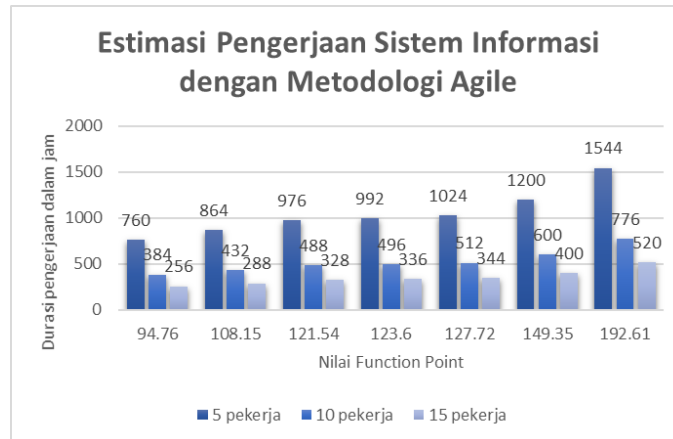
Catatan:

- Jadwal dibuat berdasarkan estimasi 1456 jam untuk 5 orang, 728 jam untuk 10 orang, dan 488 jam untuk 15 orang pekerja.
- Pembagian tugas dilakukan terutama pada tahap pengkodean agar memanfaatkan jumlah pekerja lebih banyak.
- Waktu pengerjaan berkurang dengan penambahan pekerja, tetapi tidak sebanyak pada metode *Agile*.

D.5.2. Converting *Schedule* dari semua DFD

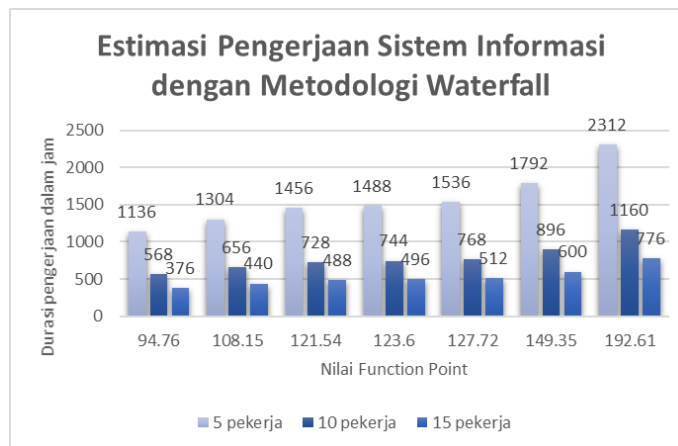
Dengan cara yang sama yaitu Converting *Function Points* into a *Schedule* di atas dilakukan converting juga dari DFD tabel 4. Maka Converting *Function Points* into a *Schedule* dari tabel 4 diperlihatkan dengan diagram:

a. Agile



Gambar 3. Diagram Estimasi waktu metode *Agile* yang didapatkan dari *Converting Function Points into a Schedule* dari tabel 5

b. Waterfall



Gambar 4. Diagram Estimasi waktu metode *Waterfall* yang didapatkan dari *Converting Function Points into a Schedule* dari tabel 5

D.5.3. Mengubah Sebagian Data Menjadi Persamaan Regresi Linear

Dengan menggunakan data waktu pengembangan dengan 5 pekerja, didapat perhitungan sebagai berikut:

a. Menghitung Mean

- Rata-rata X (*Function Point*)

$$\bar{x} = 94.76 + 108.15 + 121.54 + 123.6 + 127.72 + 149.35 + 192.61$$

$$\bar{x} \approx 131.104$$

- Rata-rata Y (Waktu Pengembangan)

$$\bar{y} = (760 + 864 + 976 + 992 + 1024 + 1200 + 1544)/7$$

$$\bar{y} \approx 1051.429$$

b. Menghitung Slope

- Jumlah hasil X dikurangi *Mean X*

$$\sum(x_i - \bar{x})(y_i - \bar{y}) =$$

$$\begin{aligned} & ((94.76 - 131.104)(760 - 1051.429)) + ((108.15 - 131.104)(864 - \\ & 1051.429)) + ((121.54 - 131.104)(976 - 1051.429)) + ((123.6 - \\ & 131.104)(992 - 1051.429)) + ((127.72 - 131.104)(1024 - 1051.429)) + \\ & ((149.35 - 131.104)(1200 - 1051.429)) + ((192.61 - 131.104)(1544 - \\ & 1051.429)) \end{aligned}$$

$$\sum(x_i - \bar{x})(y_i - \bar{y}) = 49161.017$$

- Jumlah hasil kuadrat dari X dikurangi *Mean X*

$$\begin{aligned} \sum(x_i - \bar{x})^2 &= (94.76 - 131.104)^2 + (108.15 - 131.104)^2 + (121.54 \\ & - 131.104)^2 + (123.6 - 131.104)^2 + (127.72 \\ & - 131.104)^2 + (149.35 - 131.104)^2 + (192.61 \\ & - 131.104)^2 \end{aligned}$$

$$\sum(x_i - \bar{x})^2 \approx 6122.91$$

- Rumus *slope a*

$$\begin{aligned} a &= \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2} \\ a &= \frac{49161.017}{6122.91} \approx 8.029 \end{aligned}$$

c. Menghitung *Intercept*

- Rumus *Intercept b*

$$b = \bar{y} - a \times \bar{x} =$$

$$b = 1051.429 - 8.029 \times 131.104 = -1.21168$$

d. Hasil Persamaan Regresi Linear

$$y_{a,agile,5} = a \times x + b$$

$$y_{a,agile,5} \approx 8.029 \times x - 1.212$$

D.5.4. Mengubah Semua Data Menjadi Persamaan Regresi Linear

Dengan cara yang sama yaitu Mengubah Sebagian Data Menjadi Persamaan Regresi Linear di atas dilakukan converting juga dari diagram pada gambar 3 dan gambar 4. Berikut ini merupakan hasil persamaan Regresi Linear dari data tersebut:

- Metode *Agile* dengan 5 pekerja

$$y_{a,agile,5} \approx 8.029 \times x - 1.212$$

- Metode *Agile* dengan 10 pekerja

$$y_{a,agile,10} \approx 4.031 \times x - 1.618$$

- Metode *Agile* dengan 15 pekerja

$$y_{a,agile,15} \approx 2.702 \times x - 1.0895$$

- Metode *Waterfall* dengan 5 pekerja

$$y_{waterfall,5} \approx 11.986 \times x + 3.471$$

- Metode *Waterfall* dengan 10 pekerja

$$y_{waterfall,10} \approx 6.018 \times x - 0.424$$

- Metode *Waterfall* dengan 15 pekerja

$$y_{waterfall,15} \approx 4.048 \times x - 3.911$$

D.6. Pembahasan

D.6.1. Perbandingan Waktu Pengembangan antara Metode *Agile* dan *Waterfall*

Hasil penelitian menunjukkan bahwa metode *Agile* secara konsisten memiliki waktu pengembangan yang lebih cepat dibandingkan dengan metode *Waterfall* untuk semua sistem informasi yang dianalisis dengan ukuran fungsional (*Function Point*) yang sama. Perbedaan waktu pengembangan ini cukup signifikan, misalnya untuk sistem informasi "Persediaan Barang" dengan nilai *Function Points* 192,61, jadwal pengembangan metode *Agile* dengan 15 pekerja adalah 1544 jam, sedangkan untuk *Waterfall* adalah 2312 jam - selisih waktu hampir 768 jam atau 33% lebih cepat dengan *Agile*.

Faktor utama yang menyebabkan metode *Agile* lebih cepat adalah pendekatan iteratif dan fleksibel yang dimilikinya. Dalam *Agile*, pengembangan dilakukan dalam siklus sprint yang lebih pendek, memungkinkan tim untuk menyesuaikan secara cepat terhadap perubahan kebutuhan dan memberikan umpan balik yang lebih sering dari pengguna. Selain itu, kolaborasi erat antara tim pengembang dan pengguna dalam *Agile* juga berkontribusi pada efisiensi dan kecepatan proses pengembangan.

D.6.2. Pengaruh Ukuran Fungsional terhadap Waktu Pengembangan

Ukuran fungsional yang diukur dengan metrik *Function Points* memberikan gambaran tingkat kompleksitas dari suatu sistem informasi. Semakin besar nilai *Function Point*, semakin kompleks sistem tersebut dan waktu pengembangan yang dibutuhkan juga semakin lama, baik untuk metode *Agile* maupun *Waterfall*. Namun, meskipun tren waktu pengembangan meningkat seiring dengan nilai *Function Points* yang lebih besar, kenaikan waktu tersebut lebih landai untuk *Agile* dibandingkan *Waterfall*.

Sebagai contoh, untuk sistem "Persediaan Barang" dengan *Function Points* 192,61, waktu *Agile* dengan 15 pekerja adalah 1544 jam, sedangkan untuk sistem "Penjualan Obat" dengan *Function Points* lebih rendah 108,15, waktunya hanya 872 jam - selisih 672 jam. Namun untuk *Waterfall*, selisih waktu antara kedua sistem tersebut lebih besar, yaitu 1008 jam. Ini menunjukkan bahwa *Agile* lebih efisien dalam menangani peningkatan kompleksitas sistem dibandingkan *Waterfall*.

D.6.3. Efek Jumlah Pekerja terhadap Waktu Pengembangan

Penelitian ini membandingkan efek jumlah pekerja yang berbeda-beda (5, 10, dan 15 orang) terhadap waktu pengembangan untuk masing-masing metode. Secara umum, baik untuk *Agile* maupun *Waterfall*, penambahan jumlah pekerja dapat mengurangi waktu pengembangan secara signifikan. Namun, penurunan waktu tersebut lebih terlihat pada metode *Agile* dibandingkan *Waterfall*.

Contohnya, untuk sistem "Administrasi Persuratan" dengan *Function Points* 127,72, peningkatan dari 5 pekerja ke 15 pekerja dapat mengurangi waktu sebesar

50% untuk *Agile* (dari 1024 jam menjadi 512 jam), namun hanya 33% untuk *Waterfall* (dari 1824 jam menjadi 1216 jam). Hal ini karena dalam *Agile*, pembagian tugas dan pengerjaan paralel dapat lebih optimal karena sifat iteratif dan kolaboratifnya, sehingga memanfaatkan penambahan pekerja dengan lebih baik.

D.6.4. Pemilihan Metode Pengembangan Berdasarkan Analisis Regresi Linear

Dari persamaan regresi linear yang dihasilkan, terlihat bahwa metode *Agile* dengan peningkatan jumlah pekerja cenderung memiliki slope yang lebih rendah, menunjukkan bahwa penambahan *Function Points* memerlukan waktu pengembangan yang lebih sedikit dibandingkan dengan metode *Waterfall*. Sebaliknya, metode *Waterfall* menunjukkan slope yang lebih tinggi, yang berarti bahwa untuk setiap penambahan *Function Points*, waktu pengembangan meningkat lebih tajam.

Hasil ini mengindikasikan bahwa metode *Agile* lebih efisien dalam hal penambahan *Function Points* seiring bertambahnya jumlah pekerja, sedangkan metode *Waterfall* menunjukkan peningkatan waktu pengembangan yang lebih signifikan untuk setiap penambahan *Function Points*. Analisis ini memberikan wawasan yang berharga bagi pengelolaan proyek dan pemilihan metode pengembangan yang tepat berdasarkan jumlah pekerja dan kompleksitas proyek.

D.6.5. Pemilihan Metode Pengembangan yang Tepat

Meskipun hasil penelitian menunjukkan bahwa *Agile* lebih efisien dalam hal waktu pengembangan, pemilihan metodologi pengembangan perangkat lunak yang tepat harus mempertimbangkan faktor-faktor lain selain waktu. Metode *Waterfall* mungkin lebih sesuai untuk proyek dengan persyaratan yang stabil dan terdefinisi dengan baik sejak awal, sehingga pendekatan linear dan terstruktur dalam *Waterfall* lebih menguntungkan.

Di sisi lain, *Agile* lebih cocok untuk proyek dengan persyaratan yang sering berubah atau belum sepenuhnya terdefinisi, di mana fleksibilitas dan kemampuan beradaptasi menjadi kunci. Faktor lain seperti ketersediaan sumber daya (jumlah pekerja), tingkat keterlibatan pengguna, dan prioritas proyek (waktu atau kualitas) juga harus dipertimbangkan dalam memilih metodologi yang paling sesuai.

D.6.6. Limitasi dan Penelitian Lebih Lanjut

Penelitian ini memiliki beberapa keterbatasan. Pertama, hanya membandingkan metode *Agile* dan *Waterfall*, sementara ada metodologi pengembangan lain yang juga penting untuk dieksplorasi, seperti Lean, Spiral, atau Rapid Application Development (RAD). Kedua, faktor-faktor lain yang dapat memengaruhi waktu pengembangan seperti pengalaman tim pengembang, kompleksitas teknis (di luar ukuran fungsional), dan lingkungan pengembangan tidak dipertimbangkan dalam penelitian ini.

Untuk penelitian lebih lanjut, perlu dilakukan perbandingan dengan metodologi pengembangan lainnya untuk memberikan wawasan yang lebih komprehensif. Selain itu, eksplorasi terhadap faktor-faktor lain yang dapat memengaruhi waktu pengembangan juga penting untuk mendapatkan pemahaman yang lebih lengkap tentang pemilihan metode yang paling efektif dalam pengembangan sistem informasi. Penelitian ini juga dapat diperluas dengan

menggunakan studi kasus atau proyek nyata untuk mendapatkan data yang lebih realistis dan akurat.

Setelah melalui tahap perancangan aplikasi yang dihasilkan memiliki kemungkinan tinggi untuk berhasil mengintegrasikan data antara sistem Odoo-17 dan interface pengguna yang dirancang. Proses pengembangan mengikuti pendekatan *Agile Development* di mana setiap iterasi menghasilkan komponen penting yang berkontribusi pada fungsionalitas aplikasi.

E. Simpulan

Penelitian ini membandingkan efektivitas waktu pengembangan antara metode *Agile* dan *Waterfall* dalam pengembangan sistem informasi dengan menggunakan metode Analisis *Function Point*. Hasil penelitian menunjukkan bahwa metode *Agile* secara konsisten memiliki waktu pengembangan yang lebih cepat dibandingkan dengan metode *Waterfall* untuk sistem informasi yang dianalisis dengan ukuran fungsional (*Function Point*) yang sama.

Keunggulan metode *Agile* dalam kecepatan pengembangan dikaitkan dengan pendekatan iteratif dan fleksibel yang dimilikinya, di mana pengembangan dilakukan dalam siklus yang lebih pendek, memungkinkan penyesuaian cepat terhadap perubahan kebutuhan, dan memberikan umpan balik yang lebih cepat dari pengguna. Kolaborasi erat antara tim pengembang dan pengguna juga berkontribusi pada efisiensi proses pengembangan dalam metode *Agile*.

Meskipun demikian, pemilihan metodologi pengembangan perangkat lunak yang tepat harus mempertimbangkan faktor-faktor lain seperti kompleksitas proyek, kebutuhan bisnis, dan ketersediaan sumber daya. Metode *Waterfall* mungkin lebih sesuai untuk proyek dengan persyaratan yang stabil dan terdefinisi dengan baik, sedangkan metode *Agile* dapat lebih menguntungkan untuk proyek dengan persyaratan yang sering berubah atau belum sepenuhnya terdefinisi.

Penelitian lebih lanjut diperlukan untuk mengeksplorasi faktor-faktor lain yang dapat memengaruhi waktu pengembangan serta melakukan perbandingan dengan metodologi pengembangan lainnya seperti *Lean*, *Spiral*, atau *Rapid Application Development (RAD)* untuk memberikan wawasan yang lebih komprehensif tentang pilihan metode yang tersedia bagi organisasi.

F. Ucapan Terima Kasih

Puji syukur penulis kami ucapkan kepada Allah SWT. atas karunia, rahmat dan hidayahnya yang telah memberikan kemampuan bagi penulis untuk menyusun dan menyelesaikan artikel. Tidak lupa juga pihak yang terlibat dalam penyusunan, khususnya pembimbing kami yang telah membimbing pengujian.

G. Referensi

- [1] E. Triandini, S. Jayanatha, A. Indrawan, G. Werla Putra, and B. Iswara, "Metode Systematic Literature Review untuk Identifikasi Platform dan Metode Pengembangan Sistem Informasi di Indonesia," *Indones. J. Inf. Syst.*, vol. 1, no. 2, p. 63, 2019, doi: 10.24002/ijis.v1i2.1916.
- [2] M. A. Yaqin and S. Zaman, *Rekayasa Perangkat Lunak: Kajian Teoretis dan Praktis*, no. February. 2024. [Online]. Available: https://www.researchgate.net/profile/Muhammad-Ainul-Yaqin/publication/378300314_Rekayasa_Perangkat_Lunak_Kajian_Teoretis_dan_Praktis/links/65d28af3e51f606f9979d580/Rekayasa-Perangkat-Lunak-Kajian-Teoretis-dan-Praktis.pdf

- [3] M. F. Abdurrohman, A. Momon, R. Fitriani, U. S. Karawang, P. Jaya, and T. Timur, "CENTRAL STUDIO PHOTOCOPY," vol. 12, no. 2, 2024.
- [4] E. Panja and D. Manongga, "Perancangan Sistem Informasi Keuangan Berbasis Web Pada Gks Mauliru Menggunakan Metode Rapid Application Development," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 579–584, 2023, doi: 10.36040/jati.v7i1.6401.
- [5] D. Longstreet, "Function Points Analysis Training Course," *Longstreet Consult. Inc. Accessed*, vol. 2, p. 15, 2005, [Online]. Available: http://www.poli.usp.br/d/pmr2490/fp_training.pdf
- [6] M. F. Iqbal and H. P. Putro, "Penerapan Simple Agile Methodology Dalam Pengembangan Aplikasi Web," *J. Autom.*, vol. 4, no. 1, pp. 1–6, 2023.
- [7] G. Allan, "Analisa Kebutuhan Kebutuhan Sistem Informasi Manajemen Perusahaan Dagang," *KURAWAL J. Teknol. Inf. dan Ind.*, vol. 4, no. 1, pp. 17–30, 2021.
- [8] L. A. Abdillah, "Perancangan basisdata sistem informasi penggajian," pp. 135–152, 2013, [Online]. Available: <http://arxiv.org/abs/1302.0337>
- [9] H. Mukti, "Sistem Informasi Inventory Barang Pada PT. Assami Ananda Berbasis Java Neatbeans," *Petir*, vol. 14, no. 2, pp. 139–149, 2021, doi: 10.33322/petir.v14i2.884.
- [10] M. A. Ridwan and S. Rochimah, "Peningkatan Akurasi Prediksi Waktu Perbaikan Bug dengan Pendekatan Partisi Data," *J. Sist. Inf. Bisnis*, vol. 8, no. 1, p. 76, 2018, doi: 10.21456/vol8iss1pp76-83.
- [11] F. Baharuddin and A. Tjahyanto, "Peningkatan Performa Klasifikasi Machine Learning Melalui Perbandingan Metode Machine Learning dan Peningkatan Dataset," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 11, no. 1, pp. 25–31, 2022, doi: 10.32736/sisfokom.v11i1.1337.
- [12] A. Abdul Wahid, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, pp. 1–5, 2020.
- [13] I. N. Rikajaya, M. Liandana, and P. P. Yudiastra, "Sistem Informasi Inventori Aset Berbasis Web Pada Harris Hotel Denpasar," *JTIM J. Teknol. Inf. dan Multimed.*, 2020, doi: 10.35746/jtim.v2i2.101.
- [14] O. Veza, "PERANCANGAN SISTEM INFORMASI INVENTORY DATA BARANG PADA PT.ANDALAS BERLIAN MOTORS (Studi Kasus: PT Andalas Berlian Motors Bukit Tinggi)," *J. Tek. Ibnu Sina*, 2017, doi: 10.36352/jt-ibsi.v2i2.63.
- [15] A. Kristianto, I. G. W. Sena, and R. Julianto, "Perancangan dan Pembuatan Sistem Informasi Kasir Berbasis Web pada Toko XYZ," *KONSTELASI Konvergensi Teknol. dan Sist. Inf.*, 2023, doi: 10.24002/konstelasi.v3i1.7131.
- [16] S. Subianto, "Sistem Informasi Berbasis Web untuk Penjualan Obat," *J. Ilm. Infokam*, 2021, doi: 10.53845/infokam.v17i2.301.
- [17] R. G. Franata, A. U. M.S, and R. Alfita, "Rancang Bangun Sistem Informasi Administrasi Persuratan Berbasis Web Pada Program Studi Teknik Elektro," 2020.
- [18] and R. A. A. Budiman, Alhamidi, E. Iswandy, "Perancangan Sistem Informasi Penjadwalan Produksi Pada Toko Perabot," *J. Sains dan Inform.*, 2022, doi: 10.22216/jsi.v8i1.977.
- [19] F. Damanik, R. Meilano, and T. wr, "Pengembangan Sistem Informasi

Persediaan Barang dengan Metode Waterfall," *J. Elektron. List. dan Teknol. Inf. Terap.*, 2021, doi: 10.37338/e.v2i2.153.