

Implementasi Algoritma *FP-Growth* pada Rancang Bangun PERPUSI Berbasis Website dengan Framework *Laravel*

Implementation of FP-Growth Algorithm in Website-Based PERPUSI Design with Laravel Framework

Muhammad Mas'ud ^{*1}, Salsabilla Fanny ², Novia Ratnasari³, Muhammad Amanulloh ⁴,
Zurriat Nyndia Rahmawati⁵

^{1,2,3,4,5} Program Studi Sistem Informasai, Fakultas Sains dan Teknologi, Universitas Islam
Raden Rahmat Malang

e-mail: *1muhammadmasud726@gmail.com

Abstrak - Studi kasus yang dipaparkan oleh penulis adalah membuat website perpustakaan dalam level yang sederhana dengan framework *Laravel*. Perpustakaan adalah salah satu institusi yang memiliki peran penting dalam mendukung proses pembelajaran dan penelitian. Namun, pengelolaan perpustakaan secara tradisional seringkali memakan waktu dan tidak efisien. Untuk mengatasi masalah tersebut maka penulis mencoba mengimplementasikan algoritma *FP-Growth* pada rancang bangun PERPUSI (Perpustakaan Sistem Informasi) berbasis website dengan menggunakan framework *Laravel*. Algoritma *FP-Growth*, yang menggunakan struktur pohon *FP-Tree* sebagai dasar, memiliki keunggulan dalam melakukan analisis pola asosiasi. Dalam konteks perpustakaan digital, algoritma ini dapat digunakan untuk mengidentifikasi pola peminjaman buku yang sering terjadi, sehingga dapat memberikan rekomendasi yang lebih personal dan sesuai dengan minat pengguna. Metode yang digunakan dalam merancang aplikasi ini dengan menggunakan metode *SDLC* (System Development Life Cycle) dimana tahapannya terdiri dari analisis kebutuhan, perencanaan, perancangan, implementasi dan pengujian. Hasil dari penelitian ini menunjukkan dari keempat kategori a. Status; b. Transisi; c. Kejadian; dan d. Aksi; yang telah diujikan menggunakan *blackbox testing* menghasilkan indikator bernilai sukses dan dapat diimplementasikan sebagai program pendukung pembelajaran di Program Studi Sistem Informasi Unira Malang. Untuk penelitian lanjutan dapat dikembangkan fitur-fitur dan maintenance program.

Kata kunci – Perpustakaan, Website, Framework *Laravel*, *SDLC*, *Blackbox*

Abstract - The case study presented by the author is to create a library website at a simple level with the *Laravel* framework. Libraries are one of the institutions that have an important role in supporting learning and research processes. However, traditional library management is often time consuming and inefficient. To overcome this problem, the authors try to implement the *FP-Growth* algorithm in the website-based PERPUSI (Information System Library) design using the *Laravel* framework. The *FP-Growth* algorithm, which uses the *FP-Tree* tree structure as a basis, has an advantage in analyzing association patterns. In the context of digital libraries, this algorithm can be used to identify patterns of borrowing books that often occur, so as to provide recommendations that are more personal and in accordance with user interests. The method used in designing this application uses the *SDLC* (System Development Life Cycle) method where the stages consist of needs analysis, planning, design,

implementation and testing. The results of this study show that of the four categories a. Status; b. Transition; c. Incident; and D. Action; which has been tested using blackbox testing produces indicators of success value and can be implemented as a learning support program in the Unira Malang Information Systems Study Program. For further research, features and program maintenance can be developed.

Keywords – Libraries, Websites, Laravel Frameworks, SDLC, Blackbox

I. PENDAHULUAN

Perpustakaan merupakan salah satu sarana yang sangat penting dalam menunjang proses belajar mahasiswa, melalui akses *internet* dapat lebih efisien dan mengurangi waktu [1]. Oleh karena itu, perpustakaan juga harus beradaptasi dengan perkembangan teknologi tersebut. Tujuan layanan perpustakaan adalah memberikan pemahaman dan informasi untuk meningkatkan pengetahuan dan keahlian perpustakaan seorang pembaca[2].

Berdasarkan wawancara yang dilakukan pada bulan Maret 2023 dengan staff TU perpustakaan Fakultas Saintek UNIRA Malang menyatakan bahwa dari kelima perpustakaan prodi masih belum ada yang menggunakan layanan perpustakaan berbasis teknologi informasi dan komunikasi. Perpustakaan jurusan Sistem Informasi merupakan salah satu perpustakaan yang masih belum memiliki sistem klasifikasi dan layanan buku. Berdasarkan data perpustakaan yang ada di Program Studi Sistem Informasi Universitas Islam Raden Rahmat Malang, kurang lebih ada 53 buku yang terdapat diperpustakaan tersebut. Banyaknya jenis buku yang terdapat diperpustakaan, pencatatan transaksi peminjaman dan pengembalian buku, pencatatan laporan masih dilakukan secara manual. Dari hasil wawancara kami ditemukan beberapa permasalahan, yaitu pihak pengelola perpustakaan tidak mengetahui semua judul buku yang tersedia di perpustakaan. Situasi ini menghadirkan tantangan bagi pengelola perpustakaan ketika pengunjung menanyakan lokasi buku yang sedang mereka cari[3]. Tata letak buku yang tidak beraturan dikarenakan koleksi buku belum mempunyai kategori menyulitkan pengunjung perpustakaan dalam mencari buku yang dibutuhkan [3]. Ketiadaan sistem klasifikasi dan pencarian buku memaksa pengunjung perpustakaan untuk melakukan pencarian buku secara manual.

Tahapan penelitian ini diawali dengan menggunakan algoritma Frequent Pattern Growth. Algoritma ini dapat diimplementasikan untuk mengidentifikasi himpunan data yang sering muncul (frequent itemset) dalam suatu kumpulan data [4]. Dalam konteks perpustakaan, algoritma FP-Growth dapat digunakan untuk mengidentifikasi keterkaitan antara buku-buku yang sering dipinjam, sehingga membantu petugas perpustakaan dalam mengelola koleksi buku.

Framework Laravel dipilih sebagai basis pengembangan sistem informasi perpustakaan ini karena memiliki banyak kelebihan seperti mempermudah pengembangan aplikasi *web* dengan konsep *Model-View-Controller (MVC)*, menyediakan fitur-fitur yang lengkap untuk kebutuhan pengembangan aplikasi. *Laravel* memiliki fokus yang terutama ditujukan untuk pengguna akhir (*end-user*), yang berarti menekankan pada kejelasan dan kesederhanaan baik dalam penulisan maupun tampilan [5].

Dengan dikembangkannya sistem informasi perpustakaan berbasis *website* ini, diharapkan dapat menyediakan sumber belajar yang diwadahi oleh sebuah aplikasi perpustakaan yang dapat diakses oleh seluruh civitas program studi di Universitas Islam Raden Rahmat Malang dalam meningkatkan kualitas pelayanan perpustakaan serta mempermudah pengguna dalam mendapatkan informasi dan melakukan peminjaman buku. Penelitian ini dilaksanakan dan diperuntukan pada civitas program studi.

Hasil dari pengujian pada penelitian ini akan digunakan sebagai masukan pada proses

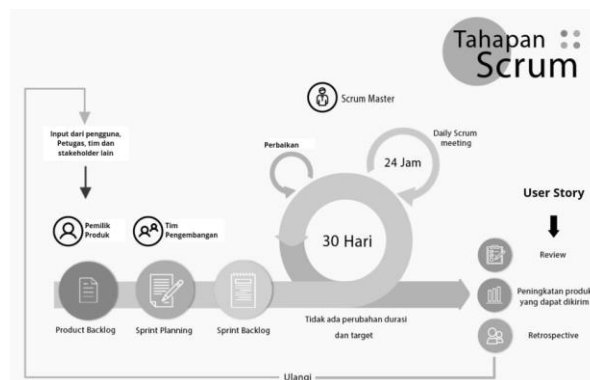
maintenance aplikasi perpustakaan di program studi Sistem Informasi Unira Malang. Dalam konteks pendidikan, hasil penelitian juga dapat dijadikan referensi untuk mengembangkan program yang menggunakan bahasa pemrograman *PHP* guna mendukung proses belajar mengajar.

II. METODE

Pada penelitian ini terdapat beberapa tahapan yaitu, pengumpulan data, analisis sistem, pengembangan sistem dan pengujian sistem. Pada tahap pengumpulan data, penulis melakukan observasi langsung di perpustakaan prodi Sistem Informasi Unira Malang untuk menganalisis kebutuhan sistem yang akan digunakan dalam aplikasi. Selain itu, penulis juga melakukan wawancara dengan staff tata usaha, dan melakukan studi literatur dari berbagai jurnal atau penelitian sebelumnya sebagai dasar dalam pengembangan aplikasi berbasis *web*. Kemudian, penulis melakukan analisis dan perancangan sistem menggunakan diagram *use case* dan diagram aktivitas. Pada pembuatan situs *web* ini dilakukan dengan metode penelitian *Agile* menggunakan pendekatan *scrum*, dengan mengikuti tahapan yang ada hingga mencapai hasil berupa aplikasi sistem informasi perpustakaan.

2.1 Metode Agile Scrum

Metode yang digunakan adalah *Agile* dengan model *Scrum*. Berikut adalah penjelasan mengenai tahapan *Agile Scrum*:



Gambar 1. Metode pengembangan sistem *Agile scrum*

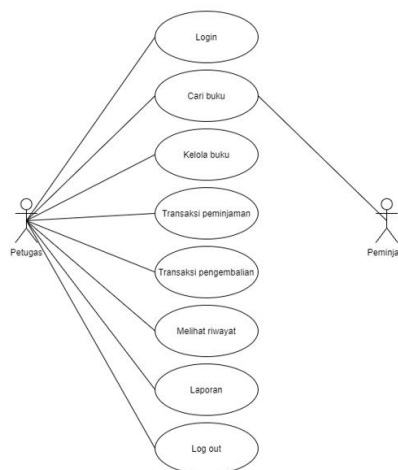
1. Pada *User Story* terdapat informasi mengenai nama pengguna sistem dan fitur-fitur yang ada dalam *website*, seperti penambahan data pengguna, data buku, dan melihat lokasi buku. Tahap perencanaan didasarkan pada hasil wawancara yang telah dilakukan, dan hasilnya menunjukkan bahwa sistem memiliki dua aktor, yaitu Petugas dan pengunjung [6]. Dari kedua aktor tersebut, sistem memiliki total delapan kebutuhan fungsional yang telah diverifikasi oleh pihak perpustakaan. Rincian kebutuhan fungsional tersebut dijelaskan dalam Tabel 1.

Tabel 1. Kebutuhan Fungsional

No	Kebutuhan Fungsional Deskripsi dan Spesifikasi
1.	Peminjam dapat mencari buku dengan kriteria pencarian. Sistem akan menampilkan buku yang telah dicari oleh peminjam.
2.	Petugas dapat melakukan login pada sistem. Sistem menampilkan form login.

3. Petugas dapat mencari buku dengan kriteria pencarian. Sistem akan menampilkan buku yang telah dicari oleh Petugas.
 4. Petugas dapat melakukan pengelolaan data buku dengan melakukan login terlebih dahulu. Sistem akan menampilkan pilihan yang akan dipilih oleh Petugas untuk melakukan pengelolaan buku.
 5. Petugas dapat melakukan transaksi peminjaman buku yang akan dipinjam oleh Peminjam. Sistem akan menampilkan form peminjaman buku.
 6. Petugas dapat melakukan validasi pengembalian buku dengan menggunakan kode peminjaman.
 7. Petugas dapat melihat daftar riwayat peminjaman buku dan mencari riwayat peminjaman buku berdasarkan tanggal serta mencetak laporan.
 8. Petugas dapat melakukan logout dari sistem.
-

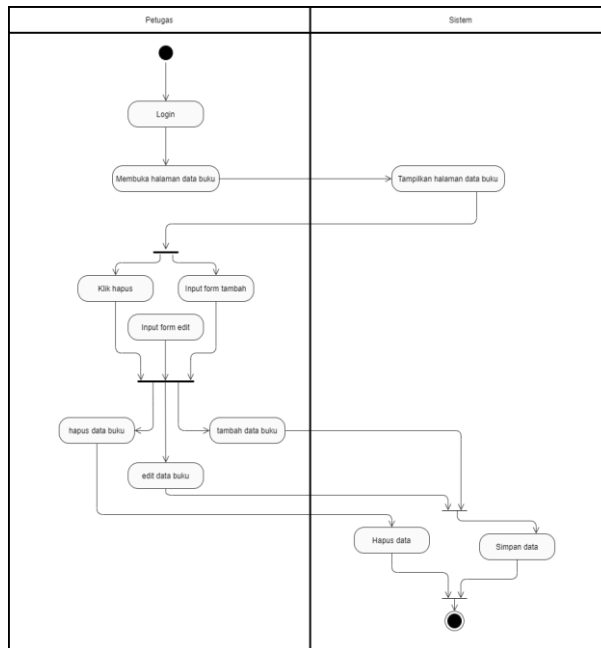
Berdasarkan hasil analisis kebutuhan yang telah dilakukan, peneliti dapat melakukan pemodelan diagram *use case*. Diagram *use case* tersebut dapat jelaskan dalam Gambar 2.



Gambar 2. Use case diagram

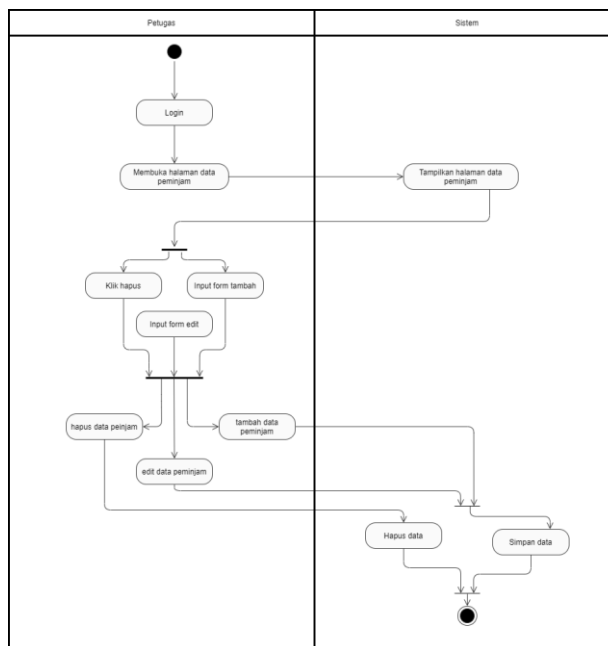
2. *Product Backlog* akan menghasilkan elemen-elemen seperti aktor dan aktivitas sistem. Konten dari *Product Backlog* didapatkan dari cerita pengguna (*User Story*). Dalam penelitian ini, peneliti menggunakan *UML* sebagai dasar analisis yang digunakan dalam pengembangan program PERPUSI. Tahap desain perancangan melibatkan pemodelan *activity* diagram dan *sequence* diagram.

Activity diagram dimodelkan untuk mengetahui alur proses dari sebuah fitur didalam sistem, sekaligus mengetahui aktor-aktor yang berperan didalamnya. Diagram ini memiliki kepentingan utama dalam memodelkan fungsi-fungsi sistem dan menekankan pada aliran kendali antara objek-objek [7]. Diagram aktivitas yang ada di Gambar 3 menggambarkan urutan langkah-langkah yang dilakukan oleh petugas untuk mengelola data buku.



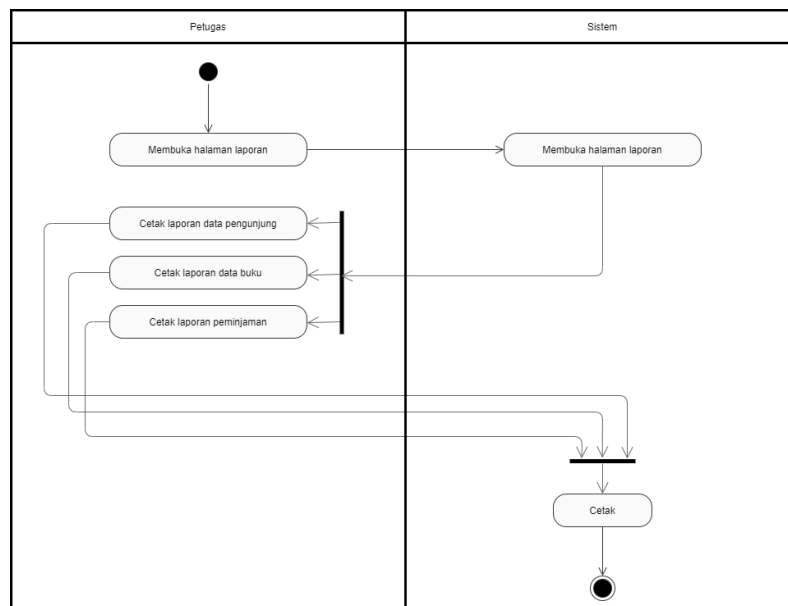
Gambar 3. Activity diagram halaman petugas mengelola data buku

Pada Gambar 4 menjelaskan alur proses petugas untuk melakukan pengelolaan data peminjam.



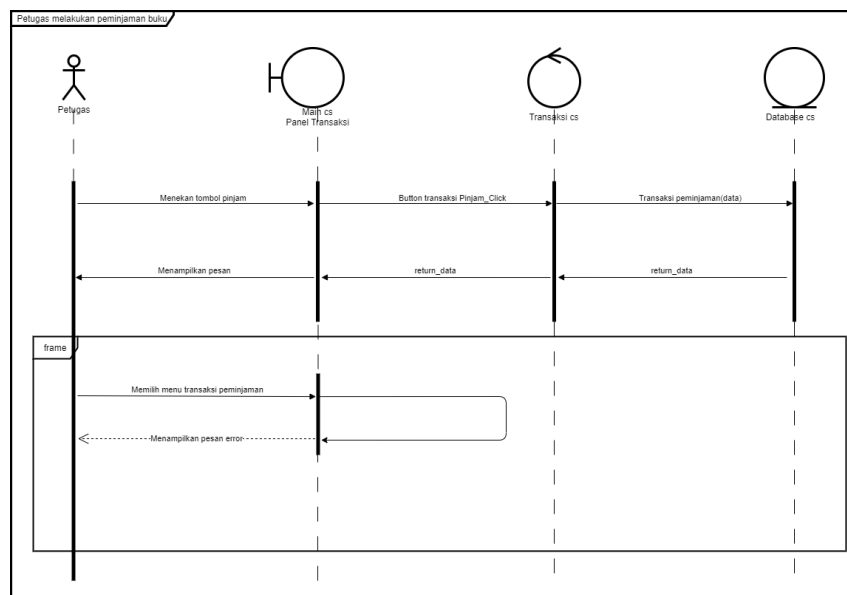
Gambar 4. Activity Diagram Halaman Admin mengelola Data Peminjam

Diagram yang terdapat pada Gambar 5 menggambarkan langkah-langkah yang dilakukan oleh petugas untuk melakukan pengelolaan data laporan. Petugas memulai proses dengan menekan menu "Laporan" yang tersedia di menu bar. Di dalam menu ini terdapat pilihan untuk mencetak laporan data pengunjung, laporan data buku, dan laporan peminjaman. Setelah memilih jenis laporan yang diinginkan, sistem akan menampilkan halaman cetak laporan, di mana petugas dapat melakukan proses pencetakan laporan.



Gambar 5 Activity diagram petugas mengelola data laporan

Setelah melakukan pemodelan diagram aktivitas, langkah berikutnya adalah melakukan pemodelan *sequence* diagram. Dalam penelitian ini, fitur-fitur yang akan digambarkan dalam *sequence* diagram hanya mencakup fitur-fitur penting, seperti pengelolaan buku, transaksi peminjaman dan pengembalian buku, serta pembuatan laporan. Interaksi yang terjadi antara class-class tersebut melibatkan operasi-operasi tertentu, di mana urutan antar operasi menjadi penting dan informasi yang diperlukan oleh setiap operasi harus diperhatikan[8]. Diagram urutan pada Gambar 6 menjelaskan alur proses petugas dalam melakukan transaksi peminjaman buku.



Gambar 6. Sequence Diagram Peminjaman Buku

3. Dalam *Sprint planing*, penulis akan membuat sebuah *website* dengan batas waktu maksimal satu bulan. Setiap fitur akan ditentukan waktu pengerjaannya [9]. Pada tahap *sprint planning*, penulis akan mengevaluasi daftar fitur yang ada dalam *product backlog*. Durasi *sprint* tidak boleh terlalu panjang, dengan batas maksimal 30 hari.

Tabel 1. *Sprint Planning*

Aktor	<i>Sprint Planning</i>	Durasi (Waktu/Hari)
Administrator	Login admin	1 Hari
	<i>Dashboard</i> admin	
	Data pengguna	1 Hari
	Data buku	
	Lokasi buku	
	Transaksi pinjam	1 Hari
	Transaksi kembali	
	Buku hilang	
	Laporan	
	Pengunjung	Data buku
	Login user	

4. *Sprint backlog*, penulis akan mengerjakan dan mengumpulkan daftar fitur yang telah dibuat pada produk *backlog* dan *sprint planning*.
pada *sprint backlog*, daftar fitur yang ada pada *Product backlog* dan *sprint planning* dikumpulkan, lalu dikerjakan, dapat dilihat pada Table 3. Pada pengerjaan fitur-fitur admin memerlukan waktu yang lama.

Tabel 2. *Sprint Backlog* pengelolaan buku

No	<i>Sprint 1</i>	Tugas	Durasi(Waktu/Jam)																	
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	Halaman pengelolaan buku	Tampilan <i>front-end</i>	■																	
		Membuat database					■													
		<i>Coding</i>									■									
		<i>Testing</i>													■					
		<i>Deployment</i>																	■	

5. *Daily Scrum* pada tahap ini bertujuan untuk menyinkronkan pekerjaan yang telah dilakukan oleh para pengembang. Tim akan berdiskusi mengenai perkembangan dari *sprint* yang telah selesai maupun yang sedang dikerjakan [10].

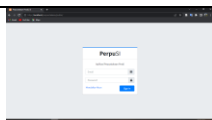
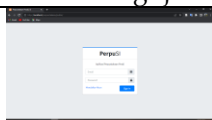
2.2 Indikator *Blackbox Testing*

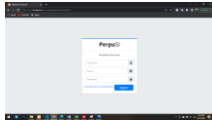
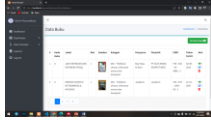
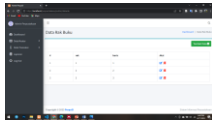
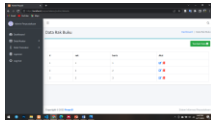
Penelitian ini menggunakan metode pengujian *Blackbox Testing* yang melibatkan seorang ahli materi dan seorang ahli media. Metode yang digunakan dalam penelitian ini adalah *State Transition Table* untuk memeriksa validitas transisi status program. *State Transition Table* memiliki beberapa indikator, termasuk:

- 1) Status yang ada dalam sistem.
- 2) Transisi antara status-status tersebut.
- 3) Kejadian yang menyebabkan transisi tersebut.
- 4) Aksi yang dihasilkan.

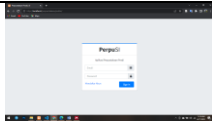
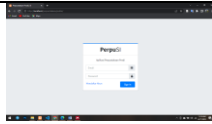
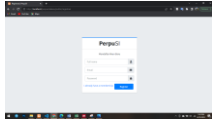
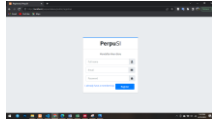


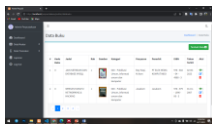
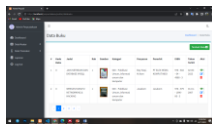
2.3 Hasil Pengujian *Blackbox Testing*

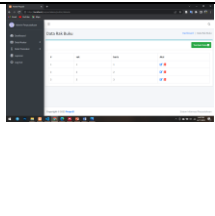
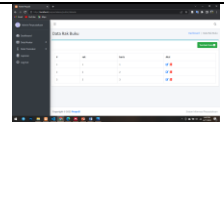
Tabel 3. Pengujian *Blackbox Testing* ahli materi

No.	Input	Test Case	Output yang diharapkan	Hasil Pengujian	Status
1.	Login menggunakan validasi data		Berhasil menampilkan halaman login		Sukses

2.	yang sesuai Registrasi menggunakan hak akses yang sesuai		Setiap pengguna yang akan melakukan registrasi berhasil.		Sukses
3.	Masuk ke halaman utama		Pengguna akan dapat menjelajahi halaman utama, menggunakan menu dan tombol yang tersedia, dan mengakses semua fungsi yang relevan dalam aplikasi.		Sukses
4.	Melakukan pengelolaan data buku		Pengguna akan dapat menambah, mengupdate, atau menghapus data buku sesuai dengan kebutuhan.		Sukses
5.	Melakukan pengelolaan data rak buku		Pengguna akan dapat menambah, mengupdate, atau menghapus data rak buku sesuai dengan kebutuhan		Sukses

Tabel 4. Pengujian *Blackbox Testing* ahli materi

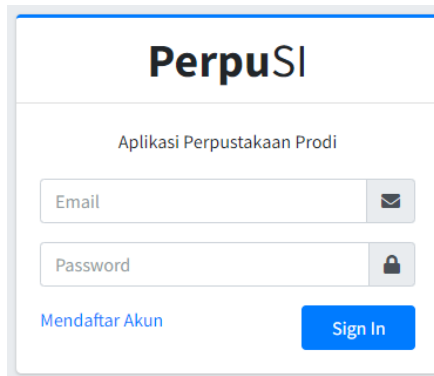
No.	Input	Test Case	Output yang diharapkan	Hasil Pengujian	Status
1.	Login menggunakan validasi data yang sesuai		Sistem akan memeriksa dan memvalidasi data yang dimasukkan oleh pengguna saat login		Sukses
2.	Registrasi menggunakan hak akses yang sesuai		Sistem dapat melakukan registrasi dan memberikan hak akses yang sesuai dengan perannya, seperti pengunjung, dan administrator.		Sukses
3.	Masuk ke halaman utama		Sistem dapat menjelajahi halaman utama, menggunakan menu dan tombol yang tersedia, dan mengakses semua fungsi yang relevan dalam aplikasi.		Sukses
4.	Melakukan pengelolaan data buku		Sistem dapat menambah, mengupdate, atau menghapus data buku sesuai dengan kebutuhan.		Sukses

5. Melakukan pengelolaan data rak buku		Sistem dapat menambah, mengupdate, atau menghapus data rak buku sesuai dengan kebutuhan.		Sukses
--	---	--	---	--------

Dari tabel 4 dan 5 dapat disimpulkan bahwa hasil pengujian yang melibatkan ahli materi dan ahli media menggunakan metode pengujian *blackbox*, kedua ahli menyatakan bahwa hasilnya sukses. Dari sini dapat disimpulkan bahwa program aplikasi telah memenuhi standar dan tidak ada kesalahan atau *error*. Ketiadaan *error* menunjukkan bahwa aplikasi dapat diakses dengan baik dan berfungsi sebagaimana mestinya.

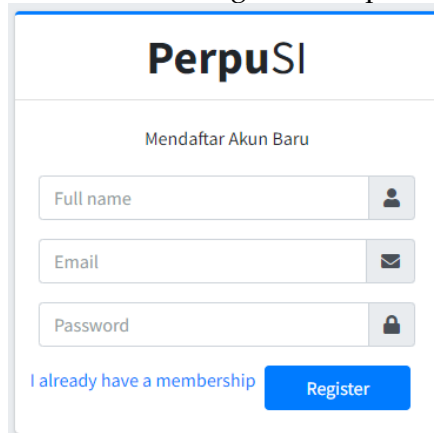
III. HASIL DAN PEMBAHASAN

Hasil penelitian ini dapat menjadi program aplikasi yang dapat digunakan pada program studi Sistem Informasi Unira Malang. Pertama kali pengguna membuka sistem, mereka akan disajikan dengan halaman login yang ditampilkan dalam gambar 8. Setelah berhasil login, mereka akan diarahkan ke halaman *dashboard*.



Gambar 7. Tampilan login

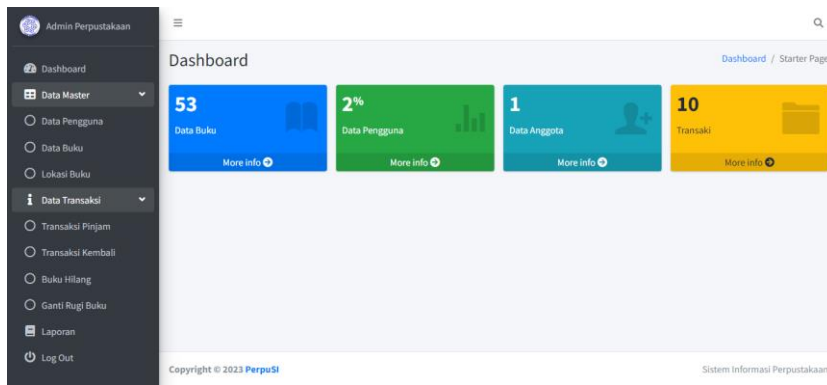
Halaman registrasi pada aplikasi perpustakaan bertujuan untuk memfasilitasi pengguna dalam melakukan pendaftaran sebagai pengguna aplikasi. Pada halaman ini, pengguna dapat mengisi formulir registrasi dengan data pribadi seperti nama, alamat *email*, kata sandi. Halaman registrasi memainkan peran penting dalam menyediakan akses yang sesuai dan memastikan bahwa hanya pengguna terdaftar yang dapat memanfaatkan fitur-fitur perpustakaan secara penuh. Halaman registrasi dapat ditunjukkan pada gambar 9.



Gambar 8. Halaman registrasi

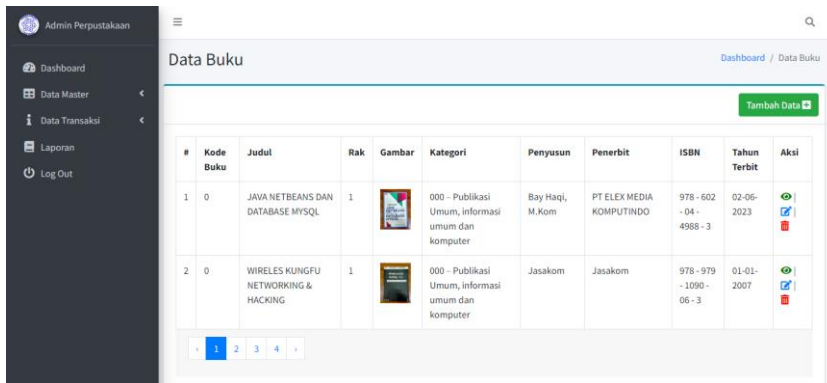
Halaman *dashboard* merupakan halaman yang mengelola semua akses terkait data master. Seperti data buku, data pengguna, data peminjaman, data pengembalian, dan laporan yang

ditunjukkan pada gambar 10. Halaman *dashboard* memiliki peran yang sangat penting dalam pembuatan aplikasi.



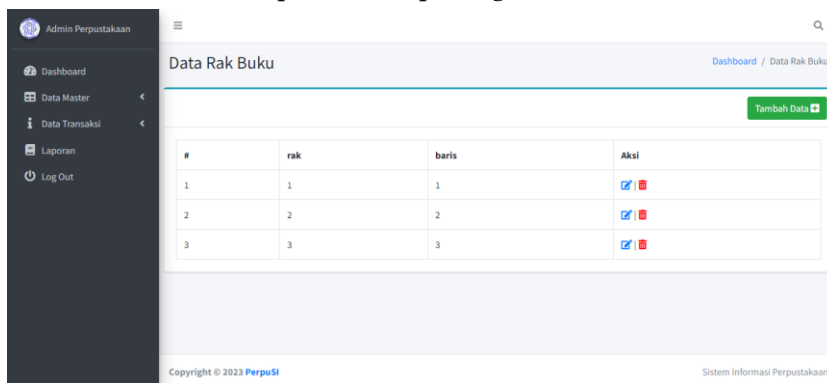
Gambar 9. Tampilan *dashboard* petugas

Pada halaman Data Buku, petugas memiliki akses untuk mengelola semua data buku. Petugas dapat melakukan penambahan, pembaruan, dan penghapusan data buku sesuai dengan kebutuhan. Halaman kelola buku dapat dilihat pada gambar 11.



Gambar 10. Tampilan kelola data buku

Pada halaman menu rak buku petugas memiliki akses untuk mengelola semua data rak buku memberikan fungsionalitas dasar yang melibatkan operasi *CRUD* (*Create, Read, Update, Delete*) untuk mengelola rak buku dalam sistem perpustakaan berbasis *website*. Tampilan kelola data rak buku dapat dilihat pada gambar 12.



Gambar 11. Tampilan kelola data rak buku

IV. KESIMPULAN

Berdasarkan implementasi *Algoritma FP-Growth* pada rancang bangun PERPUSI berbasis *website* dengan *framework laravel*, dapat disimpulkan bahwa penggunaan algoritma ini secara efektif memungkinkan sistem perpustakaan untuk mengidentifikasi pola-pola frekuensi dalam data transaksi. Hal ini memungkinkan pengguna untuk melakukan analisis yang lebih efisien terhadap data perpustakaan, termasuk penentuan asosiasi antara buku dan preferensi pengguna. Implementasi ini juga menunjukkan bahwa penggunaan *framework laravel* dalam pengembangan *website* perpustakaan memberikan keuntungan dalam hal kejelasan dan kesederhanaan dalam penulisan dan tampilan kode, serta menyediakan fungsionalitas aplikasi *web* yang dapat berfungsi dengan baik. Dengan demikian, implementasi *Algoritma FP-Growth* pada PERPUSI berbasis *website* dengan *framework laravel* memiliki potensi untuk meningkatkan pengalaman pengguna dan efisiensi dalam pengelolaan perpustakaan.

DAFTAR PUSTAKA

- [1] D. Abdullah, *Merancang Aplikasi Perpustakaan Menggunakan SDLC*, no. October. 2020.
- [2] U. Bengkulu, "Upaya meningkatkan layanan pemustaka di upt perpustakaan universitas bengkulu," vol. 3, no. 2, pp. 203–214, 2018.
- [3] V. Rizqiyani, A. Mulwinda, and R. D. M. Putri, "Klasifikasi Judul Buku dengan Algoritma Naive Bayes dan Pencarian Buku pada Perpustakaan Jurusan Teknik Elektro," *J. Tek. Elektro*, vol. 9, no. 2, pp. 60–65, 2017, [Online]. Available: <https://journal.unnes.ac.id/nju/index.php/jte/article/view/11658/7332>.
- [4] G. Gunadi and D. I. Sensuse, "Penerapan Metode Data Mining Market Basket Analysis Terhadap Data Penjualan Produk Buku Dengan Menggunakan Algoritma Apriori Dan Frequent Pattern Growth (Fp-Growth) :," *Telematika*, vol. 4, no. 1, pp. 118–132, 2012.
- [5] R. S. Zahra, O. V. Putra, and R. U. Putri, "Rancang Bangun Sistem Informasi Telemedicine Pada Darussalam Medical Center (DMC) Menggunakan Framework Laravel," no. Dmc, pp. 84–91.
- [6] M. A. Dewi and R. Irham, "Penerapan Agile Scrum Pada Pengembangan Bimbingan Daring Skripsi Mahasiswa," *J. SISKOM-KB (Sistem Komput. dan Kecerdasan Buatan)*, vol. 4, no. 2, pp. 40–45, 2021, doi: 10.47970/siskom-kb.v4i2.195.
- [7] L. Sakuroh, M. R. Julianti, and S. R. Siregar, "Perancangan Sistem Informasi Perpustakaan Pada SMP Negeri 14 Medan," *J. Sisfotek Glob.*, vol. 8, no. 2, pp. 26–30, 2018, doi: 10.38101/sisfotek.v8i2.184.
- [8] N. Hendrastuty, Y. Ihza, J. Ring Road Utara, and J. Lor, "Rancang Bangun Aplikasi Monitoring Santri Berbasis Android," *Jdmsi*, vol. 2, no. 2, pp. 21–34, 2021.
- [9] I. G. N. Suryantara, "Pengembangan Aplikasi Penjualan Mobil dengan Framework Scrum pada PT XYZ," *Go-Integratif J. Tek. Sist. dan Ind.*, vol. 2, no. 2, pp. 70–85, 2021, doi: 10.35261/gijtsi.v2i2.5326.
- [10] A. Andipradana and K. D. Hartomo, "Rancang Bangun Aplikasi Penjualan Online Berbasis Web Menggunakan Metode Scrum," pp. 161–172, 2021.