

RANCANG BANGUN SISTEM PENGAJUAN SURAT PERMOHONAN KERJA PRAKTEK YANG MODULAR DENGAN METODE FEATURE DRIVEN DEVELOPMENT

Robiyatul Adawiyah Hasibuan, M. Asep Subandri

Teknik Informatika, Politeknik Negeri Bengkalis

Jalan Bathin Alam Desa Sungai Alam Kec. Bengkalis, Kab. Bengkalis, Indonesia

robiyatuladawiyahh479@gmail.com

ABSTRAK

Proses pengajuan surat permohonan kerja praktik di Politeknik Negeri Bengkalis merupakan tahapan akademik penting yang saat ini masih dikelola secara manual. Kondisi ini memunculkan permasalahan berupa keterlambatan proses birokrasi, tingginya risiko kesalahan penulisan, dan pengelolaan arsip yang tidak terstruktur. Penelitian ini bertujuan merancang dan membangun sistem digital pengajuan surat kerja praktik berbasis arsitektur modular guna meningkatkan efisiensi administrasi. Pengembangan sistem menerapkan metode Feature Driven Development (FDD) karena pendekatannya yang terstruktur dalam mengakomodasi perubahan fitur. Kualitas sistem dievaluasi secara kuantitatif menggunakan Object-Oriented Metrics dan uji dampak perubahan (change impact). Hasil penelitian menunjukkan arsitektur berhasil mencapai tingkat ketergantungan antar modul yang sangat rendah, dibuktikan dengan nilai Average Afferent Coupling (Ca) sebesar 2,33 dan Average Efferent Coupling (Ce) sebesar 2,57. Sistem juga memiliki keterikatan fungsi internal yang tinggi dengan nilai Lack of Cohesion of Methods (LCOM) sebesar 1,33. Hasil pengujian membuktikan bahwa modifikasi pada fitur tertentu tidak mengganggu modul lainnya. Dapat disimpulkan bahwa penerapan metode FDD dengan arsitektur modular sukses menghasilkan sistem yang efisien, stabil, dan mudah dipelihara di masa mendatang.

Kata kunci: *arsitektur modular, feature driven development, kerja praktik, politeknik negeri bengkalis, sistem informasi.*

1. PENDAHULUAN

Setiap mahasiswa di perguruan tinggi, termasuk di Politeknik Negeri Bengkalis (Polbeng), diwajibkan melaksanakan kegiatan kerja praktek untuk mengintegrasikan teori dan praktik di dunia kerja [1]. Namun, proses administrasi untuk tahapan pengajuan surat permohonan ke instansi tujuan saat ini masih dilakukan secara konvensional atau manual. Kondisi ini menimbulkan berbagai permasalahan, seperti keterlambatan proses birokrasi, kurangnya transparansi pelacakan status permohonan, tingginya risiko kesalahan penulisan dokumen, serta kesulitan dalam pendokumentasian arsip oleh pihak administrasi kampus [2][3]. Permasalahan tersebut memunculkan urgensi untuk merancang sebuah sistem informasi berbasis perangkat lunak yang terstruktur, efisien, dan mudah dikelola guna mengoptimalkan proses administrasi kerja praktek secara digital [4].

Sebagai rencana penyelesaian masalah, diusulkan perancangan sistem informasi menggunakan metode Feature Driven Development (FDD) yang dipadukan dengan konsep arsitektur modular. Dalam pengembangan sistem perangkat lunak, pemilihan metode yang tepat sangat memengaruhi kualitas dan keberhasilan aplikasi yang dibangun [5]. Metode konvensional seperti Waterfall dinilai terlalu kaku terhadap perubahan, sementara pendekatan Agile seperti Scrum sering kali kurang memiliki struktur dokumentasi yang kuat untuk proyek menengah hingga besar, dan Rapid Application Development (RAD) kerap mengabaikan perencanaan

desain yang matang [6]. Oleh karena itu, metode Feature Driven Development (FDD) digunakan sebagai solusi karena berfokus pada pembangunan fitur-fitur kecil secara bertahap dan sistematis, sehingga proses pengembangan menjadi lebih terukur, terarah, dan sangat adaptif terhadap perubahan kebutuhan pengguna [7] [8].

Lebih lanjut, untuk memastikan fondasi perangkat lunak yang kokoh, metode FDD sangat relevan diaplikasikan bersama konsep arsitektur modular [9]. Pendekatan modular membagi sistem ke dalam modul-modul fungsi yang spesifik sehingga setiap fitur dapat dikembangkan secara relatif mandiri [10]. Meskipun demikian, perancangan sistem berbasis fitur tanpa adanya evaluasi kualitas struktur berisiko menimbulkan tingginya ketergantungan antar modul (coupling) dan rendahnya keterkaitan fungsi di dalam satu modul (kohesion) [11]. Oleh sebab itu, rencana penyelesaian masalah dalam penelitian ini juga mencakup evaluasi kuantitatif terhadap arsitektur modular yang dirancang untuk memastikan sistem memiliki tingkat coupling yang rendah dan kemudahan dalam modifikasi [12].

Tujuan dari penelitian ini adalah merancang dan membangun sistem informasi pengajuan surat permohonan kerja praktek berbasis arsitektur modular dengan metode Feature Driven Development (FDD), serta mengevaluasi kualitas arsitektur modular sistem tersebut berdasarkan pengukuran tingkat coupling antar modul [13]. Hasil dari penelitian ini diharapkan dapat meningkatkan efisiensi proses administrasi

akademik, menghasilkan struktur perangkat lunak yang terjaga, serta mudah dikembangkan dan dipelihara di masa mendatang [14].

2. TINJAUAN PUSTAKA

2.1. Penelitian terdahulu

Penelitian terdahulu menerapkan metode Feature Driven Development (FDD) pada sistem pencari kerja berbasis website. Hasilnya menunjukkan bahwa pembagian fitur yang kecil dan terukur dalam FDD mempermudah perencanaan serta kolaborasi tim, sehingga mempercepat pengembangan dan meningkatkan kualitas sistem informasi. Pendekatan ini sangat relevan dengan penelitian penulis yang juga mengandalkan metode FDD untuk membangun sistem berbasis web yang efisien dan fungsional [2].

Studi lain mengimplementasikan metode FDD pada perancangan web portal lomba. Penelitian ini membuktikan bahwa siklus pengembangan iteratif FDD mampu menyederhanakan kompleksitas sistem dan meningkatkan fleksibilitas terhadap perubahan kebutuhan pengguna. Hal ini mendukung fokus penelitian penulis dalam menciptakan sistem informasi yang skalabel dan adaptif terhadap dinamika pengajuan surat di kampus [3].

Selanjutnya, terdapat penelitian yang membangun aplikasi marketplace produk kreatif mahasiswa berbasis mobile menggunakan metode FDD. Pendekatan ini terbukti efektif dalam mengakomodasi perubahan fitur transaksi secara bertahap tanpa merusak sistem utama. Meskipun platform yang digunakan berbeda (mobile vs web), keberhasilan iterasi fitur ini sejalan dengan tujuan penulis dalam mengembangkan fungsionalitas sistem administrasi kerja praktek secara bertahap [4].

Pengembangan sistem presensi mahasiswa berbasis web menggunakan QR Code juga telah dirancang dengan metode FDD. Pendekatan ini memastikan pengembangan berjalan terstruktur berdasarkan peran pengguna (hak akses), sehingga pengelolaan data menjadi lebih akurat. Sama seperti penelitian tersebut, penulis juga menerapkan FDD di lingkungan institusi pendidikan dengan memisahkan fungsi secara tegas antara mahasiswa, koordinator prodi, dan admin [5].

Pada lingkup administrasi magang, sebuah studi berhasil mengembangkan sistem informasi manajemen Praktik Kerja Lapangan (PKL) menggunakan metode Personal Extreme Programming (PXP). Sistem ini sukses mengotomatisasi penuh proses birokrasi manual seperti pengajuan surat magang dan laporan. Walaupun memiliki fokus penyelesaian masalah (objek) yang persis sama, penelitian penulis menawarkan kebaruan dengan menggunakan metode FDD dan arsitektur modular yang lebih menjamin kemudahan pemeliharaan kode program [6].

Penelitian lainnya menerapkan FDD pada portal pertanian Dutatani untuk menjaga konsistensi kualitas fitur antara versi web dan mobile. Hasil

evaluasi User Experience Questionnaire (UEQ) pada studi tersebut menunjukkan tingkat kepuasan yang tinggi. Kesuksesan ini memperkuat landasan bahwa metode FDD sangat andal dalam menjaga konsistensi fungsionalitas aplikasi, yang mana hal ini sangat krusial diterapkan pada sistem pengajuan kerja praktek mahasiswa [7].

Studi lain menggunakan pendekatan FDD untuk membangun sistem monitoring aliran dana desa secara transparan. Pengembangan sistem yang dilakukan secara modular dan bertahap terbukti secara signifikan meningkatkan adaptivitas terhadap dinamika kebutuhan pengguna di lapangan. Konsep pengembangan secara modular berbasis fitur inilah yang diadaptasi secara langsung oleh penulis ke dalam arsitektur sistem pengajuan surat [8].

Penelitian selanjutnya memfokuskan penggunaan FDD pada sistem informasi bimbingan konseling untuk mengatasi masalah layanan institusi yang sebelumnya masih manual dan tidak terintegrasi. Pengujian usability yang mencapai angka 90% menegaskan bahwa sistem berbasis fitur ini sangat mudah digunakan dan dikelola. Temuan ini memberikan landasan kuat bagi penulis bahwa FDD mampu menyelesaikan kerumitan alur birokrasi administrasi akademik mahasiswa [9].

Pengembangan sistem pengelolaan inventaris juga telah dilakukan menggunakan metode Agile Feature Driven Development. Hasil penelitian tersebut menonjolkan adanya peningkatan akurasi data serta kecepatan waktu perilisasi aplikasi karena alur kerjanya yang sangat terstruktur. Kesamaan dengan penelitian penulis terletak pada pemanfaatan FDD untuk mengelola database internal organisasi (kampus) secara lebih transparan dan efisien [10].

Terakhir, sebuah penelitian merancang bangun aplikasi layanan restoran menggunakan metode FDD. Evaluasi sistem menunjukkan fungsionalitas yang sangat tinggi karena fitur-fitur kritis dikembangkan secara terpisah dan bertahap untuk meminimalkan risiko kegagalan. Struktur pengerjaan iteratif ini sangat selaras dengan konsep perancangan arsitektur modular yang digunakan penulis untuk meminimalisir change impact (error berantai) pada sistem pengajuan surat magang [11].

2.2. Pengembangan Sistem Administrasi Kerja Praktek

Pengembangan perangkat lunak merupakan serangkaian proses terstruktur, mulai dari analisis kebutuhan, perancangan, implementasi, hingga pemeliharaan sistem, yang bertujuan untuk memecahkan masalah pengguna secara efektif dan efisien. Dalam lingkungan akademik, sistem ini diimplementasikan untuk mendigitalisasi tahapan administrasi Kerja Praktek (KP), yakni kegiatan wajib yang mengintegrasikan teori akademik dengan praktik di dunia profesional. Sistem difokuskan untuk mengelola Surat Permohonan Kerja Praktek, sebuah

dokumen resmi yang memuat identitas mahasiswa, tujuan instansi, dan periode pelaksanaan magang [12].

2.3. Arsitektur Perangkat Lunak Modular

Arsitektur modular merupakan pendekatan perancangan perangkat lunak yang membagi sistem ke dalam beberapa modul fungsional yang beroperasi secara independen. Setiap modul dirancang untuk memiliki satu tanggung jawab spesifik (single responsibility) [13]. Keunggulan utama pendekatan ini adalah membatasi tingkat dampak perubahan (change impact); di mana modifikasi, penambahan, atau perbaikan pada satu modul tidak akan memberikan dampak kerusakan sistemik terhadap modul lain di sekitarnya [14]. Konsep ini sangat vital untuk menjamin kemudahan pemeliharaan (maintainability) aplikasi berskala menengah hingga besar [15].

2.4. Feature Driven Development (FDD)

Pendekatan modular dalam penelitian ini dikembangkan menggunakan metodologi Feature Driven Development (FDD). FDD adalah metode pengembangan berorientasi objek yang membagi kompleksitas sistem ke dalam fitur-fitur kecil dan terukur. Siklus pengembangan FDD terdiri dari lima tahapan sistematis: 1) Develop an Overall Model (membangun pemodelan interaksi awal); 2) Build a Feature List (menyusun daftar fitur berskala kecil yang bernilai); 3) Plan by Feature (perencanaan jadwal dan prioritas iterasi); 4) Design by Feature (perancangan alur teknis fungsi); dan 5) Build by Feature (implementasi kode dan pengujian fitur) [16].

2.5. Evaluasi Metrik Kualitas Perangkat Lunak

Keberhasilan desain arsitektur modular tidak hanya diuji dari fungsinya, tetapi juga dievaluasi secara kuantitatif terhadap kualitas internal kode sumbernya menggunakan Object-Oriented Metrics [17]. Evaluasi ini bertumpu pada dua indikator metrik utama:

- a. Coupling Between Object Classes (CBO): Mengukur seberapa besar tingkat ketergantungan antara satu class dengan class lainnya. Nilai CBO yang optimal berada pada rentang 0 hingga 5. Semakin rendah nilai CBO (low coupling), semakin modular dan mudah sistem tersebut untuk dipelihara karena tidak rentan terhadap error berantai [18].
- b. Lack of Cohesion of Methods (LCOM): Mengukur seberapa kuat keterikatan internal fungsionalitas di dalam satu buah class. Sistem yang terstruktur dengan baik harus memenuhi nilai LCOM ≤ 3 (mendekati 0). Nilai high cohesion ini menandakan bahwa method-method di dalam sebuah class bekerja sangat terfokus tanpa adanya penumpukan beban fungsi yang tidak relevan [19].

2.6. Pemodelan dan Teknologi Implementasi

Pemodelan dan Teknologi Implementasi Dalam perancangan sistem, alur logika dan interaksi pengguna dipetakan menggunakan Unified Modelling Language (UML). Pemodelan ini mencakup Use Case Diagram untuk mendefinisikan batasan hak akses aktor, serta Activity dan Sequence Diagram untuk menjabarkan rincian aktivitas sistem. Selanjutnya, dari segi implementasi, sistem dibangun menggunakan editor Visual Studio Code dengan bahasa pemrograman PHP Native. Pendekatan native digunakan agar pengembang memiliki kontrol arsitektur secara penuh tanpa ketergantungan pada framework eksternal yang berat. Sistem ini didukung oleh basis data relasional MySQL berbasis bahasa kueri Structured Query Language (SQL), yang dikonfigurasi melalui paket peladen web lokal XAMPP [20].

3. METODE PENELITIAN

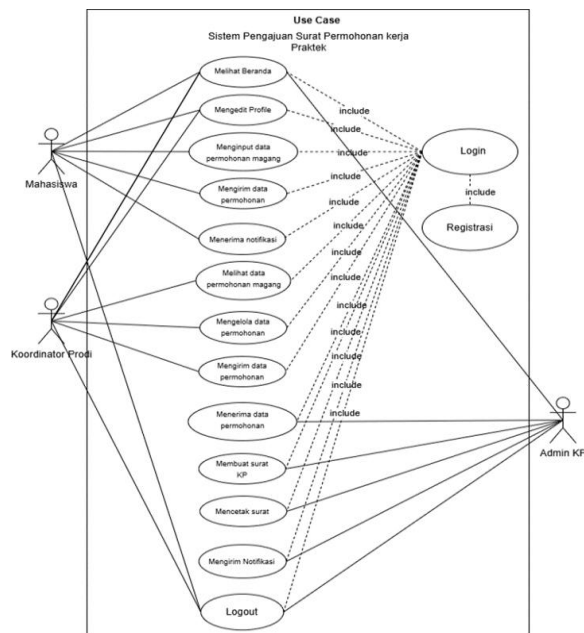
3.1. Formulasi Masalah

Proses administrasi pengajuan surat permohonan kerja praktek di Politeknik Negeri Bengkalis saat ini masih dilakukan secara manual. Permasalahan ini berdampak langsung pada keterlambatan penyelesaian proses, kurangnya transparansi status pengajuan, kesalahan penulisan dokumen, serta menyulitkan proses pengarsipan oleh pihak kampus. Untuk mengatasi permasalahan tersebut, diusulkan rancang bangun sistem informasi digital berbasis perangkat lunak dengan metode Feature Driven Development (FDD) yang dipadukan dengan arsitektur modular. Pendekatan ini diformulasikan untuk membangun sistem dengan fitur-fitur yang independen sehingga meminimalisir ketergantungan antar modul (coupling) dan meningkatkan keterkaitan fungsi di dalamnya (cohesion).

3.2. Metode Pengusulan: Feature Driven Development (FDD)

Pengembangan sistem diusulkan dengan menerapkan tahapan iteratif dan sistematis dari metode FDD yang berfokus pada pembangunan fitur fungsional berskala kecil:

- a. Develop an Overall Model: Memodelkan interaksi sistem secara keseluruhan menggunakan Use Case Diagram yang mendefinisikan batas hak akses tiga aktor utama: Mahasiswa, Koordinator Program Studi, dan Admin Kerja Praktek.
- b. Build a Feature List: Mengidentifikasi daftar fitur secara spesifik berdasarkan kebutuhan aktor, mulai dari tahap registrasi akun, dashboard, pengajuan permohonan, verifikasi, hingga fitur pencetakan otomatis dan notifikasi status.



Gambar 1. Usa Case Diagram

Tabel 1. Daftar Fitur Utama Sistem

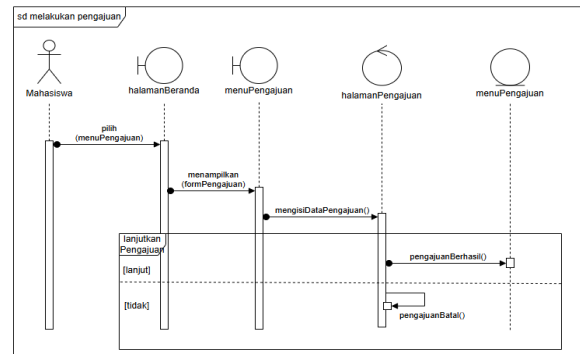
No	Fitur Utama	Aktor Terlibat	Keterangan Aksi
1	Registrasi & Login	Semua Aktor	Pendaftaran akun dan proses autentikasi masuk sistem.
2	Pengajuan	Mahasiswa	Mengisi formulir dan mengirim data permohonan magang.
3	Verifikasi	Koordinator Prodi	Mengelola, memvalidasi, dan meneruskan data permohonan.
4	Penerbitan Surat	Admin KP	Membuat, menyortir, dan mencetak surat permohonan otomatis.
5	Notifikasi	Mahasiswa & Admin KP	Mengirim dan menerima pembaruan status pengajuan surat.

- c. Plan by Feature: Menjadwalkan alur pengembangan fitur berdasarkan skala prioritas agar proses iterasi berjalan terstruktur.

Tabel 2. Rencana Iterasi Pengerjaan Fitur

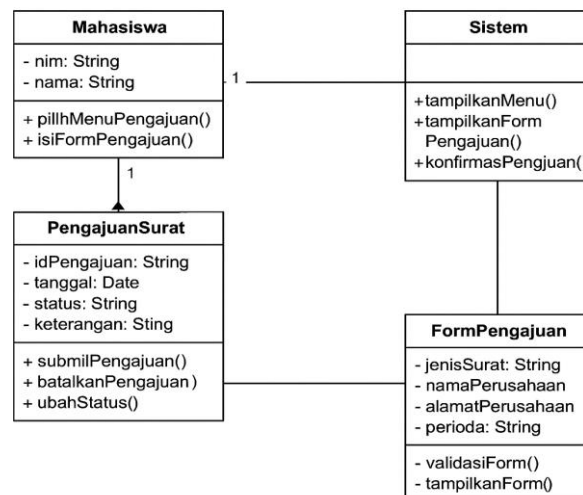
Tahap Iterasi	Fitur yang Dikembangkan	Aktor Target
Iterasi 1	Registrasi, Login, Logout	Mahasiswa, Koordinator Prodi, Admin KP
Iterasi 2	Dashboard, Pengajuan	Mahasiswa, Koordinator Prodi, Admin KP
Iterasi 3	Kelola Permohonan, Cetak Surat	Koordinator Prodi, Admin KP
Iterasi 4 & 5	Notifikasi, Manajemen Profil	Semua Aktor

- d. Design by Feature: Merancang spesifikasi teknis setiap fitur ke dalam Sequence Diagram dan Activity Diagram guna memetakan interaksi objek dan alur kerja aplikasi.



Gambar 2. Sequence Diagram Alur Pengajuan oleh Mahasiswa

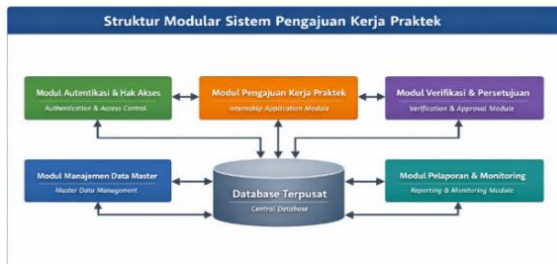
- e. Build by Feature: Tahap implementasi penulisan kode program menggunakan bahasa PHP Native dengan memecah fungsi ke dalam arsitektur Model-View-Controller (MVC) untuk perancangan basis data dan tampilan antar muka.



Gambar 3. Class Diagram Pengajuan Permohonan Surat Kerja Praktek

3.3. Perancangan Arsitektur Modular Monolith

Untuk menjamin pemeliharaan perangkat lunak jangka panjang, sistem dirancang menggunakan arsitektur Modular Monolith. Dalam arsitektur ini, keseluruhan sistem disatukan dalam satu aplikasi tunggal namun terstruktur secara logis ke dalam beberapa modul mandiri (seperti Modul Autentikasi, Modul Pengajuan, dan Modul Verifikasi). Tahapan implementasi modular meliputi Module Identification, Module Structure Design, Module Development, dan Module Integration. Pemisahan ini membatasi change impact (dampak perubahan) yang mana modifikasi pada satu modul tidak akan menyebabkan sistem error atau merusak modul lain di sekitarnya.



Gambar 4. Rancangan Arsitektur Modular Monolith

3.4. Evaluasi Kualitas Modular (Pengujian Metrik Sistem)

Setelah sistem terbangun, metode penelitian ini mengevaluasi state of the art dari rancangan arsitekturnya secara kuantitatif melalui Object-Oriented Metrics. Evaluasi kualitas struktur ini dirumuskan menggunakan dua variabel metrik utama:

- Coupling Between Object Classes (CBO):** Digunakan untuk mengukur tingkat interdependensi atau ketergantungan antar kelas. Dalam evaluasi ini, arsitektur dikategorikan baik jika nilai CBO diformulasikan dalam rentang $0 \leq CBO \leq 5$ (maksimal dapat ditoleransi hingga 10). Pengukuran coupling ini dibagi lagi menjadi dua pengujian, yakni pengukuran ketergantungan masuk (Average Afferent Coupling / Ca) dan ketergantungan keluar (Average Efferent Coupling / Ce).
- Lack of Cohesion of Methods (LCOM):** Digunakan untuk mengukur kekuatan kohesi (keterikatan fungsi) di dalam satu kelas. Secara matematis, sistem modular yang optimal harus memenuhi nilai $LCOM \leq 3$. Nilai yang mendekati 0 menunjukkan bahwa method-method saling berkaitan erat dan tidak melanggar prinsip single responsibility.

Tabel 3. Hasil Pengujian Coupling dan Cohesion

Modul Sistem (Versi Modular)	Tingkat Coupling	Tingkat Cohesion	Change Impact
PengajuanController	Rendah	Tinggi	Kecil
PengajuanService	Rendah	Tinggi	Kecil
SuratService	Rendah	Tinggi	Kecil
NotifikasiService	Rendah	Tinggi	Kecil

4. HASIL DAN PEMBAHASAN

Bab ini menguraikan hasil implementasi aplikasi pengajuan surat permohonan kerja praktek berbasis arsitektur modular yang dikembangkan menggunakan metode Feature Driven Development (FDD), dilanjutkan dengan pengujian kualitas struktur modul, serta pembahasan menyeluruh terhadap hasil penelitian.

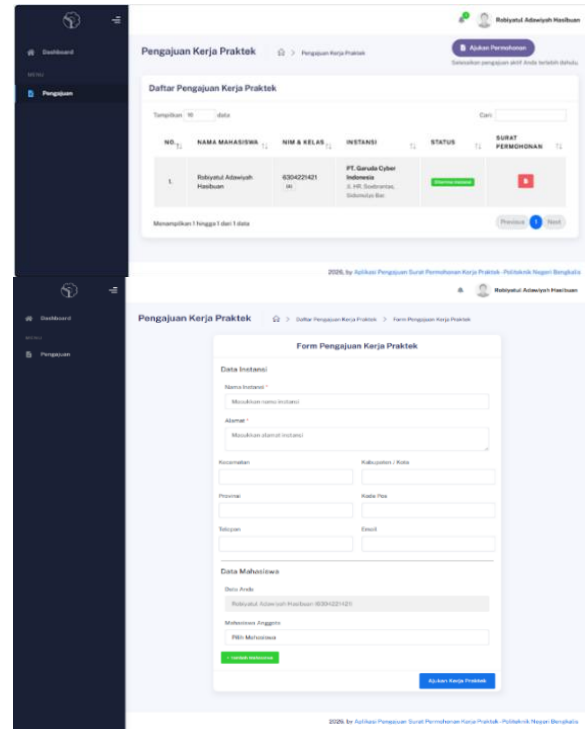
4.1. Hasil Implementasi Sistem

Sistem informasi ini dibangun menggunakan bahasa pemrograman PHP Native dengan arsitektur Modular Monolith. Implementasi antar muka (user interface) dan fungsi sistem dibagi berdasarkan tiga

hak akses aktor utama: Mahasiswa, Koordinator Program Studi, dan Admin Kerja Praktek.

4.2. Antarmuka Mahasiswa

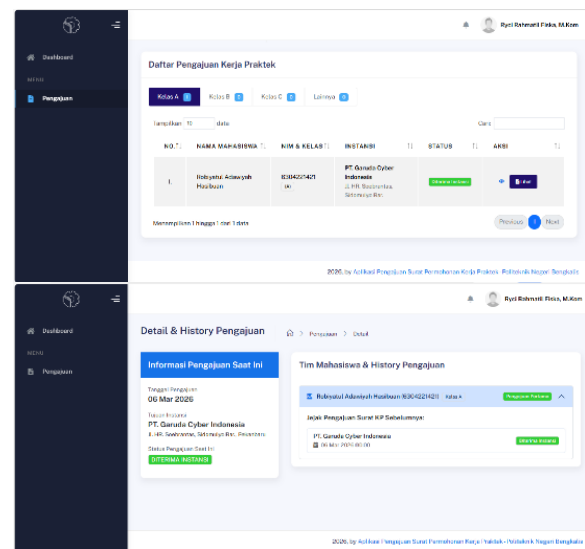
Terdiri dari fitur registrasi, login, dashboard interaktif, pengisian formulir permohonan magang, serta pelacakan status pengajuan. Melalui modul ini, mahasiswa tidak perlu lagi menyerahkan berkas fisik.



Gambar 5. Daftar Pengajuan Dan Pengisian Formulir Permohonan Magang

4.3. Antarmuka Koordinator Prodi

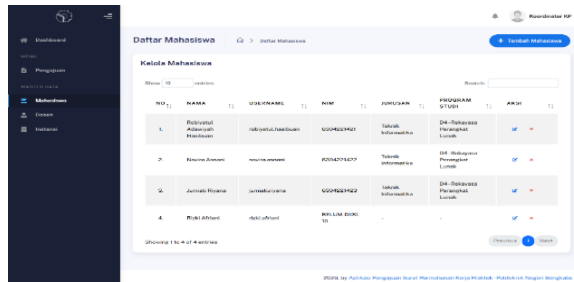
Menyediakan fitur untuk memverifikasi, menyetujui, atau menolak permohonan magang yang diajukan oleh mahasiswa berdasarkan kesesuaian data dan kuota instansi tujuan.



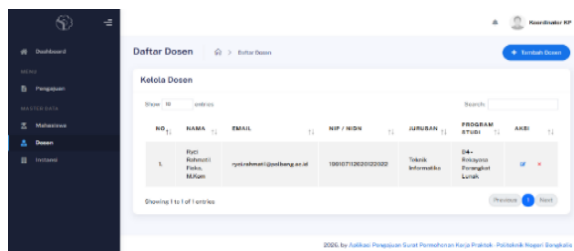
Gambar 6. Fitur Verifikasi Untuk Koordinator Prodi

4.4. Antarmuka Admin KP

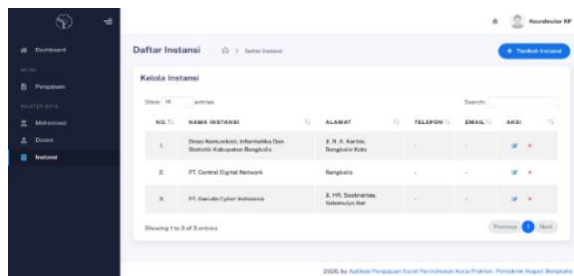
Memfasilitasi admin untuk mengelola master data (dosen, mahasiswa, instansi) serta mencetak surat permohonan resmi dalam format PDF yang di-generate secara otomatis oleh sistem.



Gambar 9. Admin Mengelola Mahasiswa



Gambar 10. Admin Mengelola Dosen



Gambar 11. Admin Mengelola Instansi

4.5. Hasil Pengujian dan Evaluasi Arsitektur Modular

Setelah sistem selesai diimplementasikan melalui tahapan Build by Feature, dilakukan pengujian dan evaluasi terhadap kualitas struktur modular perangkat lunak. Pengujian dibagi menjadi dua tahap, yaitu pengujian fungsionalitas struktur (change impact) dan pengujian metrik kuantitatif.

- Pengujian Dampak Perubahan (Change Impact)**
Pengujian ini dilakukan untuk melihat apakah arsitektur sistem benar-benar modular secara praktis. Uji coba dilakukan dengan menghapus atau mengubah salah satu fitur atau tampilan (misalnya menghapus form field nomor telepon dan email pada controller pengajuan). Hasil pengujian membuktikan bahwa perubahan atau penghapusan elemen pada satu modul sama sekali tidak merusak fungsi atau mengganggu proses berjalan pada halaman maupun modul lain di dalam sistem. Hal ini mengonfirmasi sifat independensi modul (loose coupling) telah berhasil dicapai.
- Pengujian Kualitas Modular dengan Object-Oriented Metrics Evaluasi**
ini mengukur tingkat modularitas sistem secara kuantitatif melalui analisis statis kode sumber menggunakan alat bantu PHP Metrics, yang difokuskan pada nilai Coupling dan Cohesion. Secara umum, dari hasil scanning, sistem ini terbangun dari 1085 baris kode (Lines of Code), 21 class, dan 72 method, dengan rata-rata 3,43 method per class. Pembagian method yang tidak berlebihan ini merupakan indikasi awal pengorganisasian kode yang proporsional.

Rincian hasil evaluasi metrik disajikan pada Tabel 4 berikut:

Tabel 4. Hasil Evaluasi Metrik Modularitas Sistem

Aspek Evaluasi	Metrik Pengukuran	Nilai Hasil	Interpretasi
Struktur Kode	Jumlah Class	21 class	Sistem memiliki pembagian class yang cukup terorganisir.
Struktur Kode	Jumlah Method	72 method	Setiap class memiliki jumlah method yang wajar.
Cohesion	Lack of Cohesion of Methods (LCOM)	1,33	Nilai rendah (≤ 3), menunjukkan kohesi internal fungsi class yang sangat baik.
Coupling	Average Afferent Coupling (Ca)	2,33	Ketergantungan masuk (incoming dependencies) antar class sangat rendah.
Coupling	Average Efferent Coupling (Ce)	2,57	Ketergantungan keluar (outgoing dependencies) antar class sangat rendah.

Complexity	
Average Cyclomatic complexity by class	6.57
Average Weighted method count by class	9
Average Relative system complexity	137.72
Average Difficulty	5.21
Bugs	
Average bugs by class	0.18
Average defects by class (Kan)	0.38
Violations	
Critical	0
Error	3
Warning	4
Information	0
PS C:\XAMPP2\htdocs\pengajuan-surat-kp-native> vendor/bin/phpmetrics app/	
Executing system analyzes...	
Executing composer analyzes, requesting https://packagist.org...	
LOC	
Lines of code	1085
Logical lines of code	943
Comment lines of code	142
Average volume	536.19
Average comment weight	22.58
Average intelligent content	22.58
Logical lines of code by class	45
Logical lines of code by method	13
Object oriented programming	
Classes	21
Interface	0
Methods by class	3.43
Lack of cohesion of methods	1.33
Coupling	
Average afferent coupling	2.33
Average efferent coupling	2.57
Average instability	0.62
Depth of Inheritance Tree	1.4
Package	
Packages	1
Average classes per package	21
Average distance	0
Average incoming class dependencies	0
Average outgoing class dependencies	1
Average incoming package dependencies	0
Average outgoing package dependencies	1

Gambar 13. Hasil Eksekusi Evaluasi PHP Metrics pada Kode Sumber

4.6. Pembahasan

Pembahasan dari penelitian ini difokuskan pada analisis keberhasilan penerapan metode Feature Driven Development (FDD) dan tingkat efektivitas arsitektur modular yang dihasilkan berdasarkan data empiris pada proses evaluasi. Pertama, penerapan metode FDD terbukti sangat efektif dalam menjembatani kebutuhan administrasi akademik yang dinamis. Pendekatan FDD yang berpusat pada fitur (feature-centric) memastikan bahwa fungsi-fungsi kritis seperti manajemen persetujuan (verifikasi berjenjang) dan generate dokumen PDF secara otomatis dapat dirancang, dibangun, dan diuji secara terpisah tanpa harus menunggu keseluruhan sistem selesai. Hal ini tidak hanya mempercepat waktu perilisan (delivery time), tetapi juga meminimalisir kegagalan sistem secara sistemik.

Kedua, berdasarkan evaluasi metrik pada Tabel 4, arsitektur sistem menunjukkan pencapaian kualitas struktur perangkat lunak yang unggul. Metrik pengujian Coupling yang diwakili oleh nilai Average Afferent Coupling (Ca) sebesar 2,33 dan Average Efferent Coupling (Ce) sebesar 2,57 membuktikan bahwa tingkat ketergantungan antar komponen sistem berada pada level yang sangat rendah (ideal berada di

rentang 0-5),. Kondisi Low Coupling ini menjelaskan mengapa saat dilakukan uji penghapusan fitur pada skenario change impact, modul lainnya tetap stabil. Keterisolasian modul ini membuat aplikasi jauh lebih aman dari bug berantai. Ketiga, dari sisi kepaduan tugas internal, metrik Lack of Cohesion of Methods (LCOM) mencatatkan nilai 1,33. Mengingat bahwa nilai LCOM yang mendekati 0 atau ≤ 3 adalah standar emas dalam Object-Oriented Programming,, hasil ini mengindikasikan fenomena High Cohesion. Artinya, method-method di dalam setiap class beroperasi secara sangat terfokus untuk menangani satu tujuan spesifik (sejalan dengan prinsip Single Responsibility), tanpa adanya pencampuran logika bisnis yang tidak relevan di dalam satu modul yang sama,.

Secara keseluruhan, integrasi antara metode FDD dan arsitektur Modular Monolith sukses menghasilkan aplikasi pengajuan surat magang yang tidak hanya menyelesaikan permasalahan administrasi manual di kampus, namun juga menyediakan fondasi perangkat lunak (kualitas backend) yang sangat maintainable (mudah dirawat) dan terukur untuk skalabilitas penambahan fitur di kemudian hari.

5. KESIMPULAN DAN SARAN

Penerapan arsitektur modular yang dipadukan dengan metode Feature Driven Development (FDD) telah berhasil mengoptimalkan pengembangan aplikasi pengajuan surat permohonan kerja praktek di Politeknik Negeri Bengkalis. Pendekatan metode FDD terbukti memberikan fleksibilitas yang tinggi dalam mengakomodasi dinamika perubahan kebutuhan pengguna melalui pengelolaan tahapan sistem yang terstruktur. Lebih lanjut, evaluasi kualitas arsitektur sistem menunjukkan keberhasilan perancangan dalam mencapai standar modularitas yang ideal. Hal ini dibuktikan dengan sangat rendahnya tingkat ketergantungan antar modul (coupling) serta tingginya kohesi internal di dalam kelas-kelas sistem yang dibangun. Tercapainya kualitas struktur perangkat lunak yang solid ini sangat krusial karena memberikan jaminan kemudahan dalam pemeliharaan sistem, menjaga skalabilitas aplikasi, serta mampu meminimalkan dampak atau risiko gangguan (error) saat dilakukan modifikasi maupun penambahan fitur baru di masa mendatang.

DAFTAR PUSTAKA

- [1] Z. A. Alzamil, "Software Coupling And Cohesion Model For Measuring The Quality Of Software Components," *Computers, Materials And Continua*, Vol. 77, No. 3, Pp. 3139–3161, 2023, Doi: 10.32604/Cmc.2023.042711.
- [2] V. D. R. F. G. Kusnadi; Kartika, "Penerapan Metode Feature Driven Development Pada Sistem Pencari Kerja Berbasis Website," 2024.
- [3] P. Tamir, "Implementasi Metode Feature Driven Development Pada Perancangan Web Portal Lomba," 2023, Doi: 10.3785/Kjst.V1i6.347.

- [4] A. Valerian Romero And R. Fahrudin, "Membangun Marketplace Untuk Penjualan Produk Kreatif Mahasiswa Berbasis Mobile Menggunakan Metode Fdd," 2023.
- [5] M. Setiawan, W. Hadi, And A. Surakarta, "Penerapan Metode Feature Driven Development (Fdd) Dalam Perancangan Sistem Presensi Mahasiswa Berbasis Web," 2022.
- [6] T. Rekayasa Keamanan Siber And P. Negeri Batam, "Sistem Informasi Manajemen Praktek Kerja Lapangan Berbasis Website Studi Kasus Smk Ma'arif Kota Batam," 2023. [Online]. Available: [Http://jurnal.Polibatam.Ac.Id/Index.Php/Jamn](http://jurnal.polibatam.ac.id/index.php/jamn)
- [7] A. R. W. A. C. L. R. M. N. A. Chrismanto, "Jepin (Jurnal Edukasi Dan Penelitian Informatika)," 2022, [Online]. Available: [https://Dutatani.Id](https://dutatani.id).
- [8] A. M. N. D. S. W. N. N. Tanniewa, "Pengembangan Sistem Informasi Monitoring Dana Desa Menggunakan Pendekatan Feature Driven Development," 2024.
- [9] R. R. Al-Hakim, Y. Yanuardi, R. J. Rumandan, And M. Tonggiroh, "Pengembangan Sistem Informasi Bimbingan Konseling Menggunakan Metode Feature-Driven Development," Jurnal Ilmiah Fifo, Vol. 16, No. 2, P. 193, Nov. 2024, Doi: 10.22441/Fifo.2024.V16i2.009.
- [10] E. K. A. S. T. R. L. Wahyudi, "Pengembangan Sistem Informasi Pengelolaan Inventaris Dengan Metode Agile Feature Driven Development," 2022.
- [11] A. R. D. B. E. S. Fajarudin, "Rancang Bangun Aplikasi Restoran Menggunakan Metode Feature Driven Development," 2024.
- [12] D. Ir. M.Kom. Susilo, Konsep Dasar Rekayasa Perangkat Lunak. Kota Batam, 29444: Yayasan Cendikia Mulia Mandiri, 2023. Accessed: Aug. 11, 2025. [Online]. Available: [https://Www.Google.Co.Id/Books/Edition/Konsep_Dasar_Rekayasa_Perangkat_Lunak/7om6eaaqbaj?Hl=Id&Gbpv=1&Dq=Pressman%2c%20r.%20s.%20\(2014\).%20software%20engineerin g%3a%20a%20practitioner's%20approach%20\(8th%20ed.\).%20mcgrawHill%20education.&Pg =Pa7&Printsec=Frontcover](https://www.google.co.id/books/edition/Konsep_Dasar_Rekayasa_Perangkat_Lunak/7om6eaaqbaj?hl=id&gbpv=1&dq=Pressman%2c%20r.%20s.%20(2014).%20software%20engineering%3a%20a%20practitioner's%20approach%20(8th%20ed.).%20mcgrawHill%20education.&pg=Pa7&printsec=Frontcover)
- [13] A. Y. Salim And F. A. Alijoyo, "Rancang Bangun Sistem Informasi Manajemen Layanan Magang Di Diskominfo Kabupaten Purwakarta Berbasis Web Dengan Menggunakan Metode Extreme Programming," Jipi (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika), Vol. 9, No. 1, Pp. 284–294, Feb. 2024, Doi: 10.29100/Jipi.V9i1.4434.
- [14] F. Chusna, "10 Contoh Surat Permohonan Magang Dan Cara Membuatnya," Pt Privy Identitas Digital. Accessed: May 03, 2025. [Online]. Available: <https://Privy.Id/Blog/Contoh-Surat-Permohonan-Magang/#:~:Text=Ber%20tanda%20tangan,Ap a%20itu%20surat%20permohonan%20magang? ,Memiliki%20jangka%20waktu%20yang%20ter batas.>
- [15] D. Kumala Hayati Et Al., "Penerapan Konsep Modular Dengan Gaya Industrial Di Galeri Kopi Ekowisata Manglayang Bandung," Vol. 9, No. 1, 2024, Doi: 10.37715/Aksen.V9i1.4477.
- [16] D. Sofronas, D. Margounakis, M. Rigou, E. Tambouris, And T. Pachidis, "Sqmetrics: An Educational Software Quality Assessment Tool For Java," Knowledge, Vol. 3, No. 4, Pp. 557–599, Sep. 2023, Doi: 10.3390/Knowledge3040036.
- [17] Z. A. Alzamil, "Software Coupling And Cohesion Model For Measuring The Quality Of Software Components," Computers, Materials And Continua, Vol. 77, No. 3, Pp. 3139–3161, 2023, Doi: 10.32604/Cmc.2023.042711.
- [18] E. N. Haner Kırğıl And T. Erçelebi Ayyıldız, "Predicting Software Cohesion Metrics With Machine Learning Techniques," Applied Sciences (Switzerland), Vol. 13, No. 6, Mar. 2023, Doi: 10.3390/App13063722.
- [19] S. Jin Et Al., "Software Code Quality Measurement: Implications From Metric Distributions," Jan. 2024, Doi: 10.1109/Qrs60937.2023.00054.
- [20] T. G. S. Filó, M. A. S. Bigonha, And K. A. M. Ferreira, "Evaluating Thresholds For Object-Oriented Software Metrics," Journal Of The Brazilian Computer Society, Vol. 30, No. 1, Pp. 313–346, Mar. 2024, Doi: 10.5753/Jbcs.2024.3373.