



Pengembangan Sistem Informasi Peminjaman Fasilitas Kampus Berbasis *Natural Language Processing* dengan Metode *Extreme Programming*

Mohammad Reza Maulana*¹, Muhammad Rikzam Kamal², Agus Susilo Nugroho³,
Tamamudin⁴

^{1,2,3,4} Universitas Islam Negeri K.H. Abdurrahman Wahid Pekalongan, Pekalongan, Indonesia

E-mail : mohammad.reza.maulana@uingusdur.ac.id*

*Penulis Korespondensi

Received 8 November 2025; Revised 11 December 2025; Accepted 12 December 2025

Abstrak - Sistem peminjaman fasilitas (ruang dan peralatan) di lingkungan Fakultas Ekonomi dan Bisnis Universitas Islam Negeri K.H. Abdurrahman Wahid Pekalongan selama ini masih dilakukan secara manual, sering kali menggunakan formulir *online* dan komunikasi langsung dengan petugas. Proses manual ini menimbulkan berbagai permasalahan, termasuk bentrok jadwal, data yang tidak sinkron, dan kurangnya transparansi mengenai ketersediaan ruang dan aset. Penelitian ini bertujuan untuk mengembangkan Sistem Informasi Peminjaman Ruang dan Peralatan berbasis *Natural Language Processing* (NLP) yang terintegrasi, transparan, dan dapat bekerja secara *real-time*. Metode pengembangan sistem yang digunakan adalah *Extreme Programming* (XP), yang dipilih karena sifatnya yang adaptif, fleksibel, serta fokus pada iterasi cepat dan umpan balik berkelanjutan dari pengguna. Tahapan XP meliputi *planning*, *design*, *coding*, dan *testing*. Sistem informasi dirancang menggunakan *framework* Laravel yang didukung oleh basis data PostgreSQL. Fungsionalitas utama sistem mencakup pengajuan peminjaman *online* oleh Peminjam, validasi konflik jadwal secara otomatis, dan persetujuan bertingkat oleh Admin/Superadmin. Hasil pengembangan sistem informasi menunjukkan bahwa sistem ini mampu menyederhanakan proses, meminimalkan bentrok jadwal, serta menghasilkan peningkatan efisiensi dan transparansi dalam pengelolaan aset fakultas.

Kata Kunci: Sistem Informasi, Peminjaman Ruang dan Peralatan, *Natural Language Processing* (NLP), *Extreme Programming* (XP), Laravel

Abstract - The facility (room and equipment) loan system within the Fakultas Ekonomi dan Bisnis (FEBI), Universitas Islam Negeri K.H. Abdurrahman Wahid Pekalongan is still carried out manually, often using online forms and direct communication with staff. This manual process causes various problems, including schedule conflicts, asynchronous data, and a lack of transparency regarding space and asset availability. This study aims to develop a *Natural Language Processing* (NLP)-based Room and Equipment Loan Information System that is integrated, transparent, and can work in real-time. The system development method used is *Extreme Programming* (XP), which was chosen because of its adaptive, flexible nature, and focus on rapid iteration and continuous feedback from users. The XP stages include planning, design, coding, and testing. The information system is designed using the Laravel framework supported by a PostgreSQL database. The system's main functionality includes online loan applications by Borrowers, automatic validation of schedule conflicts, and approval by the Admin/Superadmin. The results show that this system is able to simplify processes, minimize schedule conflicts, and increase efficiency and transparency in the management of faculty assets.

Keywords: Information System, Room and Equipment Loan, *Extreme Programming* (XP), Laravel



1. PENDAHULUAN

Institusi pendidikan, termasuk perguruan tinggi, saat ini sangat bergantung pada perkembangan teknologi untuk tata kelola internal institusi, mulai dari pengelolaan di bidang administrasi hingga manajemen aset dan sarana prasarana (Fajri & Mujiastuti, 2024; Praharingtyas et al., 2024). Efisiensi dalam pengelolaan sarana dan prasarana (sarpras) sangat penting untuk mendukung kegiatan tridharma perguruan tinggi, seperti kegiatan belajar mengajar, seminar, workshop, dan rapat lembaga sehingga kegiatan-kegiatan dapat dilakukan dengan maksimal dan mendapatkan hasil yang optimal (Fajri & Mujiastuti, 2024).

Fakultas Ekonomi dan Bisnis Islam UIN K.H. Abdurrahman Wahid Pekalongan merupakan salah satu institusi pendidikan yang saat ini fokus pada pengembangan digitalisasi. Salah satu peningkatan layanan yang akan dioptimalkan dalam hal digitalisasi adalah proses peminjaman fasilitas. Berdasarkan wawancara dengan petugas pelayanan peminjaman fasilitas, pelayanan proses peminjaman masih dilakukan secara manual, yaitu melalui formulir online atau komunikasi langsung dengan petugas. Hal ini menyebabkan sejumlah masalah, antara lain adanya jadwal penggunaan ruangan yang bersamaan (bentrok jadwal), ketersediaan barang yang terbatas tidak terdeteksi (data tidak sinkron), tidak bisa mengecek ketersediaan fasilitas secara *realtime* dan waktu yang terbuang lama untuk mendapatkan konfirmasi persetujuan. Kesulitan dalam mengelola jadwal dan ketidakefisienan dalam proses peminjaman menunjukkan perlunya solusi digital.

Untuk mengatasi masalah tersebut, salah satu solusi yang dapat diterapkan adalah pengembangan sistem informasi peminjaman ruang dan alat berbasis website. Hal ini bertujuan untuk mempermudah proses pengajuan, persetujuan, dan pengelolaan peminjaman, menyediakan informasi ketersediaan aset secara real-time, serta meminimalkan bentrok jadwal dan kesalahan pencatatan serta mempermudah pengguna dalam melakukan peminjaman ruang dan peralatan (Purwanto et al., 2025; Fajri & Mujiastuti, 2024). Penelitian sebelumnya yang dilakukan oleh Purwanto (2025), pengembangan sistem informasi peminjaman fasilitas meningkatkan efisiensi pengelolaan peminjaman, mengurangi waktu dan biaya, serta memberikan akses yang lebih mudah kepada pengguna (Purwanto et al., 2025).

Penerapan *Natural Language Processing* (NLP) pada sistem informasi akan meningkatkan efisiensi penggunaan sistem dan membantu pengguna dalam melakukan peminjaman. Penelitian yang dilakukan oleh Nurlinta (2025) menunjukkan integrasi NLP dalam chatbot dapat meningkatkan layanan kepada pelanggan melalui otomatisasi *customer service* (Nurlita & Munawaroh, 2025). Pada penelitian lain, pemanfaatan NLP untuk otomatisasi chatbot juga meningkatkan efisiensi layanan konsultasi pada klinik Gigi dan membantu pasien mendapatkan informasi yang lebih cepat (Ibadurrahman et al., 2025).

Dalam pengembangan sistem informasi, terdapat metode atau pendekatan dalam proses pengembangannya. Metode pengembangan sistem yang digunakan adalah metode *Extreme Programming* (XP). XP merupakan metodologi pengembangan software yang fleksibel, adaptif, dan berupaya meningkatkan efisiensi (Anike et al., 2023; Muh. Fahmi Rustan et al., 2022; Sinlae et al., 2024). XP sangat cocok digunakan untuk sistem yang kebutuhannya sering berubah-ubah atau untuk pengembangan sistem dengan waktu yang singkat (Galuh Wilujeng Saraswati & Malik Aziz Ali, 2024; Muh. Fahmi Rustan et al., 2022; Ramadhani & Fitriana, 2023; Sakinah & Ulfa, 2024). Penerapan XP memungkinkan penyederhanaan tahapan pengembangan sistem informasi sehingga menjadi lebih mudah dan cepat.

Hasil penelitian dari Beberapa penelitian sebelumnya juga telah menunjukkan bahwa XP dapat diterapkan secara efektif dalam pengembangan berbagai sistem informasi, termasuk sistem informasi tracer study, sistem informasi akreditasi, dan sistem informasi inventaris (Muh. Fahmi Rustan et al., 2022; Praharingtyas et al., 2024; Ramadhani & Fitriana, 2023). Dengan



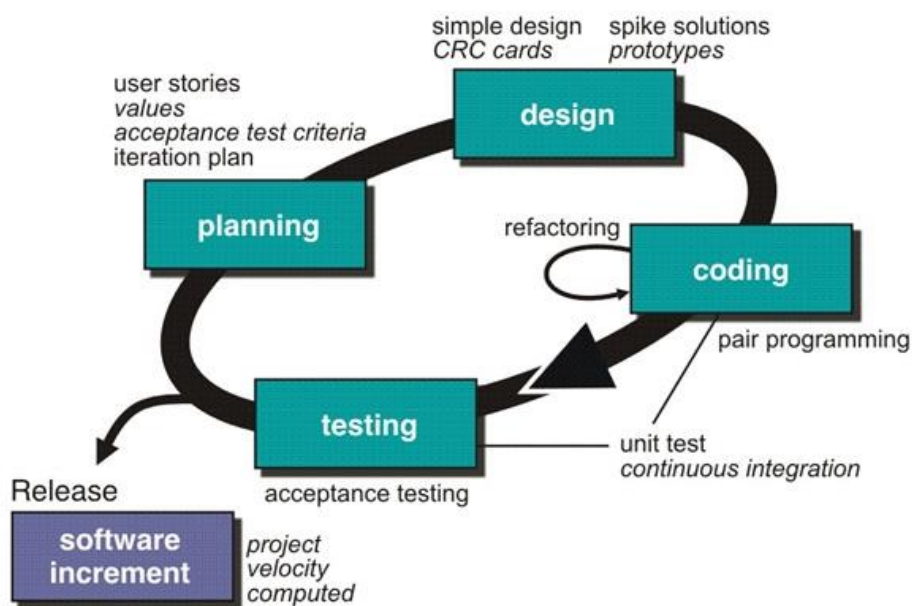
demikian, penelitian ini akan berfokus pada pengembangan sistem informasi peminjaman fasilitas berbasis NLP menggunakan metode XP untuk memastikan bahwa produk yang dihasilkan sesuai dengan kriteria dan kebutuhan pengguna secara cepat. Kebaruan penelitian ini terletak pada integrasi NLP pada sistem peminjaman fasilitas yang dikembangkan dengan metode XP.

2. METODE PENELITIAN

Metode pengembangan sistem wajib digunakan sebagai metodologi untuk mengembangkan sistem informasi. Terdapat beberapa metode pengembangan sistem dengan pendekatan Agile yang umum digunakan seperti XP, Scrum, *Pair Programming* dan Kanban. Metode XP akan digunakan pada penelitian ini dikarenakan cocok untuk tim pengembang kecil dan iterasi rilis aplikasi yang cepat (Salza et al., 2019).

XP adalah metodologi pengembangan yang diadopsi untuk memajukan mutu perangkat lunak dan bersifat tanggap terhadap permintaan dari pelanggan (Sinlae et al., 2024). XP bertujuan menyederhanakan tahapan pengembangan sistem sehingga menjadi lebih adaptif, efisien, dan fleksibel (Anike et al., 2023; Galuh Wilujeng Saraswati & Malik Aziz Ali, 2024; Muh. Fahmi Rustan et al., 2022).

Tahapan XP yang digunakan dalam pengembangan sistem ini terdiri dari empat fase utama, yaitu *Planning*, *Design*, *Coding*, dan *Testing* (Hariyanto et al., 2024; Sankhe et al., 2022).



Gambar 1. Tahapan Penelitian pada Pengembangan Sistem Informasi Peminjaman Fasilitas

Gambar 1 menunjukkan siklus pengembangan perangkat lunak menggunakan metode *Extreme Programming (XP)*. Model ini menunjukkan proses iteratif dan inkremental yang menekankan kolaborasi erat antara pengguna dan pengembang, serta fokus pada kualitas kode dan respon cepat terhadap perubahan kebutuhan.

Siklus dimulai dari tahap *planning* (perencanaan), di mana dilakukan identifikasi *user stories*, penentuan nilai bisnis, kriteria *acceptance test*, serta penyusunan rencana iterasi. Tahap ini menjadi dasar bagi tim untuk memahami kebutuhan pengguna secara mendalam dan menentukan prioritas pengembangan setiap iterasi.



Selanjutnya, proses berlanjut ke tahap design (perancangan). Pada tahap ini, dilakukan penyusunan desain sederhana (*simple design*). Prinsip utama pada tahap ini adalah menghasilkan rancangan yang cukup untuk memenuhi kebutuhan saat ini tanpa kompleksitas berlebihan termasuk rencana penerapan NLP menggunakan library python yang sederhana namun *powerfull*. Alat bantu desain yang digunakan adalah *Unified Modeling Language* (UML).

Tahap berikutnya adalah *coding* (pengkodean), yang dilaksanakan berdasarkan desain yang telah disusun. Dalam praktik XP, tahap ini menekankan kegiatan *pair programming* (dua programmer bekerja bersama pada satu komputer), *refactoring* (penyempurnaan struktur kode tanpa mengubah perilaku), serta *continuous integration* (penggabungan kode secara berkala agar sistem selalu dapat dijalankan dan diuji). Tahap *coding* dilakukan dengan mengembangkan 2 sistem utama yaitu, pengembangan service NLP menggunakan python dan pengembangan aplikasi website dengan laravel untuk melakukan manajemen peminjaman.

Setelah proses pengkodean, sistem akan melalui tahap testing (pengujian). Setiap unit kode diuji melalui *unit test* untuk memastikan fungsionalitasnya berjalan sesuai rancangan, kemudian dilakukan *acceptance testing* untuk memverifikasi kesesuaian sistem dengan kebutuhan pengguna.

Apabila sistem telah memenuhi kriteria sesuai kebutuhan pengguna, maka dilakukan proses release (rilis) sebagai *software increment*. Hasil dari setiap iterasi menghasilkan versi sistem informasi yang dapat digunakan oleh pengguna, sekaligus memberikan umpan balik bagi pengembangan pada iterasi berikutnya.

3. HASIL DAN PEMBAHASAN

3.1. *Planning* (perencanaan)

Pada tahap ini proses pengumpulan data dilakukan dengan melakukan wawancara kepada petugas yang bertanggung jawab dalam menangani proses peminjaman fasilitas. Setelah data didapatkan, tahap selanjutnya adalah menentukan hak akses pengguna sistem dan menentukan kebutuhan fungsional dan non-fungsional.

a. Hak akses pengguna

Sistem informasi peminjaman ruang dan alat akan digunakan oleh beberapa tipe pengguna. Hak akses untuk tipe pengguna tersebut dibedakan menjadi 3 yaitu:

- 1) Superadmin: Mengelola seluruh data, *user*, hak akses (manajemen *user* dan *role*), dan konfigurasi sistem.
- 2) Admin (Petugas Peminjaman): Mengelola data master seperti fasilitas, mengelola peminjaman, melakukan *approval/decline*, dan memantau status aset.
- 3) Peminjam (Dosen, Mahasiswa, Tendik): Mengajukan peminjaman ruang atau peralatan, melihat ketersediaan, dan memantau status pengajuan.

b. *User Stories* pengguna

User Stories pengguna merupakan bagian penting agar sistem yang dibangun sesuai dengan kebutuhan pengguna. Tabel 1 merupakan *user stories* utama dari pengguna sistem.



Tabel 1. *User Stories* pengguna sistem informasi

Kode	Judul <i>User Story</i>	Aktor	Deskripsi (Sebagai... saya ingin... sehingga...)	<i>Acceptance Criteria</i>
S1	Autentikasi	User	Sebagai user, saya ingin login menggunakan email dan password sehingga saya dapat mengakses sistem sesuai dengan peran saya.	1) Hanya user valid yang dapat login. 2) Role menentukan hak akses. 3) Pesan error muncul jika login gagal.
S2	Manajemen User	Superadmin	Sebagai superadmin, saya ingin melakukan CRUD data user dan menetapkan role agar pengelolaan hak akses lebih terkontrol.	1) Superadmin dapat menambah, ubah, hapus, dan lihat data user. 2) Role dapat diatur untuk tiap user.
S3	Manajemen Data Master	Admin	Sebagai admin, saya ingin mengelola data fasilitas agar data yang tersedia untuk peminjaman selalu akurat.	1) CRUD fasilitas tersedia. 2) Tidak dapat hapus jika sedang digunakan.
S4	Peminjaman Ruang & Peralatan	User	Sebagai user, saya ingin mengajukan peminjaman ruang/peralatan agar dapat menggunakan fasilitas sesuai kebutuhan.	1) User dapat isi form peminjaman. 2) Data tersimpan dengan status "pending". 3) Dapat melihat riwayat.
S5	Validasi Jadwal Otomatis	Sistem	Sebagai sistem, saya ingin memeriksa konflik dengan jadwal lain agar tidak terjadi bentrok pemakaian.	1) Sistem menolak jika waktu bentrok. 2) Validasi otomatis sebelum simpan.
S6	Persetujuan Admin	Admin	Sebagai admin, saya ingin menyetujui atau menolak peminjaman agar penggunaan ruang/aset dapat dikontrol.	1) Admin dapat ubah status menjadi approved/declined 2) Notifikasi dikirim ke user.
S7	Monitoring Ruang & Aset	Admin	Sebagai admin, saya ingin melihat ruang/aset yang sedang dipinjam agar dapat memantau ketersediaan.	1) Menampilkan status realtime. 2) Ada indikator visual ketersediaan.
S8	Pelaporan	Admin/ Superadmin	Sebagai admin/superadmin, saya ingin mengunduh laporan peminjaman dalam format PDF/Excel agar dapat digunakan untuk analisis dan arsip.	1) Laporan bisa difilter. 2) Format PDF/Excel tersedia.

c. Kebutuhan fungsional dan non-fungsional

Kebutuhan fungsional merupakan hasil ringkasan dari *user stories* yang telah diidentifikasi pada tahap sebelumnya. Tabel 2 menunjukkan kebutuhan fungsional dan tabel 3 menunjukkan kebutuhan non-fungsional sistem informasi peminjaman fasilitas.



Tabel 2. Kebutuhan Fungsional Sistem Informasi Peminjaman Fasilitas

Kode	Kebutuhan Fungsional	Deskripsi
F1	Autentikasi	User dapat login dengan email & password, role dibedakan berdasarkan kolom role.
F2	Manajemen user	Superadmin dapat melakukan CRUD data user dan menetapkan role.
F3	Manajemen data master fasilitas	Admin dapat melakukan manajemen data fasilitas yang nantinya dapat dipinjam oleh peminjam
F4	Peminjaman fasilitas	User mengajukan peminjaman ruang dan/atau peralatan melalui form yang terhubung ke data peminjaman.
F5	Validasi jadwal otomatis	Sistem memeriksa konflik dengan jadwal perkuliahan atau jadwal lainnya sebelum menyimpan peminjaman.
F6	Persetujuan admin	Admin dapat mengubah status peminjaman menjadi <i>approved</i> atau <i>declined</i> .
F7	Monitoring ruang & aset	Admin dapat melihat ruang yang sedang dipinjam, kondisi aset, dan status pemakaian.
F8	Pelaporan	Admin dan superadmin dapat mengunduh laporan peminjaman dalam format PDF/Excel.

Tabel 3. Kebutuhan Non-Fungsional Sistem Informasi Peminjaman Fasilitas

Jenis	Deskripsi
Keamanan	Menggunakan autentikasi Laravel, validasi input, proteksi CSRF, <i>role-based access</i> .
Reliabilitas	Transaksi peminjaman harus konsisten dan atomic (menggunakan <i>database transaction</i>).
<i>Usability</i>	Antarmuka sederhana, responsif, dan ramah pengguna.
Portabilitas	Dapat diakses melalui desktop dan perangkat mobile.
Pemeliharaan	Menggunakan arsitektur MVC Laravel untuk memudahkan maintenance.

3.2. Design (perancangan)

a. Arsitektur sistem

Dalam pengembangan sebuah sistem informasi, arsitektur merupakan sebuah pondasi awal yang perlu diperhatikan agar sistem informasi yang dihasilkan dapat berjalan optimal. Terdapat 2 arsitektur utama sistem yaitu arsitektur sistem informasi peminjaman berbasis website dan arsitektur service NLP.

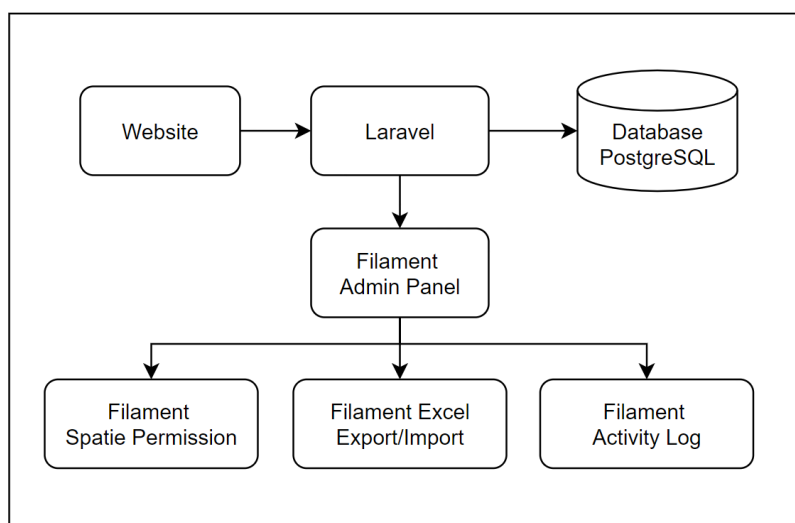
Pada Gambar 1 dapat dilihat bahwa arsitektur sistem yang dikembangkan pada penelitian ini dirancang berbasis website agar dapat diakses secara luas melalui peramban (web browser) tanpa perlu instalasi perangkat lunak tambahan pada sisi pengguna. Pendekatan berbasis web dipilih karena memberikan kemudahan dalam hal pemeliharaan, pembaruan sistem, serta kompatibilitas lintas platform.

Secara teknis, sistem ini dibangun menggunakan framework Laravel sebagai *core framework* pada sisi server. Laravel dipilih karena memiliki struktur pengembangan yang modular, mendukung prinsip *Model-View-Controller* (MVC), serta menyediakan berbagai komponen bawaan untuk manajemen autentikasi, otorisasi, validasi, dan pengelolaan basis data (Singla, 2025). Penerapan arsitektur MVC dalam Laravel membantu memisahkan logika aplikasi, antarmuka pengguna, dan manajemen data secara terstruktur, sehingga meningkatkan keterbacaan dan skalabilitas kode.

Sistem menggunakan PostgreSQL sebagai basis data utama. PostgreSQL dipilih karena merupakan sistem manajemen basis data relasional yang andal, mendukung *transaction integrity*, serta memiliki kemampuan penanganan data yang kompleks melalui tipe data lanjutan, relasi antar tabel, dan constraint referensial (*PostgreSQL: About*, n.d.). Integrasi antara Laravel dan PostgreSQL dilakukan melalui *Object-Relational Mapping* (ORM) yang disediakan



oleh *Eloquent*, sehingga interaksi terhadap basis data menjadi lebih efisien dan mudah dikelola melalui model yang terstruktur.



Gambar 1. Arsitektur sistem informasi peminjaman fasilitas

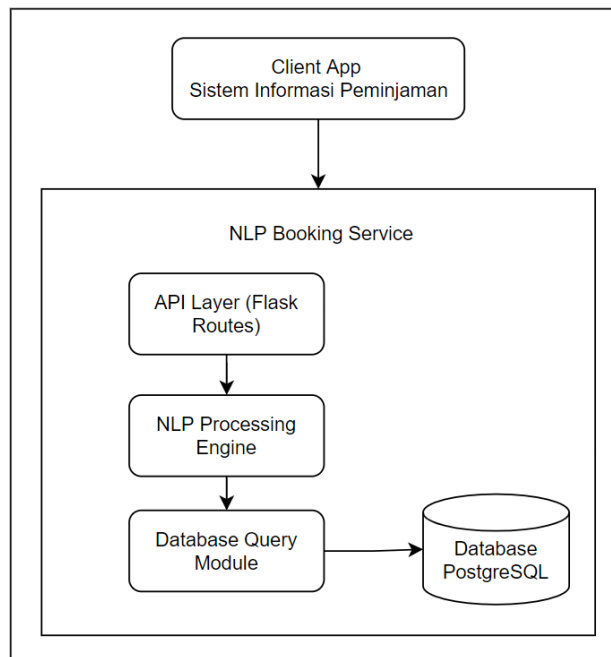
Untuk mendukung antarmuka dan pengelolaan data secara administratif, sistem ini memanfaatkan *Filament Admin Panel*, yaitu framework admin *interface* berbasis Laravel yang menyediakan komponen antarmuka modern, interaktif, dan responsif. Filament digunakan untuk membangun modul-modul seperti manajemen pengguna, validasi peminjaman, pengelolaan aset, dan pelaporan. Selain itu, sistem juga mengintegrasikan beberapa plugin Filament guna meningkatkan fungsionalitas, antara lain:

- 1) *Filament Spatie Permission* untuk pengelolaan peran dan hak akses pengguna (*Role-Based Access Control*).
- 2) *Filament Excel Import/Export* untuk mendukung proses impor dan ekspor data dari dan ke format Excel.
- 3) *Filament Activity Log* untuk mencatat aktivitas pengguna secara otomatis sebagai bagian dari sistem audit trail.

Secara keseluruhan, arsitektur sistem informasi peminjaman menggabungkan keunggulan framework Laravel, kekuatan PostgreSQL dalam pengelolaan data, serta kemudahan manajemen melalui Filament, sehingga menghasilkan sistem informasi berbasis web yang aman, efisien, dan mudah dikembangkan lebih lanjut sesuai kebutuhan organisasi.

Selanjutnya untuk arsitektur service NLP merupakan bagian penting agar aplikasi client dapat menggunakan fitur pencarian berbasis bahasa natural. Gambar 2 menunjukkan arsitektur sistem peminjaman ruang dan peralatan berbasis Natural Language Processing (NLP). Pengguna berinteraksi melalui Client App, yaitu aplikasi sistem informasi peminjaman yang mengirimkan permintaan berbasis bahasa alami ke layanan NLP *Booking Service*.

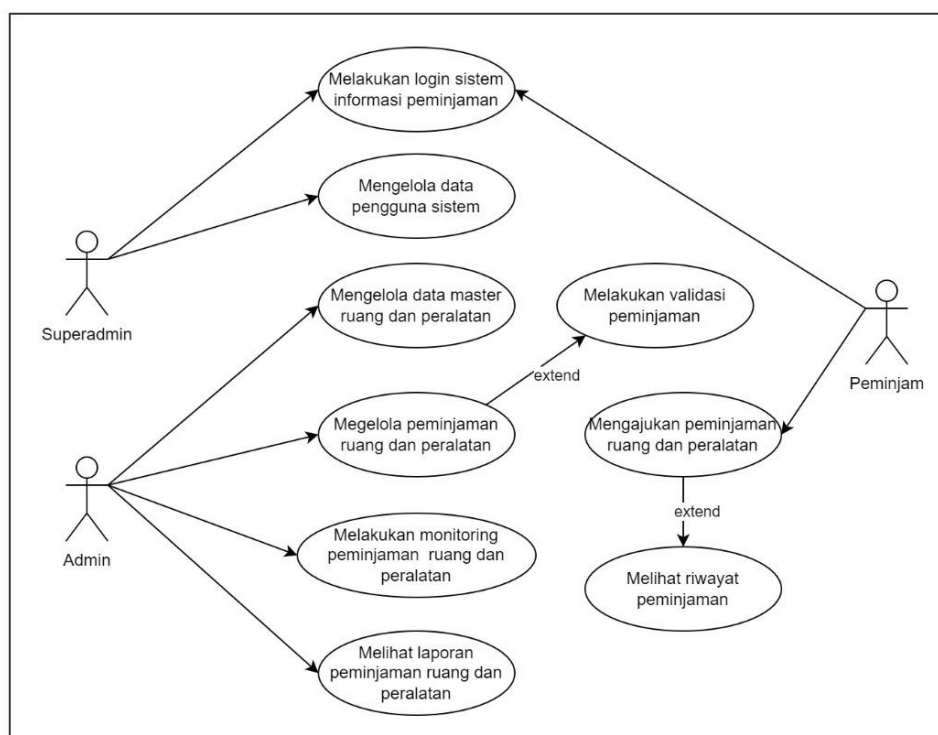
Di dalam layanan ini, API Layer berbasis Flask menerima input pengguna dan meneruskannya ke *NLP Processing Engine* untuk dianalisis. Mesin NLP kemudian mengidentifikasi maksud perintah serta mengekstraksi informasi penting seperti jenis ruang, tanggal, atau waktu yang dimaksud. Hasil analisis tersebut diteruskan ke *Database Query Module* untuk diubah menjadi perintah terstruktur dan dieksekusi pada Database PostgreSQL. Data yang diperoleh, seperti ketersediaan ruang atau hasil pemesanan, kemudian dikembalikan lagi melalui API dan ditampilkan ke pengguna dalam aplikasi. Dengan arsitektur ini, proses peminjaman menjadi lebih intuitif karena pengguna cukup menggunakan kalimat biasa tanpa harus memahami format sistem secara teknis.



Gambar 2. Arsitektur service NLP

b. Desain diagram *Use Case* sistem dan database

Diagram *Use Case* digunakan untuk menggambarkan hubungan aktor khususnya pengguna dengan sistem. Berikut *Use Case* sistem informasi peminjaman fasilitas.



Gambar 3. Diagram *Use Case* sistem informasi peminjaman fasilitas

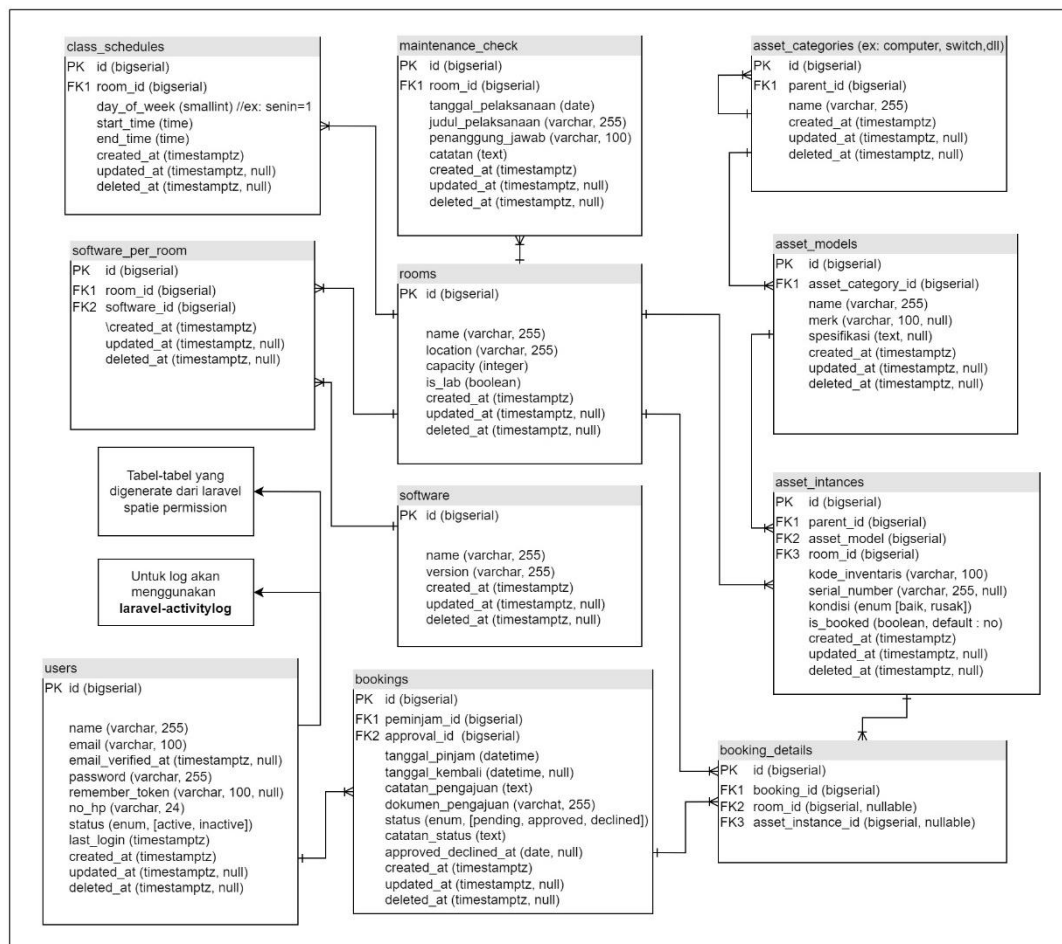
Dari Gambar 3 dapat dilihat bahwa sistem informasi peminjaman fasilitas dirancang untuk memfasilitasi proses pengelolaan data, pengajuan, serta pemantauan kegiatan



peminjaman secara terintegrasi. Sistem ini melibatkan tiga aktor utama, yaitu Superadmin, Admin, dan Peminjam, dengan peran dan tanggung jawab yang berbeda.

Superadmin memiliki hak akses tertinggi untuk melakukan login ke sistem, mengelola data pengguna, serta mengelola data master fasilitas. Superadmin juga dapat memantau seluruh aktivitas dalam sistem untuk memastikan operasional berjalan dengan baik. Admin berperan dalam operasional harian sistem dengan kemampuan untuk mengelola data master fasilitas, mengelola peminjaman fasilitas, melakukan monitoring peminjaman, serta melihat laporan peminjaman. Admin juga memiliki wewenang untuk melakukan validasi terhadap pengajuan peminjaman yang dilakukan oleh pengguna. Peminjam sebagai pengguna akhir dapat melakukan login ke sistem dan mengajukan peminjaman ruang atau peralatan sesuai kebutuhan. Setelah melakukan pengajuan, peminjam dapat melihat riwayat peminjaman sebagai bentuk transparansi dan dokumentasi aktivitas peminjaman. Relasi *extend* menunjukkan bahwa proses validasi merupakan bagian lanjutan dari pengelolaan peminjaman oleh admin, sedangkan akses melihat riwayat merupakan bagian tambahan dari pengajuan peminjaman oleh peminjam.

Selanjutnya desain database dilakukan untuk memastikan data yang digunakan dalam sistem telah dinormalisasi untuk mengurangi *data redundancy*. Berikut desain database beserta relasi antar tabelnya.



Gambar 4. Desain database dan relasi tabel sistem informasi peminjaman fasilitas

Gambar 4 menunjukkan rancangan basis data untuk mendukung pengelolaan peminjaman fasilitas secara terintegrasi serta memastikan keterlacakan data secara akurat.



Struktur basis data terdiri dari beberapa kelompok entitas utama yang merepresentasikan fungsi dan proses dalam sistem.

Entitas users menyimpan data seluruh pengguna sistem, termasuk superadmin, admin, dan peminjam. Tabel ini juga mendukung integrasi dengan paket *Spatie Permission* untuk pengelolaan hak akses serta *Laravel Activity Log* untuk pencatatan aktivitas pengguna. Entitas rooms menyimpan data ruang yang dapat dipinjam, meliputi nama, lokasi, kapasitas, dan status laboratorium. Sementara itu, entitas asset_categories, asset_models, dan asset_instances digunakan untuk mengelola data aset atau peralatan, mulai dari kategori, model, hingga unit fisik dengan informasi seperti kondisi, nomor seri, dan keterhubungan dengan ruang tertentu.

Tabel bookings berfungsi untuk mencatat proses peminjaman yang dilakukan oleh pengguna, termasuk data peminjam, tanggal pinjam, tanggal kembali, status persetujuan, serta dokumen pendukung. Detail dari setiap peminjaman direpresentasikan dalam tabel booking_details, yang menghubungkan antara ruang atau aset dengan data peminjaman utama. Untuk mendukung keterpaduan kegiatan akademik dan operasional, terdapat tabel class_schedules yang merekam jadwal penggunaan ruang untuk kegiatan perkuliahan, serta maintenance_check yang mencatat aktivitas perawatan ruang atau peralatan. Kedua tabel ini berperan penting dalam memastikan tidak terjadi konflik jadwal peminjaman.

Selain itu, entitas software dan software_per_room digunakan untuk mendata perangkat lunak yang terpasang pada setiap ruang atau peralatan, sehingga sistem juga dapat mencatat informasi teknis secara menyeluruh. Secara keseluruhan, rancangan basis data ini dirancang dengan prinsip normalisasi dan keterhubungan antarentitas yang jelas, sehingga mampu mendukung fungsi utama sistem dalam pengelolaan peminjaman, monitoring, dan pelaporan secara efisien dan akurat.

3.3. Coding (pengkodean)

Tahap *coding* atau Implementasi sistem informasi peminjaman fasilitas dilakukan dengan mengacu pada rancangan arsitektur yang telah ditetapkan dan kebutuhan fungsional serta kebutuhan non-fungsional. Sistem dikembangkan berbasis website menggunakan framework Laravel dengan bahasa pemrograman PHP dan basis data PostgreSQL. Proses implementasi dimulai dengan inisialisasi proyek Laravel dan konfigurasi koneksi basis data PostgreSQL melalui file lingkungan (.env).

```
_000000_create_users_table.php
_000001_create_cache_table.php
_000002_create_jobs_table.php
_044939_create_permission_tables.php
_045115_create_activity_log_table.php
_045116_add_event_column_to_activity_log_table.php
_045117_add_batch_uuid_column_to_activity_log_table.php
_045141_create_notifications_table.php
_061720_create_rooms_table.php
_061838_create_asset_categories_table.php
_061918_create_asset_models_table.php
_064734_create_asset_instances_table.php
_064819_create_software_table.php
_064936_create_software_per_room_table.php
_065207_create_class_schedules_table.php
_065255_create_maintenance_checks_table.php
_065322_create_bookings_table.php
_065418_create_booking_details_table.php

10 class Booking extends Model
11 {
12     use SoftDeletes;
13
14     protected $fillable = [ ...
15 ];
16
17     protected $casts = [ ...
18 ];
19
20     protected static function booted()
21     { ...
22     }
23
24     public function peminjam(): BelongsTo
25     {
26         return $this->belongsTo(User::class, 'pemohon_id');
27     }
28
29     public function approver(): BelongsTo
30     {
31         return $this->belongsTo(User::class, 'approval_id');
32     }
33
34     public function bookingDetails(): HasMany
35     {
36         return $this->hasMany(BookingDetail::class);
37     }
38 }
```

Gambar 5. Migrasi tabel database dan Model class Booking



Seluruh entitas utama seperti pengguna, ruang, peralatan, dan peminjaman direpresentasikan dalam model Eloquent, sementara migrasi digunakan untuk membentuk tabel-tabel sesuai rancangan database seperti yang terlihat pada Gambar 5. Interaksi antara sistem dan basis data dilakukan melalui *Object-Relational Mapping* (ORM) bawaan Laravel untuk menjaga efisiensi serta konsistensi data.

Untuk mendukung pengelolaan data administratif, sistem dilengkapi dengan Filament Admin Panel yang berfungsi sebagai antarmuka manajemen internal. Filament memudahkan pembuatan modul seperti manajemen pengguna, validasi peminjaman, serta pelaporan. Beberapa plugin tambahan juga diintegrasikan untuk memperluas fungsi sistem, antara lain Filament Spatie Permission untuk manajemen peran dan hak akses, Filament Excel Export/Import untuk ekspor dan impor data, serta Filament Activity Log untuk pencatatan aktivitas pengguna sebagai bentuk audit trail. Sebagai contoh halaman peminjaman fasilitas dapat dilihat pada Gambar 6.

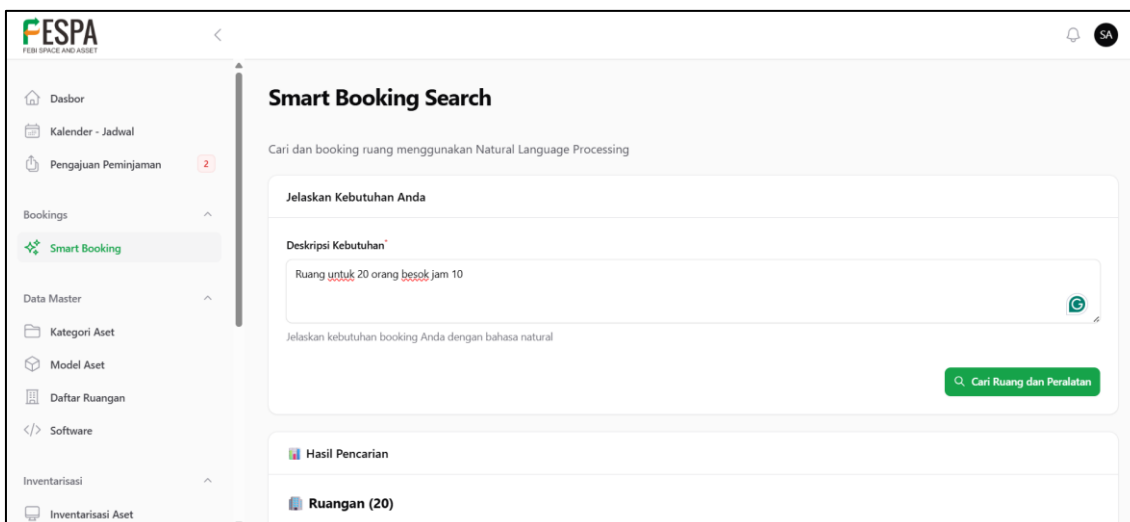
Selain itu, sistem dilengkapi fitur validasi jadwal otomatis yang memeriksa potensi konflik dengan jadwal perkuliahan atau peminjaman lain sebelum data disimpan ke dalam basis data. Antarmuka pengguna dikembangkan menggunakan Blade Template Engine dengan dukungan Tailwind CSS agar responsif dan mudah digunakan. Hasil implementasi menunjukkan bahwa seluruh komponen sistem saling terintegrasi dengan baik, mendukung proses peminjaman fasilitas secara efisien, terkontrol, dan dapat dipantau melalui panel administrasi.

No	Pembuat	Data Peminjam	Tipe Peminjam	Tanggal Peminjaman	Tanggal Dibuat	Ruangan
1	ukmkordaisnusa	Nailatul Thisania ID: [REDACTED] No: [REDACTED]	mahasiswa	Pinjam : 08 Nov 2025 07:00 Kembali : 08 Nov 2025 17:00	21 Okt 2025 14:16	Meeting Room
2	Merwa	Naf'ah Labiba Al-Huda ID: [REDACTED] No: [REDACTED]	mahasiswa	Pinjam : 15 Nov 2025 12:00 Kembali : 15 Nov 2025 18:00	28 Okt 2025 11:57	Halaman Depan FEBI Meeting Room R. Podcast
		Kamaedi		Pinjam : 09 Nov 2025 08:00		

Gambar 6. Halaman data peminjaman fasilitas

Dari sisi *service NLP*, python digunakan sebagai bahasa pemrograman utama. Terdapat beberapa library yang digunakan seperti flask, psychopg2 dan fuzzywuzzy. Library utama untuk pemrosesan NLP menggunakan fuzzywuzzy, dimana library ini bertugas untuk mencocokkan nilai string deskripsi yang dituliskan pengguna yang sudah dilakukan *extraction* dengan data yang ada di database dengan sistem *scoring* kecocokan nilai.

Gambar 7 merupakan halaman yang digunakan pengguna untuk melakukan *booking* ruang dan peralatan menggunakan bahasa natural. Sebagai contoh, pengguna memasukkan kalimat “Ruang untuk 20 orang besok jam 10”, maka kalimat tersebut akan diproses oleh NLP dan selanjutnya melakukan pencarian data di database. Jika data ditemukan maka akan ditampilkan daftar ruangan yang tersedia sesuai kriteria. Setelah itu pengguna bisa memilih ruang dan/atau peralatan untuk dipinjam.



Gambar 7. Halaman peminjaman fasilitas menggunakan NLP

3.4. Testing (pengujian)

a. Unit testing

Unit testing merupakan salah pengujian penting untuk memastikan kualitas kode program dan meminimalkan munculnya bug yang ada di sistem. Alat yang digunakan untuk pengujian *unit testing* ini adalah menggunakan extension PHPUnit Test Explorer di Visual Studio Code. Gambar 8 merupakan salah contoh hasil pengujian salah satu halaman sistem informasi yaitu halaman pemesanan ruang dan/atau peralatan menggunakan NLP.

```
PASS Tests\Feature\SmartSearchTest
✓ smart search api with valid query 2.10s
✓ smart search api with empty query 0.72s
✓ smart search api with missing query 0.72s
✓ smart search api with nlp service error 0.71s
✓ smart search api service unavailable 0.71s
✓ smart search api with no results 0.77s
✓ smart search api with multiple results 0.76s
✓ smart search returns query field 0.70s
✓ smart search date extraction 0.71s
✓ smart search time extraction 0.72s
✓ smart search response time 0.78s
✓ smart search nlp service integration 1.12s
✓ smart search with complex query 1.07s
✓ smart search error handling nlp timeout 1.04s
✓ smart search malformed nlp response 0.93s
✓ smart search filters by capacity 1.07s
✓ smart search filters by equipment 0.77s
✓ smart search room has availability field 0.97s
✓ smart search room has match score 1.03s

Tests: 25 passed (49 assertions)
Duration: 22.45s
```

Gambar 8. Hasil *unit testing* halaman login

3.4.2. Acceptance testing

Setelah aplikasi berfungsi, tahap berikutnya adalah melakukan pengujian penerimaan pengguna. Hal ini dilaksanakan agar sistem informasi yang telah dibangun sesuai dengan kebutuhan pengguna. Tabel 4 merupakan hasil *acceptance testing* sesuai kriteria yang telah ditentukan pada tahap *user journey*.



Tabel 4. Hasil *acceptance testing* sistem informasi peminjaman ruang dan alat

Kode	Judul <i>User Story</i>	Aktor	<i>Acceptance Criteria</i>	Hasil Pengujian	Keterangan
AT1	Autentikasi	User	1) Hanya user valid yang dapat login. 2) Role menentukan hak akses. 3) Pesan error muncul jika login gagal.	Berhasil	Sistem menolak login dengan kredensial salah dan mengarahkan user sesuai perannya setelah berhasil login.
AT2	Manajemen User	Super admin	1) CRUD data user tersedia. 2) Role dapat diatur untuk tiap user.	Berhasil	Superadmin dapat menambah, mengubah, dan menghapus data user serta mengatur role menggunakan Filament Spatie Permission.
AT3	Manajemen Data Master	Admin	1) CRUD fasilitas tersedia. 2) Tidak dapat hapus jika sedang digunakan.	Berhasil	Fungsi CRUD berjalan baik, sistem menolak penghapusan data yang sedang digunakan dalam tabel peminjaman.
AT4	Peminjaman Ruang & Peralatan	User	1) User dapat mengisi form peminjaman. 2) Data tersimpan dengan status "pending". 3) Dapat melihat riwayat.	Berhasil	Form peminjaman berfungsi dan data tersimpan dengan status awal "pending". Riwayat peminjaman tampil sesuai pengguna.
AT5	Validasi Jadwal Otomatis	Sistem	1) Sistem menolak jika waktu bentrok. 2) Validasi otomatis sebelum simpan.	Berhasil	Sistem melakukan pemeriksaan bentrok secara otomatis dan menampilkan pesan error saat jadwal tumpang tindih.
AT6	Persetujuan Admin	Admin	1) Admin dapat ubah status menjadi approved/declined. 2) Notifikasi dikirim ke user.	Berhasil	Admin dapat mengubah status peminjaman dan sistem mengirimkan notifikasi status melalui dashboard user.
AT7	Monitoring Ruang & Aset	Admin	1) Menampilkan status real-time. 2) Ada indikator visual ketersediaan.	Berhasil	Dashboard menampilkan daftar ruang dan aset dengan indikator warna menunjukkan status ketersediaan.
AT8	Pelaporan	Admin / Super admin	1) Laporan dapat difilter. 2) Format PDF/Excel tersedia.	Berhasil	Laporan dapat diunduh sesuai rentang waktu atau status tertentu, tersedia format PDF dan Excel.

Hasil pengujian pada Tabel 4 menunjukkan bahwa seluruh fitur utama telah berfungsi sesuai dengan kriteria penerimaan yang ditetapkan. Proses autentikasi berhasil membedakan hak akses berdasarkan peran pengguna, dan sistem mampu menolak login tidak valid. Fungsi manajemen data pengguna, ruang, dan peralatan berjalan dengan baik.

Fitur pengajuan peminjaman dan validasi jadwal otomatis bekerja sesuai rancangan, di mana sistem dapat mendeteksi konflik waktu dan menolak peminjaman yang bentrok. Proses



persetujuan peminjaman oleh admin juga berjalan lancar dengan pengiriman notifikasi status kepada pengguna. Selain itu, tampilan monitoring ruang dan aset menampilkan informasi ketersediaan secara visual, dan fitur pelaporan mendukung ekspor ke format PDF maupun Excel.

Secara keseluruhan, hasil acceptance test menunjukkan bahwa sistem informasi peminjaman fasilitas telah memenuhi kebutuhan fungsional dan dapat dioperasikan dengan baik oleh seluruh aktor sesuai perannya.

3.5. Software increment/release

Tahapan *software increment/release* dilakukan setelah seluruh proses pada metode Extreme Programming (XP) meliputi *planning*, *design*, *coding*, dan *testing* terselesaikan. Pada fase ini, setiap *user story* yang telah diuji dan memenuhi kriteria penerimaan (acceptance criteria) digabungkan menjadi satu kesatuan sistem yang siap dirilis melalui proses *deployment*.

Setiap *increment* mencerminkan penambahan fungsi yang bernilai langsung bagi pengguna. Proses *release* dilakukan secara bertahap agar pengguna dapat segera memanfaatkan fitur utama, sekaligus memberikan ruang untuk umpan balik sebelum pengembangan dilanjutkan ke fitur berikutnya. Selain itu ketika ada bug yang muncul dan telah diselesaikan, maka proses *deployment* akan langsung dilakukan. Gambar 9 merupakan *pipeline* penerapan *Continuous Deployment* yang dilakukan dengan Gitlab untuk proses deployment sistem informasi secara otomatis.

Status	Pipeline	Created by	Stages	Actions
Passed 00:00:54 3 weeks ago	update composer and fix report i... #36 main -> 8cc54e18 latest branch		✓	Download
Passed 00:00:54 3 weeks ago	add field and update some colu... #35 main -> a5d78aa2 branch		✓	Download
Passed 00:00:32 1 month ago	fix some bug in calender and bo... #34 main -> 63078314 branch		✓	Download
Passed 00:00:28	edit link template surat peminja... #33 main -> 98a38742		✓	Download

Gambar 9. *Continuous Deployment* untuk *increment/release* sistem informasi peminjaman fasilitas

Dari Gambar 9 dapat dilihat bahwa proses *deployment* sistem informasi dilakukan secara bertahap. Hal ini dilakukan untuk memastikan pengguna dapat dengan segera memanfaatkan fitur yang dikembangkan atau fitur yang telah diperbaiki.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem informasi peminjaman ruangan dan peralatan berbasis NLP yang mampu meningkatkan efisiensi, akurasi, dan transparansi pengelolaan sarana prasarana di Fakultas Ekonomi dan Bisnis Islam (FEBI) Universitas Islam Negeri K.H. Abdurrahman Wahid Pekalongan. Dengan pendekatan metode pengembangan sistem *Extreme Programming* dan penerapan teknologi Laravel dan PostgreSQL serta *service* NLP, sistem ini dapat mengotomatisasi proses peminjaman, mendeteksi bentrok jadwal secara *real-time*, serta mendukung pengelolaan berbasis peran melalui *Filament Admin Panel*. Hasil pengujian menunjukkan bahwa seluruh fungsi berjalan efektif, sehingga sistem ini layak diterapkan sebagai solusi digitalisasi manajemen peminjaman fasilitas di fakultas FEBI.



DAFTAR PUSTAKA

- Anike, M., Adoe, V. S., & Babys, I. (2023). Implementasi Extreme Programming Pada Aplikasi Pengolahan Data Inventarisasi Penyebaran Varietas Benih Tanaman Pangan. *Jurnal Media Informatika*, 4(2), 1836–1844.
- Fajri, N. Z., & Mujiastuti, R. (2024). Sistem Informasi Peminjaman Ruang dan Barang Pada Fakultas Teknik Universitas Muhammadiyah Jakarta Berbasis Website. *Jurnal Sistem Informasi, Teknologi Informatika dan Komputer*, 299–307. <https://doi.org/10.24853/justit.15.1.299-307>
- Galuh Wilujeng Saraswati & Malik Aziz Ali. (2024). Pengembangan Sistem Informasi Manajemen Pengelolaan Sampah Menggunakan Metode Extreme Programming. *Jurnal Manajemen Informatika Dan Sistem Informasi*, 7(2), 81–93. <https://doi.org/10.36595/misi.v7i2.1132>
- Hariyanto, A. R., Khodijah, S., & Aulia, W. S. (2024). Rancang Bangun Sistem Informasi Pembayaran Sumbangan Penunjang Pendidikan Berbasis Web dengan Metode Extreme Programming (XP). *Jurnal Ilmu Komputer*, 7(1), 40–52.
- Ibadurrahman, I., Yusuf, F., & Budianto, H. (2025). Implementasi Chatbot Berbasis Natural Language Processing (NLP) Menggunakan Model Bert Untuk Meningkatkan Pelayanan Klinik Gigi: Implementation of Natural Language Processing (NLP) Based Chatbot Using BERT Model to Improve Dental Clinic Services. *Rabit: Jurnal Teknologi dan Sistem Informasi Univrab*. <https://jurnal.univrab.ac.id/index.php/rabit/article/view/6061>
- Muh. Fahmi Rustan, Hasriani, & Nuralamsah Zulkarnaim. (2022). Visualisasi Data Akreditasi Program Studi Berbasis Web Menggunakan Extreme Programming. *Jurnal Informatika Teknologi Dan Sains*, 4(4), 413–420. <https://doi.org/10.51401/jinteks.v4i4.2143>
- Nurlita, A., & Munawaroh. (2025). Pengembangan Chatbot Dengan Metode Natural Language Processing untuk Layanan Pelanggan (Studi Kasus PT Masterlink Internet Solution). *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(3S1). <https://doi.org/10.23960/jitet.v13i3S1.8176>
- PostgreSQL: About. (n.d.). Retrieved November 1, 2025, from <https://www.postgresql.org/about/>
- Praharaningtyas, R., Waluyo, R., Andika, R., & Hani, N. (2024). Implementasi Metode Extreme Programming untuk Pengembangan Sistem Inventory Sarana dan Prasarana pada Universitas Amikom Purwokerto. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 7(4), 1566–1573. <https://doi.org/10.32493/jtsi.v7i4.44861>
- Purwanto, A. J., Setiawan, A., & Latifah, N. (2025). Penerapan UCD dalam Sistem Peminjaman Barang dan Ruang Dengan Laravel. *INNOVATIVE: Journal Of Social Science Research*, 5(1). <https://j-innovative.org/index.php/Innovative/article/view/18235>
- Ramadhani, R., & Fitriana, G. F. (2023). Metode eXtreme Programming (XP) dalam Pengembangan Fitur PPDB di SDN 1 Purwokerto Kulon. *Jurnal Sistem Dan Informatika (JSI)*, 18(1), 90–102. <https://doi.org/10.30864/jsi.v18i1.604>
- Sakinah, N., & Ulfa, M. (2024). Penerapan Metode Extreme Programming pada Sistem Informasi Perpustakaan Jurusan Manajemen Informatika Politeknik Negeri Fakfak. *Jurnal Informasi, Sains Dan Teknologi*, 7(2), 232–248. <https://doi.org/10.55606/isaintek.v7i2.238>
- Salza, P., Musmarra, P., & Ferrucci, F. (2019). *Agile Methodologies in Education: A Review: Bringing Methodologies from Industry to the Classroom* (pp. 25–45). https://doi.org/10.1007/978-981-13-2751-3_2
- Sankhe, P., Mathur, S., Rehman, T. B., & Dixit, M. (2022). Review of an Agile Software Development Methodology with SCRUM & Extreme Programming. *2022 IEEE International Conference on Current Development in Engineering and Technology (CCET)*, 1–6. <https://doi.org/10.1109/CCET56606.2022.10080640>



- Singla, L. (2025, January 28). *Why Laravel Framework is a Top Choice for PHP Web Development?* Net Solutions. <https://www.netsolutions.com/insights/laravel-framework-benefits/>
- Sinlae, A. A. J., Samane, I. P. A., Amaral, M. A. L., & Manehat, B. Y. (2024). Implementasi Metode Extreme Programming pada Pengembangan Sistem Informasi Bank Sampah Mutiara Timor. *Decode: Jurnal Pendidikan Teknologi Informasi*, 4(2), 359–374. <https://doi.org/10.51454/decode.v4i2.365>