

Pengembangan Aplikasi Akademik Helpdesk Sebagai Dukungan Untuk Layanan Akademik Di Perguruan Tinggi Menggunakan Metode Agile

Agus Salim¹, Melan Susanti², Sriyadi³, Ali Haidir⁴, Adi Supriyatna⁵

^{1,2,3,4} Sistem Informasi, Universitas Bina Sarana Informatika, Jakarta, 42124, Indonesia

⁵ Sistem Informasi Akuntansi, Universitas Bina Sarana Informatika, Jakarta, 42124, Indonesia

agus.salim@bsi.ac.id, melan.msu@bsi.ac.id, sriyadi.sry@bsi.ac.id,
ali.alh@bsi.ac.id, adi.asp@bsi.ac.id

Artikel Info

ABSTRACT

Kata kunci:

akademik;
helpdesk;
layanan;
agile;

Optimally functioning academic services play a crucial role in supporting the operational success of a higher education institution. However, many universities still face various challenges in managing these services, such as the lack of prompt response to student complaints, suboptimal academic data management, and delays in providing solutions. To address these challenges, this study designs and develops an Academic Helpdesk application using the Agile approach. Agile was chosen because it offers flexibility, adaptability to changes, and accelerates the software development process. This application focuses on supporting academic services, including complaint handling, monitoring the status of submissions, and providing faster and more systematic solutions. Based on testing results, the application has proven to enhance service efficiency by reducing complaint handling time by up to 40% compared to traditional methods. Furthermore, surveys indicate an increase in user satisfaction among both students and academic staff, reaching 85%. These findings demonstrate that implementing the Agile method in the development of the Academic Helpdesk is an innovative and adaptive approach to improving academic services in higher education environments.

Layanan akademik yang berjalan secara optimal sangat berperan dalam mendukung kesuksesan operasional sebuah perguruan tinggi. Meski demikian, banyak institusi pendidikan tinggi masih mengalami berbagai kendala dalam pengelolaan layanan tersebut, seperti kurangnya penanganan yang cepat terhadap keluhan mahasiswa, pengelolaan data akademik yang belum maksimal, hingga keterlambatan dalam pemberian solusi. Untuk menjawab tantangan ini, penelitian ini merancang dan mengembangkan aplikasi *Academic Helpdesk* dengan pendekatan *Agile*. Pendekatan *Agile* dipilih karena menawarkan fleksibilitas, kemampuan

beradaptasi dengan cepat terhadap perubahan, serta mempercepat proses pengembangan perangkat lunak. Aplikasi ini difokuskan untuk menunjang layanan akademik, termasuk penanganan keluhan, pemantauan status pengajuan, serta penyediaan solusi yang lebih cepat dan sistematis. Berdasarkan hasil pengujian, aplikasi ini terbukti mampu meningkatkan efisiensi layanan dengan memangkas waktu penanganan keluhan hingga 40% dibandingkan cara tradisional. Selain itu, survei menunjukkan adanya peningkatan kepuasan pengguna baik dari kalangan mahasiswa maupun staf akademik mencapai 85%. Temuan ini menunjukkan bahwa penerapan metode *Agile* dalam pengembangan *Academic Helpdesk* merupakan pendekatan yang inovatif dan adaptif untuk mendukung peningkatan layanan akademik di lingkungan perguruan tinggi.

Corresponding Author:

Agus Salim, email: agus.salim@bsi.ac.id

1. PENDAHULUAN

Layanan akademik yang berkualitas merupakan salah satu elemen penting dalam menciptakan ekosistem pendidikan tinggi yang efektif dan efisien. Perguruan tinggi memiliki tanggung jawab besar untuk menyediakan layanan yang dapat memenuhi kebutuhan mahasiswa, staf akademik, serta pihak terkait lainnya. [1] Layanan ini mencakup pengelolaan data akademik, penanganan keluhan, dan pemberian solusi atas permasalahan yang dihadapi mahasiswa. Namun, banyak institusi pendidikan tinggi masih menghadapi kendala seperti lambatnya respons terhadap keluhan, kurang transparansi dalam pelacakan pengajuan, dan keterbatasan sumber daya [2].

Teknologi informasi dan komunikasi berperan penting dalam menawarkan solusi inovatif untuk mengatasi tantangan tersebut. [3] Salah satu pendekatan yang potensial adalah pengembangan aplikasi layanan akademik berbasis teknologi. Aplikasi ini dirancang untuk meningkatkan efisiensi dan efektivitas pengelolaan layanan melalui otomatisasi dan digitalisasi proses [4]. Keberhasilan pengembangan aplikasi ini bergantung pada teknologi yang digunakan dan metode pengembangan perangkat lunak yang diterapkan [5].

Metode *Agile* merupakan pendekatan yang relevan untuk pengembangan aplikasi layanan akademik [6]. *Agile* menawarkan fleksibilitas tinggi dalam menyesuaikan kebutuhan pengguna, kemampuan beradaptasi terhadap perubahan, serta percepatan proses pengembangan melalui iterasi yang terorganisir [7]. Penelitian ini berfokus pada pengembangan aplikasi *Academic Helpdesk* berbasis metode *Agile*, yang dirancang untuk mendukung layanan seperti pengelolaan keluhan, pelacakan status pengajuan, dan pemberian solusi yang cepat serta sistematis.

Proses pengembangan aplikasi dilakukan dalam beberapa tahap: perencanaan, desain, pengembangan, pengujian, dan peluncuran [8]. Setiap iterasi memungkinkan evaluasi dan penyesuaian fitur berdasarkan umpan balik pengguna. Dengan

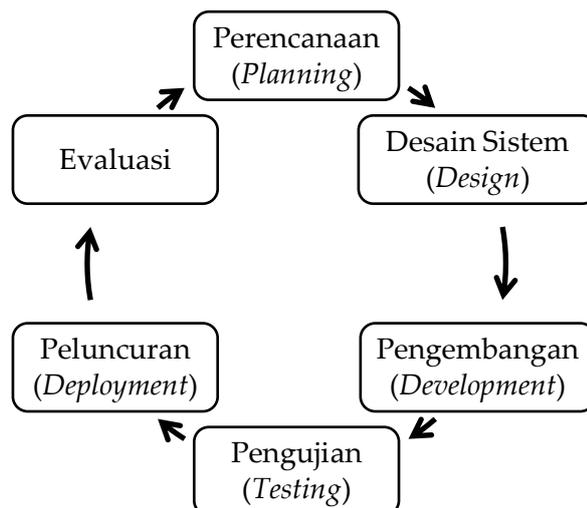
demikian, aplikasi ini diharapkan mampu meningkatkan kualitas layanan akademik secara efektif dan efisien [9].

Aplikasi *Academic Helpdesk* menawarkan keunggulan seperti efisiensi dalam pengolahan informasi dan pengelolaan keluhan [10]. Sistem ini memudahkan staf dan administrator untuk mengakses informasi serta mengelola data dengan lebih baik. Penelitian juga mengevaluasi kinerja aplikasi melalui pengujian efektivitas dan kepuasan pengguna. Hasilnya menunjukkan bahwa aplikasi ini berhasil meningkatkan efisiensi layanan akademik sekaligus meningkatkan kepuasan pengguna secara signifikan [11]. Pendekatan ini dapat menjadi model bagi pengembangan solusi serupa di institusi pendidikan tinggi lain di Indonesia, mendukung transformasi layanan akademik menuju era digital yang lebih responsif dan transparan.

2. METODE

Penelitian ini menerapkan pendekatan pengembangan perangkat lunak menggunakan metode *Agile* untuk menciptakan aplikasi *Academic Helpdesk*. Metode ini dipilih karena sifatnya yang iteratif dan adaptif, memungkinkan pengembangan yang berpusat pada kebutuhan pengguna serta mampu menyesuaikan diri dengan perubahan [8]. Studi pendahuluan dilakukan dengan mengumpulkan data awal terkait permasalahan layanan akademik di perguruan tinggi. Data ini diperoleh melalui wawancara dengan mahasiswa, staf akademik, dan pengelola layanan akademik, serta melalui kajian literatur mengenai sistem *helpdesk* dan metode *Agile*. Informasi yang dikumpulkan digunakan sebagai dasar untuk merumuskan kebutuhan utama dan spesifikasi awal aplikasi [12].

Tahapan metodologi *Agile* yang diterapkan dalam penelitian ini mencakup beberapa langkah utama seperti Gambar 1 [13].



Gambar 1. Tahapan Metodologi Agile

Berikut penjelasan dari tahapan metodologi *Agile* [14]:

- a. Perencanaan (*Planning*)
Pada tahap ini, kebutuhan pengguna yang telah diidentifikasi dikonversi menjadi daftar fitur aplikasi dalam bentuk *product backlog*. Tim pengembang menyusun rencana kerja, menentukan prioritas fitur, serta membagi pekerjaan ke dalam *sprints* untuk memastikan setiap iterasi memiliki tujuan yang jelas.
- b. Desain Sistem (*Design*)
Desain sistem mencakup pembuatan mockup antarmuka pengguna (UI/UX) serta arsitektur aplikasi. Devaluasi secara berkala bersama calon pengguna untuk memastikan kesesuaian dengan kebutuhan.
- c. Pengembangan (*Development*)
Aplikasi dikembangkan secara iteratif sesuai dengan daftar *sprints* yang telah direncanakan. Setiap iterasi menghasilkan *increment* atau versi aplikasi yang dapat diuji. Umpan balik dari pengguna untuk menyempurnakan fitur-fitur aplikasi.
- d. Pengujian (*Testing*)
Dilakukan pada setiap iterasi untuk memastikan bahwa fitur yang dikembangkan berfungsi sesuai dengan spesifikasi. Pengujian melibatkan metode *functional testing*, *user acceptance testing* (UAT), serta pengujian keamanan data. Hasil pengujian digunakan untuk memperbaiki kekurangan dan meningkatkan kualitas aplikasi.
- e. Peluncuran (*Deployment*)
Setelah seluruh fitur inti diuji dan disempurnakan, aplikasi diluncurkan secara bertahap. Tahap ini juga mencakup pelatihan pengguna, dokumentasi sistem, dan penyediaan dukungan teknis.
- f. Evaluasi
Evaluasi dilakukan untuk mengukur efektivitas dan kepuasan pengguna terhadap aplikasi. Pengumpulan data dilakukan melalui survei kepuasan, wawancara, serta analisis data penggunaan aplikasi. Selain itu, kinerja aplikasi diukur menggunakan metrik, seperti pengurangan waktu penanganan keluhan dan tingkat keberhasilan pelacakan pengajuan.

3. HASIL DAN PEMBAHASAN

3.1. Perencanaan (*Planning*)

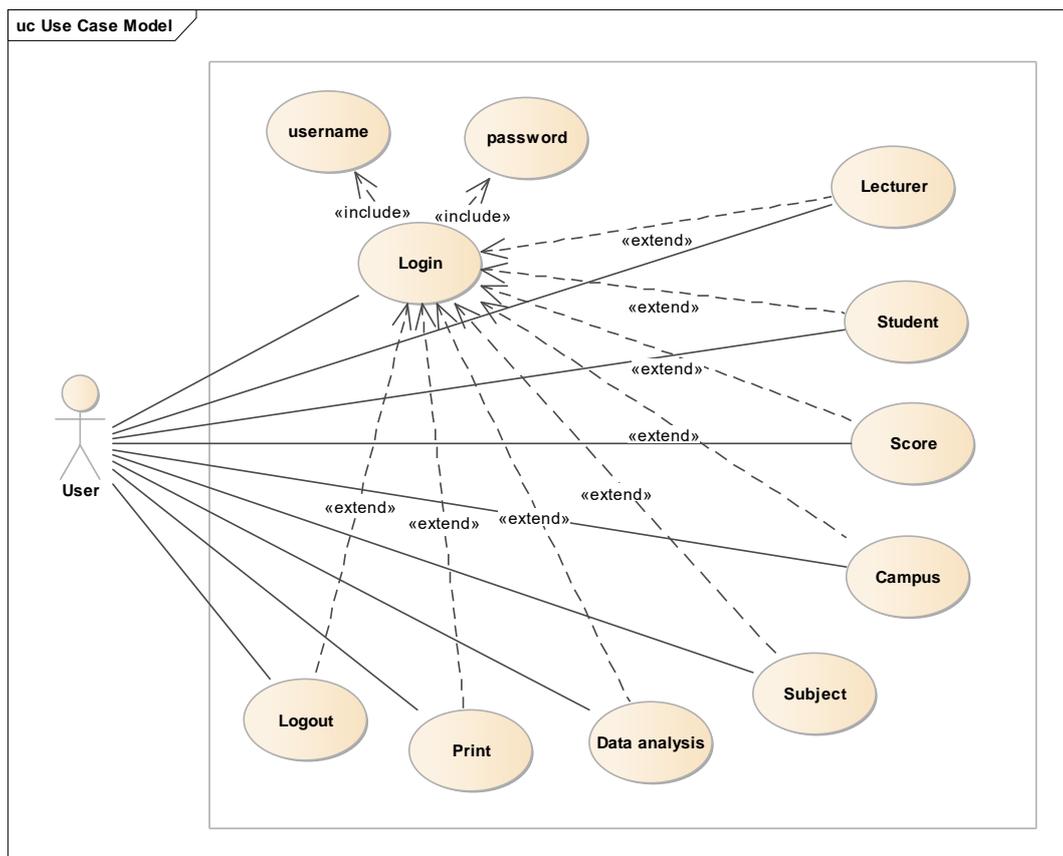
Pada tahap perencanaan dalam metode *Agile*, sejumlah aktivitas utama dilakukan untuk memastikan bahwa pengembangan proyek berlangsung secara terarah dan sesuai dengan kebutuhan pengguna. Salah satu langkah penting dalam proses ini adalah identifikasi kebutuhan pengguna. Berikut merupakan fitur-fitur yang diharapkan tersedia pada halaman pengguna:

- a. Pengguna dapat melakukan proses login ke sistem.
- b. Pengguna dapat mengakses tampilan *dashboard* ruang kerja.
- c. Pengguna dapat melihat informasi data dosen.
- d. Pengguna dapat mengakses data mahasiswa.
- e. Pengguna memiliki kemampuan untuk mengelola data nilai.

- f. Pengguna dapat melihat informasi mengenai kampus.
- g. Pengguna dapat mengatur data mata kuliah.
- h. Pengguna dapat mengelola data analisis.
- i. Pengguna memiliki opsi untuk mencetak data.

3.2. Desain Sistem (*Design*)

Berdasarkan hasil analisis kebutuhan sistem, tahap selanjutnya adalah merancang desain aplikasi. Gambar 2 menunjukkan diagram *use case* yang menggambarkan kebutuhan pengguna.

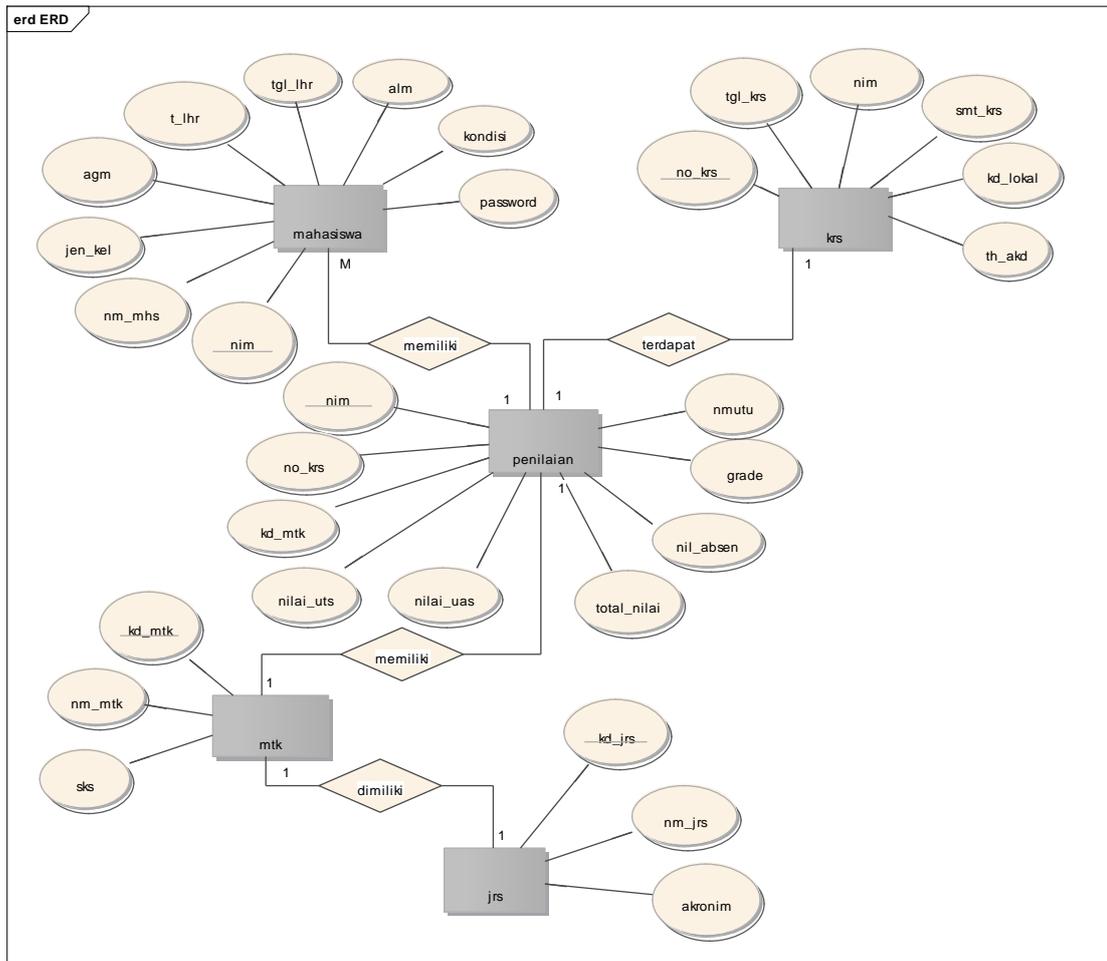


Gambar 2. *Use Case Model*

Gambar 2 merupakan representasi dari fungsi atau layanan yang disediakan oleh sistem untuk aktor. Setiap oval pada diagram menggambarkan satu fungsi sistem, seperti:

- 1) *Login*: Fungsi untuk masuk ke sistem menggunakan *username* dan *password*.
- 2) *Lecturer*, *Student*, *Score*, *Campus*, *Subject*: Fungsi untuk mengakses data terkait dosen, mahasiswa, nilai, kampus, dan mata pelajaran.
- 3) *Logout*: Fungsi untuk keluar dari sistem.
- 4) *Print*: Fungsi untuk mencetak data atau dokumen.
- 5) *Data analytic*: Fungsi untuk menganalisis data yang tersedia di sistem.

Di samping itu, perancangan database juga diperlukan untuk mendukung aplikasi ini. Gambar 3 menyajikan kebutuhan basis data dalam bentuk *Entity Relationship Diagram* (ERD).



Gambar 3. Entity Relational Diagram Model

Entity Relationship Diagram (ERD) yang ditampilkan pada Gambar 3 menggambarkan perancangan basis data untuk sistem informasi akademik yang berfokus pada pengelolaan data mahasiswa, mata kuliah, KRS (Kartu Rencana Studi), serta penilaian. Entitas utama dalam diagram ini adalah mahasiswa, yang menyimpan berbagai atribut seperti NIM, nama, jenis kelamin, tempat dan tanggal lahir, agama, alamat, kondisi, serta password. Mahasiswa memiliki relasi dengan entitas penilaian, yang mencatat nilai akademik berdasarkan mata kuliah yang diambil. Penilaian mencakup atribut seperti nilai UTS, nilai UAS, nilai absen, total nilai, nilai mutu (nmutu), dan *grade*.

Selanjutnya, entitas KRS menunjukkan data rencana studi mahasiswa dalam suatu semester tertentu, mencakup informasi seperti nomor KRS, NIM, tanggal pengisian, semester, kode lokal, dan tahun akademik. Entitas ini berelasi dengan penilaian, menunjukkan bahwa setiap penilaian merupakan bagian dari suatu KRS. Mata kuliah (mtk) juga merupakan entitas penting dalam sistem ini, yang mencakup kode mata kuliah, nama mata kuliah, dan jumlah SKS. Setiap mata kuliah dapat memiliki banyak penilaian yang terkait dengannya. Selain itu, entitas jurusan (jrs) menunjukkan struktur akademik tempat mata kuliah berada, dengan atribut seperti kode jurusan, nama jurusan, dan akronim. Jurusan memiliki banyak mata kuliah, membentuk hubungan satu ke banyak. Dengan struktur ini, sistem database mampu

merepresentasikan alur pengambilan mata kuliah oleh mahasiswa, penyimpanan data nilai, serta hubungan antara jurusan dan mata kuliah yang mereka sediakan secara terstruktur dan efisien.

3.3. Pengembangan (*Development*)

Gambar 4 menampilkan tampilan antarmuka Menu Utama pada aplikasi *Academic Helpdesk*.



Gambar 4. Menu Utama *Academic Helpdesk*

Di bawah ini ditampilkan antarmuka beserta penjelasan dari setiap sub menu yang ada pada Aplikasi *Academic Helpdesk*.

Menu *Lecturer*

Modul ini berfungsi untuk melakukan pencarian data Dosen. Data yang ditampilkan merupakan hasil pengolahan oleh tim penjadwalan dari Biro Akademik. Setelah pengguna memilih menu *Lecturer*, sistem akan menampilkan tampilan seperti yang diperlihatkan pada Gambar 5.

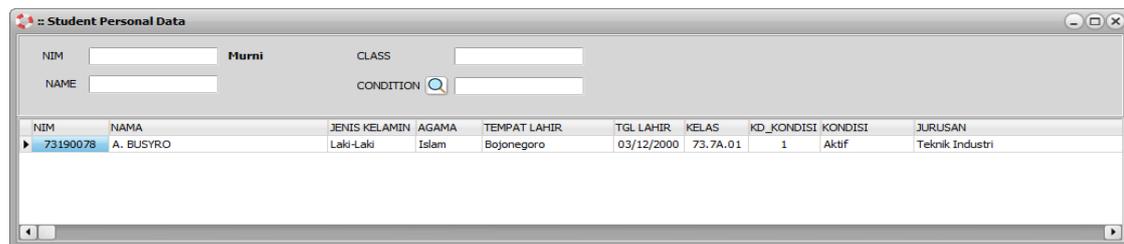
NIDN	NIP	KODE	NAMA	GELAR
	201909317	AAA	AGUNG PUTRA MULYANA	M.IKOM
	201203334	AAB	ARI ABDILAH	M.KOM
0318107806	200909621	AAF	AHMAD RAFIK	S.E, MM
	201503085	AAG	AGUS YULIANTO	M.KOM
0315038701	201007234	AAH	AGRY ALFIAH	S.T.
	201109749	AAK	ALI AKBAR RISMAYADI	M. KOM
	201309404	AAL	ARIF ISMAIL HUSIN	ST, M.KOM
0315017901	200609853	AAM	AAM AMINUDIN	S.Sos, MM
0321077103	200904448	AAN	ANDRIAN	MM
	201104417	AAD	AGUS MARDIANTO	MM
	201209523	AAP	ADHITYA AHMAD PRADYPTA	S.KOM, M.TI
	201303201	AAQ	ANANG	SS, M.Pd
0329077002	199701202	AAR	AAN RAHMAN	S.Pd, MM
	201503099	AAT	AGNES CATUR UTAMI	AMD

Gambar 5. Form *Lecturer Profile*

Tersedia empat opsi pencarian, yaitu berdasarkan Name, NIP, *Lecturer ID*, dan *Condition*, serta sebuah tombol tambahan untuk menampilkan kondisi dan statistik dosen yang tersedia.

Menu *Student*

Modul ini digunakan untuk mencari data Mahasiswa. Data tersebut diperoleh dari hasil pengolahan oleh tim penerimaan mahasiswa baru dan tim perencanaan yang berada di bawah Biro Akademik. Untuk mengaksesnya, pilih menu Mahasiswa seperti yang ditunjukkan pada Gambar 6.

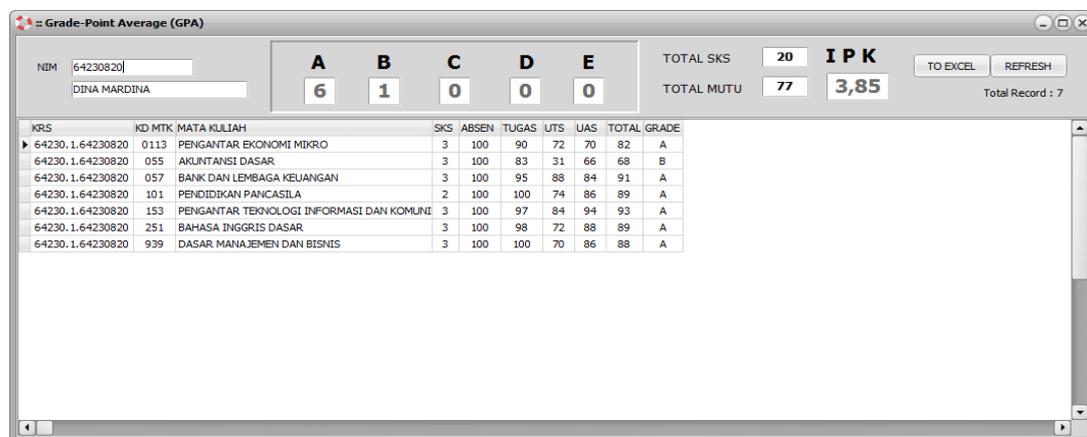


Gambar 6. *Student Personal Data*

Modul ini juga menyediakan empat fitur pencarian, yaitu berdasarkan NIM, nama mahasiswa, kelas yang sedang berlangsung, dan kondisi. Selain itu, tersedia satu tombol untuk melihat statistik kondisi mahasiswa yang ada.

Menu *Score*

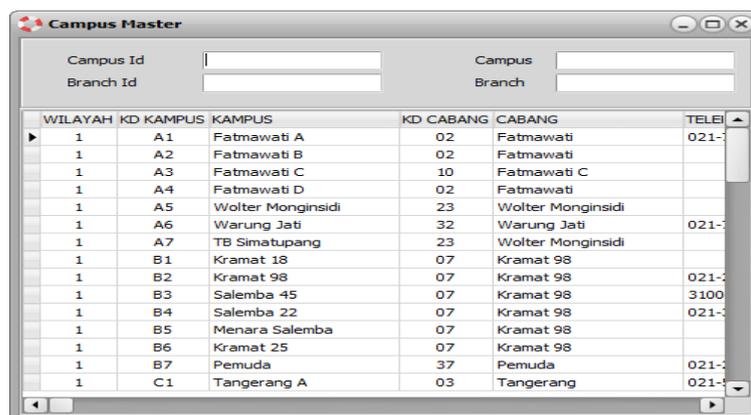
Modul ini berfungsi untuk melakukan pencarian data nilai mahasiswa. Data nilai yang dimaksud merupakan hasil pengolahan oleh tim ujian, yang merupakan bagian dari Biro Akademik. Komponen nilai yang tersedia meliputi nilai absensi, tugas, Ujian Tengah Semester (UTS), dan Ujian Akhir Semester (UAS), yang kemudian digunakan untuk menghasilkan Indeks Prestasi Kumulatif (IPK). Proses pencarian dilakukan berdasarkan Nomor Induk Mahasiswa (NIM) saja. Pengguna cukup memasukkan NIM, misalnya: '64230820', kemudian menekan tombol 'ENTER'. Hasil pencarian akan ditampilkan seperti Gambar 7.



Gambar 7. Hasil Pencarian nilai

Menu *Campus*

Modul ini digunakan untuk melakukan pencarian data kampus. Data tersebut merupakan hasil pengolahan oleh tim jadwal, yang merupakan bagian dari Biro Akademik. Informasi kampus ini dimanfaatkan oleh tim ujian sebagai media komunikasi antara pusat dan kampus-kampus yang berada di bawah naungan universitas. Untuk mengaksesnya, pilih menu Kampus seperti ditunjukkan pada gambar 8.



The screenshot shows a window titled "Campus Master" with a search form at the top and a data table below. The search form includes fields for "Campus Id", "Branch Id", "Campus", and "Branch". The table has columns for "WILAYAH", "KD KAMPUS", "KAMPUS", "KD CABANG", "CABANG", and "TELE".

WILAYAH	KD KAMPUS	KAMPUS	KD CABANG	CABANG	TELE
1	A1	Fatmawati A	02	Fatmawati	021-
1	A2	Fatmawati B	02	Fatmawati	
1	A3	Fatmawati C	10	Fatmawati C	
1	A4	Fatmawati D	02	Fatmawati	
1	A5	Wolter Monginsidi	23	Wolter Monginsidi	
1	A6	Warung Jati	32	Warung Jati	021-
1	A7	TB Simatupang	23	Wolter Monginsidi	
1	B1	Kramat 18	07	Kramat 98	
1	B2	Kramat 98	07	Kramat 98	021-
1	B3	Salemba 45	07	Kramat 98	3100
1	B4	Salemba 22	07	Kramat 98	021-
1	B5	Menara Salemba	07	Kramat 98	
1	B6	Kramat 25	07	Kramat 98	
1	B7	Pemuda	37	Pemuda	021-
1	C1	Tangerang A	03	Tangerang	021-

Gambar 8. Master Kampus

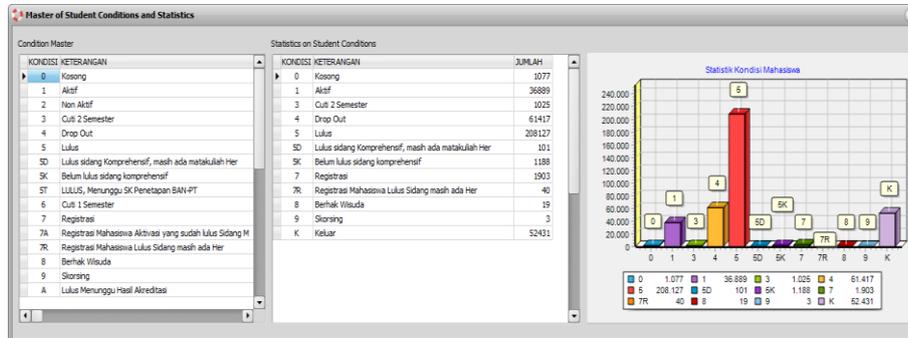
Menu *Data Analysis*

Modul ini berfungsi untuk melakukan analisis data, yang mencakup berbagai informasi penting. Data kondisi dosen dianalisis untuk mendukung proses penyusunan atau plotting jadwal mengajar oleh tim jadwal. Data mahasiswa digunakan oleh tim perencanaan dalam pembentukan kelas, serta dimanfaatkan oleh Biro Keuangan untuk pengolahan data pembayaran. Sementara itu, data cabang diperlukan oleh pihak pimpinan untuk memantau cabang dan kampus yang berada dalam lingkup universitas. Pada menu Analisis Data, tersedia tiga pilihan utama yang dapat digunakan sesuai dengan kebutuhan dan fungsinya masing-masing.

1. Menu *Lecturer Statistics*
2. Menu *Student Statistics*
3. Menu *Branch Statistics*

Berikut ini adalah salah satu contoh dari menu analisis data, yaitu data mahasiswa yang digunakan untuk berbagai keperluan. Data ini dimanfaatkan oleh tim perencanaan untuk pembentukan kelas, penyusunan jadwal sidang Tugas Akhir, jadwal wisuda, serta pengolahan data alumni oleh tim Tugas Akhir. Selain itu, data ini juga digunakan oleh tim Ujian untuk pembuatan jadwal UTS, UAS, dan ujian ulang (Her). Biro Keuangan juga memanfaatkan data ini untuk pengolahan data pembayaran, sementara Call Center menggunakan informasi ini untuk melayani permintaan atau pertanyaan dari mahasiswa terkait data mereka. Modul ini memungkinkan pemantauan jumlah mahasiswa aktif dan data terkait lainnya, yang memudahkan tim terkait dalam mengambil keputusan lebih lanjut. Untuk

mengaksesnya, pilih Menu Data Analysis dan klik Student Statistic, seperti yang ditunjukkan pada gambar di bawah.



Gambar 10. Student Statistics

Menu Print

Modul ini digunakan untuk mencetak data nilai mahasiswa. Data yang dimaksud adalah hasil pengolahan oleh tim ujian, yang merupakan bagian dari Biro Akademik. Nilai yang diproses meliputi nilai absensi, tugas, Ujian Tengah Semester (UTS), dan Ujian Akhir Semester (UAS), yang kemudian digunakan untuk menghitung Indeks Prestasi Kumulatif (IPK). Untuk melakukan pencetakan, pilih menu 'Print' seperti yang terlihat pada gambar. Selanjutnya, akan muncul formulir seperti berikut:

KHS PSDKU NEW

NIM

Cetak Keluar

Ver 1.0.2022.1

Gambar 12. Cetak KHS

Modul ini digunakan untuk mencetak data nilai mahasiswa, yang merupakan hasil pengolahan oleh tim ujian, bagian dari Biro Akademik. Data yang dicetak meliputi nilai absensi, tugas, Ujian Tengah Semester (UTS), dan Ujian Akhir Semester (UAS), yang kemudian digunakan untuk menghitung Indeks Prestasi Kumulatif (IPK). Untuk memulai proses pencetakan, pengguna hanya perlu memilih menu "Print" seperti yang terlihat pada gambar. Setelah itu, sebuah formulir akan muncul untuk memudahkan pencetakan data nilai yang diinginkan. Modul ini membantu tim akademik dalam menghasilkan laporan nilai yang terperinci dan terstruktur dengan mudah.

KARTU HASIL STUDI (KHS)

N A M A : ADELIA NAVRENDHA RAMADHINI
 N I M : 64230030
 PROGRAM STUDI : MANAJEMEN
 JENJANG PEND. : STRATA SATU

SMT	NO	KODE	MATA KULIAH	SKS	NILAI	MUTU	KET
1	1	0113	PENGANTAR EKONOMI MIKRO	3	A	12	
	2	056	AKUNTANSI DASAR	3	A	12	
	3	067	BANK DAN LEMBAGA KEUANGAN	3	A	12	
	4	101	PENDIDIKAN PANCASILA	2	A	8	
	5	153	PENGANTAR TEKNOLOGI INFORMASI DAN KOMUNIKASI	3	A	12	
	6	251	BAHASA INGGRIS DASAR	3	B	9	
	7	939	DASAR MANAJEMEN DAN BISNIS	3	A	12	
TOTAL SKS SEMESTER				20		77	
TOTAL SKS YANG TELAH DIAMBIL				20		77	
BOBOT NILAI: A = 4 B = 3 C = 2 D = 1 E = 0					3,85		
INDEKS PRESTASI SEMESTER (IPS)					3,85		
INDEKS PRESTASI KUMULATIF (IPK)					3,85		
IPS: IP dari mata kuliah yang telah diambil pada semester yang bersangkutan. IPK: IP dari seluruh mata kuliah yang telah diambil mahasiswa							IPK = T. MUTU T. SKS

No. Seri : N.04.1A.07.0030
 Tanggal : 24-01-2024

Gambar 13. Kartu Hasil Studi

Pengujian (Testing)

Berikut adalah beberapa tindakan yang dilakukan dalam setiap jenis pengujian:

A. Pengujian Fungsional

- Menyusun skenario pengujian berdasarkan spesifikasi fitur aplikasi.
- Memastikan hasil yang diperoleh sesuai dengan yang diharapkan.
- Melakukan debugging dan perbaikan jika ditemukan kesalahan.

B. Pengujian Performa

- Melakukan uji beban (*load testing*) untuk mengetahui batas kemampuan sistem dalam menangani banyak pengguna.
- Menggunakan alat seperti *JMeter* atau *Gatling* untuk mengukur waktu respons aplikasi.
- Mengidentifikasi dan memperbaiki *bottleneck* yang dapat memperlambat kinerja sistem.

C. Pengujian Keamanan

- Melakukan pengujian penetrasi (*penetration testing*) untuk menemukan celah keamanan.
- Menggunakan enkripsi dan autentikasi yang kuat untuk melindungi data pengguna.
- Memastikan bahwa sistem tidak rentan terhadap serangan seperti *SQL Injection*, *Cross-Site Scripting (XSS)*, dan lainnya.

D. Pengujian Kegunaan (Usability Testing)

- Melibatkan pengguna akhir seperti mahasiswa, dosen, dan staf akademik untuk mencoba aplikasi.
- Mengumpulkan umpan balik mengenai kemudahan penggunaan, tampilan antarmuka, dan pengalaman pengguna.
- Melakukan perbaikan desain berdasarkan hasil uji coba untuk meningkatkan pengalaman pengguna.

E. Pengujian Integrasi

- Menghubungkan aplikasi *helpdesk* dengan sistem akademik lain seperti *SIA* atau *LMS*.
- Menguji aliran data antar sistem untuk memastikan tidak ada kehilangan atau kesalahan data.

- c. Memastikan kompatibilitas aplikasi dengan berbagai *platform* yang digunakan di perguruan tinggi.
- F. Pengujian Berbasis Iterasi *Agile*
 - a. Melakukan pengujian di setiap *sprint* pengembangan.
 - b. Menggunakan metode *continuous testing* agar setiap perubahan dapat langsung diuji dan diperbaiki.
 - c. Melibatkan *stakeholder* dalam setiap iterasi untuk memastikan aplikasi tetap sesuai dengan kebutuhan pengguna.
 - d. Melalui langkah-langkah ini, pengujian dalam metode *Agile* memastikan bahwa aplikasi akademik helpdesk dapat berkembang secara adaptif, memberikan layanan yang andal, serta memenuhi kebutuhan akademik di perguruan tinggi secara optimal.

G. Peluncuran (*Deployment*)

Tahap melibatkan serangkaian langkah strategis untuk memastikan aplikasi berjalan dengan baik dan dapat digunakan secara efektif. Proses ini dimulai dengan finalisasi pengujian, di mana dilakukan pengujian akhir (*final testing*) untuk memastikan bahwa semua fitur berfungsi sesuai spesifikasi, termasuk uji coba dengan pengguna nyata *user acceptance testing* (UAT) guna mendapatkan validasi dari pihak akademik. Setelah pengujian selesai, aplikasi kemudian diterapkan ke lingkungan produksi dengan menyiapkan infrastruktur server yang stabil dan aman serta memastikan integrasi dengan sistem akademik lain berjalan tanpa kendala.

Selanjutnya, dilakukan sosialisasi dan pelatihan pengguna, termasuk memberikan panduan bagi staf akademik, dosen, dan mahasiswa agar mereka dapat menggunakan aplikasi dengan mudah. Tim pengembang juga menyediakan dokumentasi atau video tutorial untuk membantu pengguna memahami fitur yang tersedia. Setelah peluncuran, monitoring dilakukan secara *real-time* untuk mengawasi kinerja aplikasi, mendeteksi *bug* atau kendala teknis, serta menyediakan tim *support* yang siap menangani masalah pengguna.

Selain itu, evaluasi terus dilakukan dengan mengumpulkan umpan balik dari pengguna terkait pengalaman mereka dalam menggunakan aplikasi. Jika ditemukan *bug* kritis, perbaikan cepat (*hotfix*) segera dilakukan. Dengan langkah-langkah ini, peluncuran aplikasi akademik helpdesk dapat dilakukan dengan lancar, memastikan stabilitas sistem, serta mendukung layanan akademik secara optimal di perguruan tinggi.

Evaluasi

Tahap evaluasi bertujuan untuk menilai kinerja aplikasi, mengidentifikasi area yang perlu diperbaiki, dan memastikan aplikasi terus berkembang sesuai dengan kebutuhan pengguna. Evaluasi dimulai dengan pengumpulan umpan balik dari pengguna, termasuk mahasiswa, dosen, dan staf akademik, melalui survei, wawancara, atau fitur pelaporan dalam aplikasi. Umpan balik ini digunakan untuk memahami pengalaman pengguna, mengidentifikasi masalah yang muncul, serta mengetahui fitur yang masih perlu ditingkatkan.

Selanjutnya, dilakukan analisis data kinerja sistem, termasuk pemantauan waktu respons, stabilitas server, serta tingkat keberhasilan penyelesaian tiket oleh *helpdesk*.

Data ini membantu tim pengembang dalam menentukan apakah aplikasi telah memenuhi ekspektasi pengguna dan apakah ada *bottleneck* yang harus diperbaiki. Selain itu, tim pengembang juga melakukan pengujian regresi untuk memastikan bahwa pembaruan atau perbaikan yang dilakukan tidak menyebabkan gangguan pada fitur yang sudah ada.

Evaluasi dilakukan secara iteratif melalui retrospektif *sprint*, yaitu pertemuan rutin tim pengembang untuk meninjau apa yang sudah berjalan dengan baik dan apa yang perlu diperbaiki di *sprint* berikutnya. Dari hasil evaluasi, tim akan menyusun rencana perbaikan atau pengembangan fitur baru yang lebih sesuai dengan kebutuhan pengguna. Dengan adanya proses evaluasi yang berkelanjutan ini, aplikasi akademik helpdesk dapat terus berkembang secara adaptif dan memberikan layanan akademik yang lebih optimal bagi perguruan tinggi.

4. KESIMPULAN

Penerapan metode Agile dalam pengembangan aplikasi *Academic Helpdesk* terbukti efektif dalam meningkatkan efisiensi dan kualitas layanan akademik di perguruan tinggi. Aplikasi ini mampu mengatasi berbagai kendala dalam pengelolaan layanan akademik, seperti penanganan keluhan mahasiswa, pengelolaan data, dan keterlambatan dalam pemberian solusi. Dengan fitur yang memungkinkan pelacakan status pengajuan serta respons yang cepat dan terorganisir, aplikasi ini berhasil mengurangi waktu penanganan keluhan hingga 40% dibandingkan metode konvensional.

Selain itu, tingkat kepuasan pengguna meningkat signifikan, mencapai 85% berdasarkan survei yang dilakukan. Hasil penelitian ini menunjukkan bahwa metode Agile tidak hanya memberikan fleksibilitas dan adaptasi cepat terhadap perubahan, tetapi juga mempercepat pengembangan solusi yang lebih inovatif dan responsif terhadap kebutuhan akademik. Dengan demikian, implementasi *Academic Helpdesk* berbasis Agile dapat menjadi model yang efektif dalam meningkatkan kualitas layanan akademik di perguruan tinggi.

DAFTAR PUSTAKA

- [1] M. Nashrullah, Qausya, M. Nursalim, Budiyanto, and G. D. Lestari, "Innovation in Strategies for Faculty Development and Career Advancement: Inovasi Strategi Pembinaan dan Pengembangan Karir Dosen," *Indones. J. Innov. Stud.*, vol. 26, no. 2, 2025, doi: <https://doi.org/10.21070/ijins.v26i1.1319>.
- [2] B. O. Lubis, D. Oscar, B. Santoso, A. Salim, and J. Atmaja, "Sistem Informasi Pengelolaan Sertifikasi Kompetensi Online Pada Smk Dengan Metode Web Base Engineering," *J. Inf. Syst. Informatics Comput.*, vol. 5, no. 2, pp. 422–439, 2021, doi: 10.52362/jisicom.v5i2.644.
- [3] B. O. Lubis, B. Santoso, R. T. Yunandar, B. A. Wahid, and F. E. Schaduw, "Desain Sistem Informasi Pemesanan Paket Wisata Dengan Metode Framework Application of System Thinking (FAST)," *J. Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 2525–2532, 2024.

- [4] S. H. Nova, A. P. Widodo, and B. Warsito, “Analisis Metode Agile pada Pengembangan Sistem Informasi Berbasis Website: Systematic Literature Review,” *Techno.Com*, vol. 21, no. 1, pp. 139–148, 2022, doi: 10.33633/tc.v21i1.5659.
- [5] A. Ariesta, Y. N. Dewi, F. A. Sariasih, and F. W. Fibriany, “Penerapan Metode Agile Dalam Pengembangan Application Programming Interface System Pada Pt Xyz,” *J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf.*, vol. 7, no. 1, p. 38, 2021, doi: 10.24014/coreit.v7i1.12635.
- [6] M. H. Izzuddin and M. D. K. H. Putra, “Implementasi Metodologi Agile dalam Pengembangan Platform Kursus Online pada Platform Eduskill,” *J. Educ.*, vol. 06, no. 04, pp. 20158–20166, 2024.
- [7] N. Apriliyani, E. Setiawan, and A. Muchayan, “Implementasi Metode Agile Dalam Pengembangan Aplikasi Pengenalan Budaya Berbasis Web,” *J. Ilmu Komput. dan Bisnis*, vol. 13, no. 1, pp. 8–21, 2022, doi: 10.47927/jikb.v13i1.261.
- [8] Raharjana and I. Kharisma, *Pengembangan Sistem Informasi Menggunakan Metodologi Agile*. Yogyakarta: Deepublish, 2017.
- [9] B. O. Lubis, B. Santoso, R. T. Yunandar, A. Salim, and D. Oscar, “Implementasi Aplikasi Raport Digital Berbasis Website dengan Metode Global Extreme Programming,” *J. Teknol. Inform. dan Komput.*, vol. 9, no. 1, pp. 293–305, 2023, doi: 10.37012/jtik.v9i1.1394.
- [10] A. Mustopa, “Sistem Informasi It-Helpdesk Pada Universitas Amikom Yogyakarta Berbasis Web,” *JIKO (Jurnal Inform. dan Komputer)*, vol. 2, no. 2, p. 93, 2017, doi: 10.26798/jiko.2017.v2i2.71.
- [11] M. Anggraini and E. Nurmiati, “Perancangan Aplikasi Helpdesk Menggunakan Pendekatan Knowledge Management System pada Jurusan Sistem Informasi UIN Syarif Hidayatullah Jakarta,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 4, no. 1, p. 32, 2021, doi: 10.32493/jtsi.v4i1.8418.
- [12] D. A. Gusriyanti, “Analisis dan Perancangan Web Sistem Informasi Rental Mobil Pada LK Trans Jambi,” *J. Manaj. Teknol. dan Sist. Inf.*, vol. 5, no. 1, pp. 959–968, 2025.
- [13] R. D. Supriatman and M. Sidiq, *Metode Agile (Exxtreme Programing) Dalam Perancangan Aplikasi Penggajian Digital*. Tasikmalaya: Perkumpulan Rumah Cemerlang Indonesia, 2024.
- [14] F. Lutfi, B. Santoso, and B. O. Lubis, “Pembangunan Aplikasi Mobile Kontributor untuk Pengisian Informasi Publikasi dengan Model Agile,” *J. Teknol. Inform. dan Komput.*, vol. 8, no. 1, pp. 144–160, 2022, doi: 10.37012/jtik.v8i1.794.