

Sistem Absensi dan Penghitung Fee Bimbingan Belajar SMART berbasis Website dengan Metode Agile

DOI: <http://dx.doi.org/10.35889/jutisi.v15i1.3412>

Creative Commons License 4.0 (CC BY – NC)



Muhammad Sammy Atha¹, Badzlan Nur Dhabith², Nur'afia Avanza³, Adzkia Nifa Adha⁴,
Argy Fawwaz Akbar⁵, Aditya Wicaksono^{6*}, Endang Purnama Giri⁷, Hanifah Nur Zahrah⁸

^{1,2,3,4,5,6}Teknologi Rekayasa Perangkat Lunak, IPB University, Bogor, Indonesia

⁷Ilmu Komputer, IPB University, Bogor, Indonesia

⁸Hukum Ekonomi Syariah, STAI Persis Jakarta, Jakarta, Indonesia

*e-mail Corresponding Author: adityawicaksono@apps.ipb.ac.id

Abstract

The tutor attendance recording and honorarium calculation process at Bimbel SMART was previously conducted manually using spreadsheets, which was inefficient and prone to errors. This study aimed to design a web-based information system to automate attendance tracking and honorarium calculation accurately. The Agile Development methodology was applied through requirements identification, design, implementation, and functional testing stages. The result of this research was an integrated system covering data management, scheduling, and automated honorarium calculation. Based on black-box testing, the system proved successful in validating honorarium calculation logic that excluded absence status and enforcing access control limits between admins and tutors. It was concluded that the system implementation effectively minimized manual calculation error risks and ensured administrative data validity through verified system mechanisms.

Keywords: Agile Development; Attendance; Honorarium; Information systems; Tutoring center.

Abstrak

Proses absensi tutor dan perhitungan honorarium di Bimbingan Belajar SMART sebelumnya dilakukan secara manual menggunakan *spreadsheets* yang tidak efisien dan rentan kesalahan. Penelitian ini bertujuan merancang sistem informasi berbasis web untuk mengotomatisasi pencatatan absensi dan perhitungan honorarium secara akurat. Metodologi Agile Development diterapkan melalui tahapan identifikasi kebutuhan, perancangan, implementasi, dan pengujian fungsional. Hasil penelitian ini berupa sistem terintegrasi yang mencakup manajemen data, penjadwalan, dan kalkulasi honor otomatis. Berdasarkan pengujian *black-box*, sistem terbukti berhasil memvalidasi logika perhitungan honor yang mengecualikan status ketidakhadiran (alpha) dan menegakkan batasan hak akses antara admin dan tutor. Implementasi sistem ini disimpulkan efektif dalam meminimalkan risiko kesalahan perhitungan manual dan menjamin validitas data administratif melalui mekanisme sistem yang teruji.

Kata kunci: Absensi; Agile Development; Bimbingan belajar; Honorarium; Sistem informasi.

1. Pendahuluan

Pengelolaan sumber daya manusia dalam institusi pendidikan, khususnya terkait absensi dan honorarium tutor, memegang peranan penting dalam menjaga kelancaran operasional. Namun, pelaksanaan absensi dan penghitungan honor yang masih dilakukan secara manual sering menimbulkan berbagai permasalahan, seperti inefisiensi waktu, tingginya potensi kesalahan rekapitulasi, serta kurangnya transparansi data. Permasalahan tersebut banyak ditemukan di berbagai institusi pendidikan. Sebagai contoh, Keivin *et al.* melaporkan bahwa proses penggajian guru yang masih dilakukan secara manual menyebabkan ketidakakuratan data dan keterlambatan pencarian informasi [1]. Kondisi ini menunjukkan pentingnya penerapan sistem informasi digital untuk meningkatkan efisiensi dan akurasi administrasi.

Permasalahan serupa juga ditemukan pada objek kajian penelitian ini, yaitu Bimbel SMART. Saat ini, pencatatan kehadiran tutor dan perhitungan honor masih dilakukan menggunakan *spreadsheets*, sehingga rawan terhadap kesalahan input, duplikasi data, serta keterlambatan rekapitulasi. Selain itu, proses pemantauan kehadiran dan perhitungan honor belum terintegrasi secara langsung, sehingga berpotensi menurunkan akuntabilitas terhadap tutor. Kondisi tersebut menunjukkan perlunya sistem absensi dan penghitungan honorarium yang terkomputerisasi dan terintegrasi.

Berbagai penelitian terdahulu telah membahas solusi untuk permasalahan absensi dan penggajian dengan pendekatan model proses yang beragam, meskipun masih memiliki keterbatasan fungsional. Triyono et al. mengembangkan sistem absensi berbasis web yang memfokuskan alur kerja pada digitalisasi logbook manual menjadi basis data terpusat, sehingga pencatatan kehadiran menjadi lebih efisien namun belum mencakup pemrosesan honorarium [2]. Pada aspek validasi data, Pulungan et al. merancang aplikasi dengan fitur pemindaian QR Code pada platform Android untuk mempercepat input presensi [3], sedangkan Putra et al. mengombinasikan fitur autentikasi biometrik fingerprint dan validasi lokasi geolocation untuk menutup celah kecurangan kehadiran fisik [7].

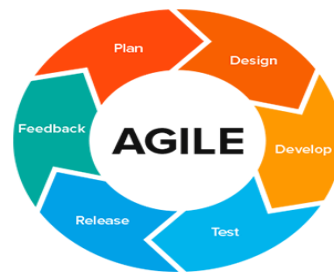
Di sisi lain, pada aspek manajemen keuangan, Keivin et al. serta Yani membangun sistem informasi penggajian dengan model proses kalkulasi otomatis berdasarkan parameter kehadiran guru yang terbukti meningkatkan ketepatan perhitungan honor [1][5], sementara Ansanay et al. melengkapi model tersebut dengan integrasi data kehadiran yang secara langsung memicu rekapitulasi honorarium guna mengurangi latensi pelaporan [4]. Meskipun demikian, mayoritas penelitian tersebut masih menerapkan model proses yang terfragmentasi. Penelitian Wijoyo et al. telah mencoba mengintegrasikan kedua fungsi tersebut, namun antarmuka pengguna yang disajikan belum menyediakan visualisasi data (*dashboard*) yang memadai [6]. Hal ini menunjukkan adanya celah penelitian pada pengembangan sistem terpadu yang tidak hanya mengintegrasikan basis data absensi dan honor, tetapi juga menyajikan alur kerja validasi bertingkat dengan kontrol akses berbasis peran (*Role-Based Access Control*).

Berdasarkan gap tersebut, penelitian ini mengusulkan pengembangan sistem absensi dan penghitungan honorarium tutor berbasis web yang terintegrasi. Kebaruan (*novelty*) dari penelitian ini terletak pada penggabungan modul absensi dan rekap honor dalam satu sistem dengan dukungan *multi-level user* (admin dan tutor) serta penyajian data dalam bentuk *dashboard* informatif. Konsep ini didukung oleh penelitian-penelitian sebelumnya yang menyatakan bahwa sistem digital berbasis web maupun mobile efektif dalam meningkatkan efisiensi layanan dan transparansi data, seperti yang ditunjukkan oleh Jaya et al., Septiawan et al., serta Ramadhan et al. yang menekankan pentingnya aksesibilitas layanan digital bagi sivitas akademika [9][10][11]. Selain itu, pendekatan pengembangan sistem menggunakan metode Agile Scrum dan *framework* Laravel juga didukung oleh penelitian Bintang et al. serta Atallah et al. yang menyimpulkan bahwa metode Agile efektif dalam pengembangan sistem yang adaptif terhadap kebutuhan pengguna [12][14]. Dengan konsep tersebut, sistem yang diusulkan diyakini mampu meningkatkan efisiensi, akurasi, dan transparansi pengelolaan absensi serta honorarium tutor di Bimbel SMART.

2. Metodologi

Penelitian ini menggunakan metode *Agile Development* dalam pengembangan perangkat lunak. Pendekatan Agile dipilih karena memungkinkan fleksibilitas dalam mengakomodasi perubahan kebutuhan pengguna dan iterasi pengembangan yang cepat, serta terbukti efektif dalam meningkatkan efisiensi pengembangan sistem informasi administratif [14].

Proses pengembangan dibagi ke dalam beberapa tahap utama, yaitu: pengumpulan kebutuhan, perancangan, implementasi, dan pengujian. Masing-masing tahap tidak dilaksanakan secara linear, melainkan iteratif dalam empat siklus *sprint*. Setiap *sprint* menghasilkan *increment* fungsional pada sistem sesuai prioritas kebutuhan.



Gambar 1. Metodologi Agile

1) Tahap Kebutuhan

Pada tahap ini, dilakukan identifikasi kebutuhan pengguna melalui wawancara dengan pemilik Bimbel SMART. Hasilnya disusun dalam dokumen *Software Requirements Specification (SRS)* yang memuat daftar kebutuhan fungsional dan non-fungsional sistem.

Salah satu temuan penting adalah aturan bisnis bahwa tutor dibayar per siswa per sesi jika siswa tersebut hadir atau izin/sakit, sedangkan ketidakhadiran (alpa) tidak dihitung dalam honor. Selain itu, terdapat batas maksimal 3 sesi per minggu per kelas, dan jika ada sesi tambahan (sesi ke-7) maka dihitung sebagai bonus 50% sesuai kebijakan bimbel. Aturan-aturan ini diakomodasi sebagai kebutuhan fungsional sistem. Prioritas kebutuhan ditentukan dengan pendekatan *MoSCoW (Must, Should, Could, Won't)* untuk penjadwalan dalam tiap sprint.

Tabel 1. Kebutuhan Fungsional

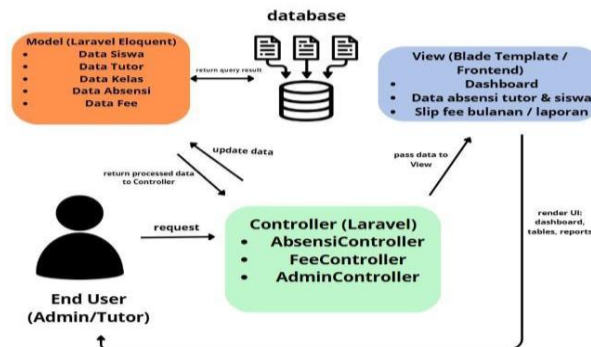
ID	Jenis	Deskripsi	Prioritas
FR-01	Fungsional	Login & RBAC sederhana untuk 3 peran: Siswa (user), Tutor, Admin	Tinggi
FR-02	Fungsional	CRUD data Siswa (nama, kelas, kontak minimal)	Tinggi
FR-03	Fungsional	CRUD data Tutor (nama, mata pelajaran/jadwal)	Tinggi
FR-04	Fungsional	CRUD data Kelas + Enrolment (Siswa ke Kelas + assign Tutor)	Tinggi
FR-05	Fungsional	Buat Sesi Pertemuan minimal (penjadwalan kelas 2–3x per minggu)	Tinggi
FR-06	Fungsional	Input Absensi (Hadir/Izin/Sakit/Alpa) per Sesi oleh Tutor	Tinggi
FR-07	Fungsional	Logika validasi absensi (alpa tidak dihitung pada honor)	Tinggi
FR-08	Fungsional	Hitung Fee Bulanan (otomatis, fee per siswa per sesi, bonus sesi ke-7)	Tinggi
FR-09	Fungsional	Dashboard Tutor sederhana (progress YTD & estimasi fee bulan berjalan)	Sedang
FR-10	Fungsional	Dashboard Admin (rekap tutor & honor, manajemen total fee)	Sedang
FR-11	Fungsional	Laporan Bulanan + Ekspor (Excel untuk Tutor/Admin/Periode)	Sedang
FR-12	Fungsional	Import CSV untuk seed data Siswa & Enrolment	Rendah
FR-13	Fungsional	Generator Slip Bulanan otomatis (opsional)	Rendah

2) Tahap Perancangan

Tahap perancangan dilakukan menggunakan pemodelan *Unified Modeling Language (UML)* dan perancangan antarmuka pengguna. Pemodelan UML meliputi diagram arsitektur sistem, *use case diagram*, *Entiti Relationship Diagram (ERD)*, dan *class diagram* untuk memastikan kesesuaian antara kebutuhan fungsional dan implementasi sistem.

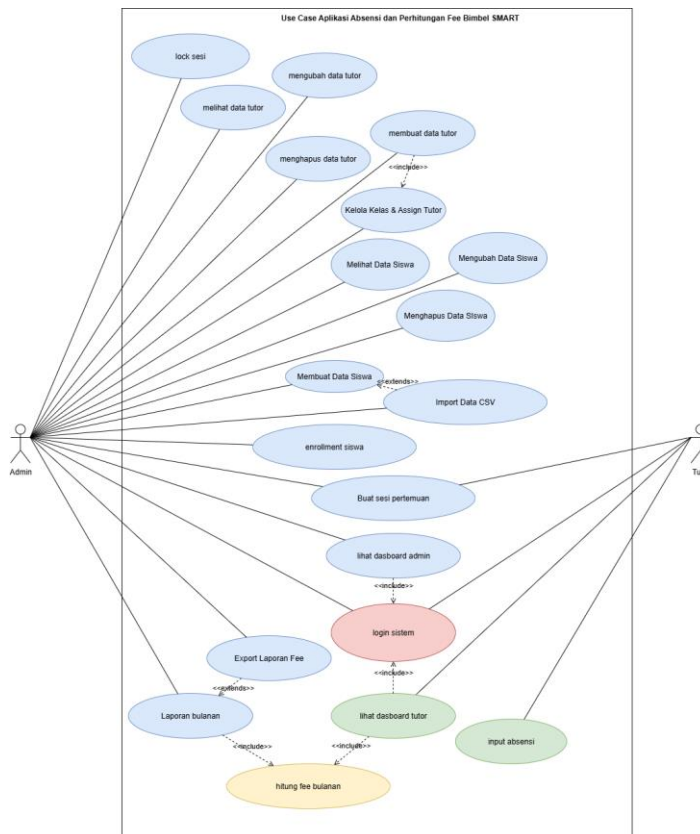
a) Model Arsitektur Sistem

Sistem ini menerapkan arsitektur *Model-View-Controller (MVC)* berbasis Laravel, yang alurnya dapat dilihat pada Gambar 2. *End User* berinteraksi dengan *Controller*, yang berkoordinasi dengan *Model* untuk memanipulasi data di *Database*, kemudian *View* menyusun dan merender antarmuka pengguna yang akan disajikan kembali kepada *End User*.



Gambar 2. Model Arsitektur Sistem

b) Use Case Diagram

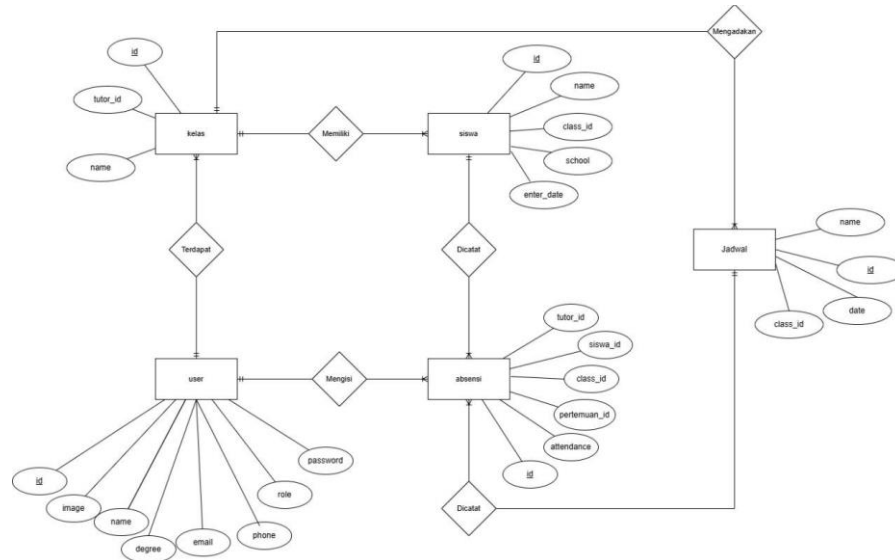


Gambar 3. Use Case Diagram

Diagram *Use Case* pada Gambar 3 di atas, berfungsi memberikan gambaran interaksi dua aktor pada sistem ini. Admin memiliki kontrol penuh, mulai dari *Login* sistem, pengelolaan data master (Siswa, Tutor, Kelas), penguncian sesi, hingga pembuatan Laporan Bulanan dan *Export* Laporan *Fee*. Sementara itu, Tutor berfokus pada *Input* Absensi dan melihat *Dashboard* Tutor, yang keduanya memicu proses utama hitung *fee* bulanan.

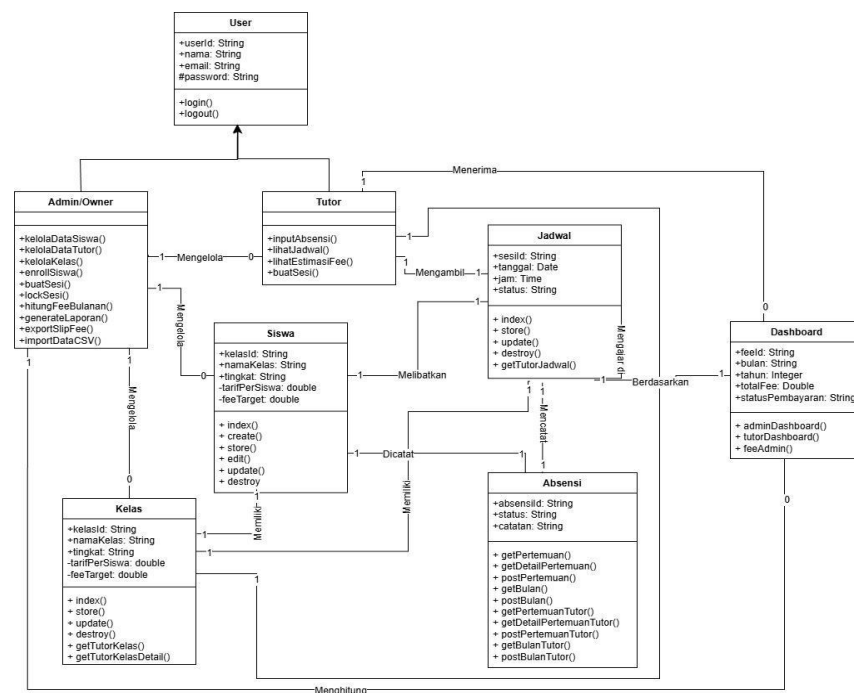
c) Entity Relationship Diagram (ERD)

Struktur data sistem, seperti terlihat pada Gambar 4, ditentukan oleh entitas utama *User*, *Kelas*, *Siswa*, *Jadwal*, dan *Absensi*. Entitas *Absensi* menjadi pusat yang merekam *attendance* atau kehadiran dengan mengaitkan data dari *User* (Tutor), *Siswa*, *Kelas*, dan *Jadwal* dalam satu kesatuan.



Gambar 4. Perancangan Basis Data (ERD)

d) Class Diagram



Gambar 5. Perancangan Kelas (Class Diagram)

Class diagram pada Gambar 5 memvisualisasikan struktur berorientasi objek sistem, *class* Admin/Owner dan Tutor mewarisi dari *User*. Diagram ini memisahkan tanggungjawab fungsional dan atribut *class* utama, seperti Kelas, Siswa, Jadwal, dan Absensi, serta secara jelas membedakan *method* manajerial Admin (seperti *hitungFeeBulanan*) dari *method* operasional Tutor (seperti *inputAbsensi*).

3) Tahap Pengembangan

Implementasi sistem dilakukan secara bertahap sesuai prinsip *Agile*, dengan membagi pekerjaan ke dalam empat *sprint* utama. Setiap *sprint* memiliki target *product backlog* tertentu sebagai berikut:

a) Sprint 1

Pengembangan modul autentikasi dan dashboard awal. Fitur yang direalisasikan meliputi halaman login (validasi *email* dan *password* untuk admin/tutor), dashboard admin yang menampilkan ringkasan data (jumlah tutor, siswa, kelas, dll.), dan modul kelola data tutor (CRUD data tutor). Pada tahap ini juga disiapkan struktur *database* awal dan integrasi kontrol akses berbasis peran yang terdiri dari admin dan tutor.

b) Sprint 2

Implementasi fungsionalitas inti absensi dan perhitungan honor. Fitur yang dibangun antara lain modul jadwal sesi (admin/tutor dapat menjadwalkan pertemuan atau sesi kelas), halaman *input* absensi (tutor memasukkan kehadiran siswa per sesi secara *online*), serta modul rekap honor bulanan untuk admin. Perhitungan honor tutor dilakukan otomatis oleh sistem berdasarkan data absensi yang ditandai *valid* setiap akhir bulan. Logika perhitungan mengikuti aturan yang ditetapkan (hanya menghitung siswa hadir/izin/sakit, mengabaikan alpa, batas sesi per minggu, dsb.) sehingga honor yang terhitung akurat sesuai kebijakan bimbel.

c) Sprint 3

Penyempurnaan fitur administrasi dan pelaporan. Pada *sprint* ini ditambahkan fitur ekspor laporan (mengunduh laporan absensi dan honor dalam format Excel/CSV) dan implementasi penuh *multi-user role* sesuai skenario: admin dapat mengelola seluruh data, sedangkan tutor memiliki akses terbatas untuk melihat jadwal mengajarnya, menginput absensi, dan memonitor rekap honor pribadinya. Otentikasi dan otorisasi ditingkatkan sehingga setiap menu dan aksi dibatasi sesuai peran.

d) Sprint 4

Pengujian menyeluruh, perbaikan *bug*, dan finalisasi fitur. Setelah semua komponen utama selesai, dilakukan *system testing* termasuk uji fungsional (*black-box*) dan *user acceptance test* sederhana dengan melibatkan calon pengguna (admin bimbel). Setiap *bug* atau kekurangan fungsional yang teridentifikasi diperbaiki pada *sprint* terakhir ini. Hasil akhir adalah prototipe sistem yang siap digunakan, lengkap dengan dokumentasi panduan penggunaan.

Teknologi yang digunakan selama pengembangan antara lain, *framework* Laravel (PHP) untuk pengembangan *backend*, Tailwind CSS untuk *frontend*, serta PostgreSQL sebagai basis data. Seluruh komponen di-*deploy* pada *platform hosting* web (Vercel) untuk keperluan uji coba *online*. Pemilihan *framework* Laravel yang dikombinasikan dengan metode *Agile* ini sejalan dengan penelitian Bintang *et al.* serta Naofal dan Daffa yang membuktikan bahwa pendekatan tersebut efektif dalam mendukung pengembangan layanan berbasis website karena fleksibel serta lebih terstruktur yang dapat dibuktikan melalui arsitektur MVC pada Laravel dan tahapan *sprint* yang sistematis [12][13].

4) Tahap Pengujian

Metode pengujian yang diterapkan adalah pengujian black-box terhadap fungsionalitas sistem, guna memastikan setiap fitur berjalan sesuai spesifikasi. Pendekatan ini dinilai tepat untuk memverifikasi logika aplikasi sistem informasi tanpa perlu membedah struktur kode internal, sejalan dengan penerapan pengujian fungsional [8][15]. Pengujian dilakukan dengan menyusun skenario uji untuk setiap use case utama, seperti skenario login, skenario *input* absensi, dan skenario ekspor laporan. Contoh skenario pengujian: "*Tutor melakukan input absensi untuk 10 siswa dengan 2 di antaranya berstatus alpa; sistem diharapkan*

hanya menghitung 8 siswa hadir dalam perhitungan honor", kasus ini diuji untuk memastikan siswa berstatus alpa benar-benar tidak terhitung dalam total honor tutor. Hasil setiap pengujian dicatat, apabila *output* aktual sesuai dengan *output* yang diharapkan, maka fungsionalitas dinyatakan lulus uji. Jika belum sesuai, maka dilakukan *debugging* dan perbaikan kode.

Selain pengujian *black-box*, dilakukan pula uji pengguna (*User Acceptance Test*) secara informal dengan melibatkan admin bimbingan untuk mencoba antarmuka sistem. Masukan pengguna mengenai kemudahan navigasi, kelengkapan informasi pada layar, dan kecepatan respon sistem dicatat dan dijadikan dasar perbaikan antarmuka pada iterasi akhir. Dengan kombinasi pengujian internal dan umpan balik pengguna, diyakini sistem yang dihasilkan memenuhi kebutuhan fungsional sekaligus mudah digunakan oleh target *user*.

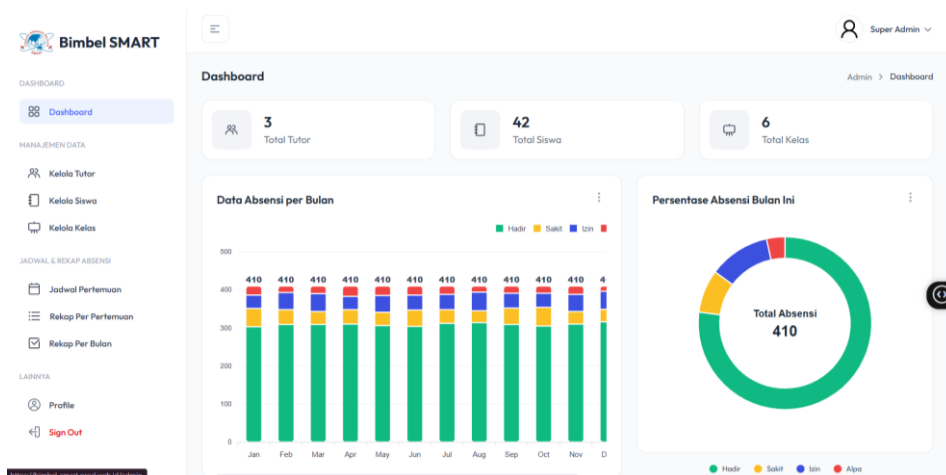
3. Hasil dan Pembahasan

3.1 Implementasi Sistem

Sistem absensi dan penghitungan fee Bimbel SMART dikembangkan dalam bentuk aplikasi berbasis web untuk mendukung proses administrasi pembelajaran dan pengelolaan honor tutor secara terkomputerisasi. Implementasi sistem dilakukan sesuai dengan hasil analisis kebutuhan yang telah didefinisikan pada bab sebelumnya. Pada bagian ini, implementasi difokuskan pada antarmuka utama yang merepresentasikan alur penggunaan sistem, yaitu halaman *login*, *dashboard* admin, halaman input absensi, dan halaman rekap absensi bulanan.

1) Dashboard Admin

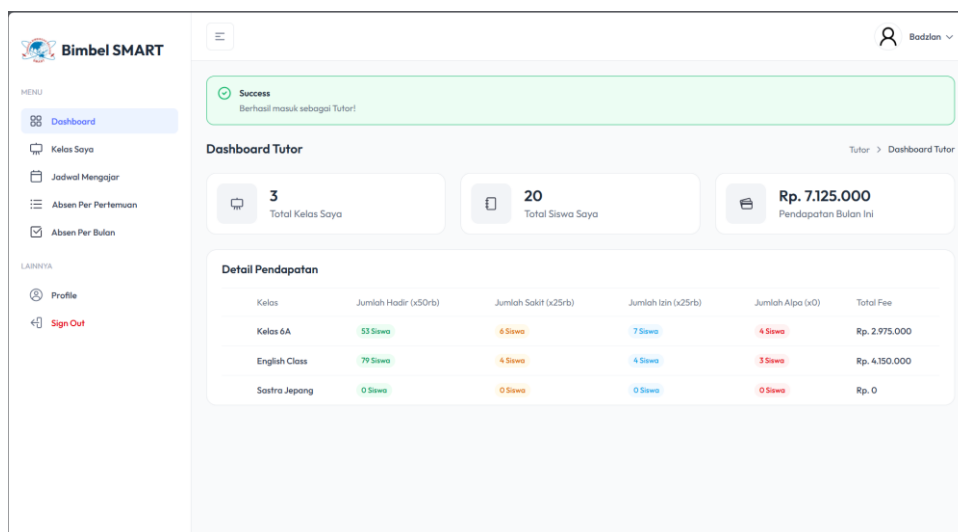
Gambar 6 menampilkan antarmuka *Dashboard* Admin pada sistem absensi dan fee Bimbel SMART. *Dashboard* ini menyajikan ringkasan informasi utama, meliputi jumlah tutor terdaftar, jumlah siswa aktif, dan kelas yang sedang berjalan. Selain itu, ditampilkan visualisasi data absensi siswa dalam bentuk grafik batang dan diagram lingkaran untuk memudahkan admin dalam memantau tingkat kehadiran siswa secara keseluruhan. Penyajian informasi dalam bentuk ringkasan dan grafik membantu admin dalam melakukan pemantauan dan pengambilan keputusan terkait pengelolaan kegiatan bimbingan belajar.



Gambar 6. Implementasi *Dashboard* Admin

2) Dashboard Tutor

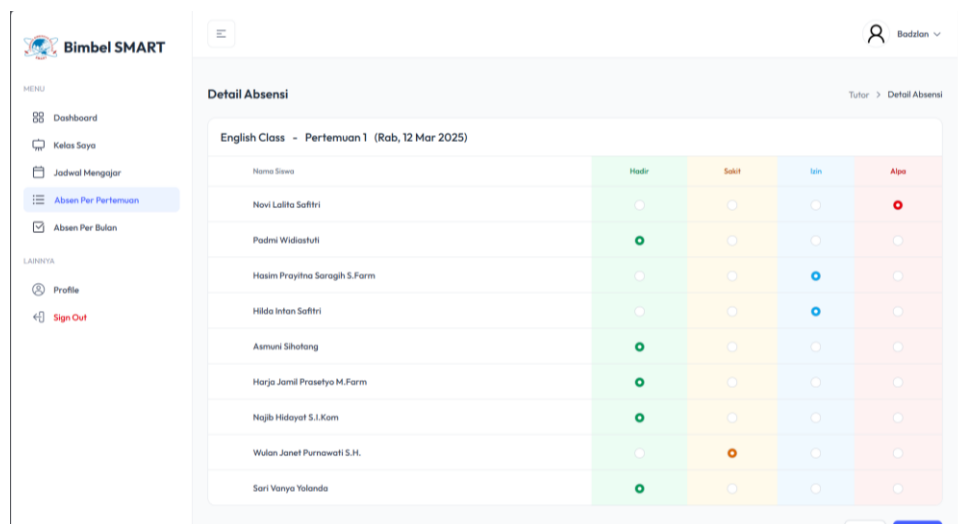
Gambar 7 menampilkan antarmuka *Dashboard* Tutor pada sistem absensi dan fee Bimbel SMART. *Dashboard* ini menyajikan ringkasan informasi berupa jumlah kelas yang diampu, total siswa yang diajar, serta pendapatan tutor pada periode berjalan yang dihitung secara otomatis berdasarkan data absensi. Selain itu, ditampilkan rincian pendapatan per kelas dalam bentuk tabel yang memuat informasi kehadiran siswa dan total honor yang diperoleh. Penyajian informasi tersebut memungkinkan tutor untuk memantau absensi dan pendapatan secara mandiri secara lebih transparan dan terstruktur.



Gambar 7. Implementasi Dashboard Tutor

3) Antarmuka Input Absensi Tutor

Gambar 8 menampilkan antarmuka Input Absensi Tutor yang berfungsi sebagai tahap input data kehadiran siswa pada setiap pertemuan kelas yang diampu. Pada tahap ini, sistem digunakan untuk mencatat data kehadiran siswa secara langsung melalui penentuan status Hadir, Izin, Sakit, atau Alfa. Data absensi yang diinput akan tersimpan secara otomatis ke dalam sistem dan menjadi data mentah yang selanjutnya diproses pada tahap rekapitulasi absensi sebagai dasar perhitungan fee tutor. Dalam perhitungan tersebut, status Alfa tidak diperhitungkan, sedangkan status Hadir, Izin, dan Sakit tetap dihitung sesuai dengan aturan yang berlaku.

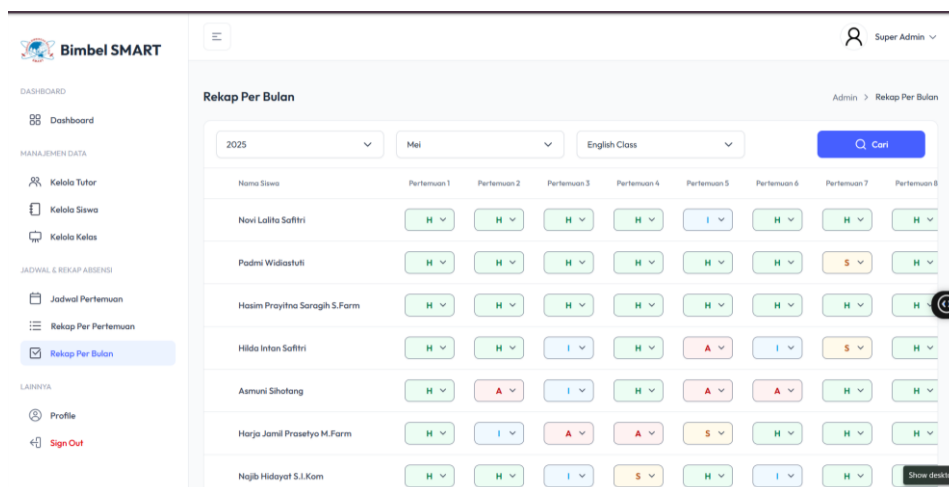


Gambar 8. Antarmuka Input Absensi Tutor

4) Antarmuka Rekap Absensi Bulanan

Antarmuka Rekap Absensi Bulanan merupakan tahap dari proses pengolahan data yang berfungsi untuk mengolah dan mengakumulasi data absensi yang telah diinput pada setiap pertemuan kelas. Admin dapat menampilkan hasil pengolahan data absensi berdasarkan periode tertentu, meliputi tahun, bulan, dan kelas. Data kehadiran siswa disajikan dalam bentuk tabel dengan status Hadir, Sakit, Izin, dan Alfa untuk setiap pertemuan. Hasil rekap absensi ini digunakan sebagai dasar perhitungan fee tutor sesuai

dengan aturan yang berlaku, di mana status Alfa tidak diperhitungkan. Dengan demikian, sistem mampu mendukung proses administrasi absensi dan perhitungan fee secara lebih akurat dan terstruktur.



Gambar 9. Antarmuka Rekap Absensi Bulanan

3.2 Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode *black-box testing* untuk memastikan bahwa seluruh fitur fungsional sistem berjalan sesuai dengan kebutuhan yang telah ditetapkan. Pengujian difokuskan pada fungsi utama sistem, meliputi input absensi, rekap absensi bulanan, perhitungan fee tutor, serta kontrol hak akses antara admin dan tutor. Hasil pengujian fungsionalitas sistem disajikan pada Tabel 2.

Tabel 2. Hasil Pengujian Sistem (*Black-box Testing*)

No	Fitur yang Diuji	Skenario Pengujian	Input	Output yang Diharapkan	Hasil
1	Input Absensi	Input status Hadir	Pilih status Hadir lalu Simpan	Data tersimpan, rekap bertambah	Sesuai
2	Input Absensi	Input status Izin	Pilih status Izin lalu Simpan	Data tersimpan, rekap bertambah	Sesuai
3	Input Absensi	Input status Sakit	Pilih status Sakit lalu Simpan	Data tersimpan, rekap bertambah	Sesuai
4	Input Absensi	Input status Alfa	Pilih status Alfa lalu Simpan	Data tersimpan, tidak dihitung pada akumulasi	Sesuai
5	Rekap Absensi Bulanan	Menampilkan rekap sesuai filter	Pilih tahun-bulan-kelas lalu cari	Rekap tampil sesuai periode yang dipilih	Sesuai
6	Perhitungan Honor Otomatis	Alfa tidak dihitung	Data absensi mengandung Alfa	Nilai honor/akumulasi tidak bertambah untuk Alfa	Sesuai
7	Hak Akses	Tutor akses menu/URL admin	Login tutor: akses halaman admin	Sistem menolak akses/redirect	Sesuai
8	Hak Akses	Admin akses menu admin	Login admin: akses fitur admin	Sistem memberikan akses	Sesuai

Berdasarkan hasil pengujian yang disajikan pada Tabel 2, seluruh skenario pengujian menunjukkan keluaran yang sesuai dengan hasil yang diharapkan. Hal ini menunjukkan bahwa fungsi-fungsi utama sistem telah berjalan dengan baik dan sesuai dengan kebutuhan fungsional yang telah didefinisikan pada tahap analisis kebutuhan.

3.3. Pembahasan

Hasil implementasi dan pengujian menunjukkan bahwa sistem absensi dan penghitungan fee tutor berbasis web yang dikembangkan telah mampu menjawab permasalahan utama yang diidentifikasi pada awal penelitian, yaitu inefisiensi dan kerentanan kesalahan pada proses rekapitulasi manual di Bimbel SMART. Melalui digitalisasi, pencatatan kehadiran tutor yang semula bersifat konvensional kini terintegrasi dalam basis data terpusat, sehingga risiko kesalahan input dapat diminimalisir dan validitas data administratif meningkat. Hal ini sejalan dengan temuan Putra et al. yang menegaskan bahwa transisi ke sistem absensi berbasis website efektif dalam meningkatkan disiplin serta mempercepat proses pelaporan kehadiran secara *real-time* [7].

Penerapan aturan bisnis spesifik, seperti logika kalkulasi honorarium yang secara otomatis mengecualikan status ketidakhadiran (alpha), membuktikan akurasi sistem dalam menjalankan kebijakan institusi secara konsisten. Otomatisasi ini menghilangkan perbedaan interpretasi perhitungan antara tutor dan manajemen karena seluruh pihak mengacu pada data yang tervalidasi oleh sistem. Temuan ini memperkuat penelitian Ansanay et al. yang melaporkan bahwa integrasi modul absensi dengan kalkulasi honorarium secara langsung mampu mengurangi penundaan pelaporan dan meningkatkan transparansi bagi seluruh pemilik kepentingan [4]. Selain itu, penyediaan *dashboard* interaktif bagi admin dan tutor memberikan nilai tambah melalui visualisasi data yang informatif, sebuah aspek yang sebelumnya menjadi keterbatasan pada penelitian Wijoyo et al. [6].

Selain aspek visualisasi data, penerapan *Role-Based Access Control (RBAC)* dalam sistem ini berperan penting dalam menjaga keamanan dan integritas data administratif. Pembatasan hak akses berdasarkan peran memastikan bahwa admin memiliki kewenangan penuh dalam pengelolaan data dan rekapitulasi fee, sementara tutor hanya dapat mengakses informasi yang berkaitan dengan jadwal, absensi, dan honor yang diperoleh. Mekanisme ini meminimalkan potensi manipulasi data serta meningkatkan akuntabilitas sistem, sejalan dengan temuan Ansanay et al. yang menekankan pentingnya kontrol akses bertingkat dalam sistem pengelolaan honor berbasis web [4].

Dari sisi pendekatan pengembangan, penerapan metode *Agile Development* memungkinkan sistem dikembangkan secara iteratif dan adaptif terhadap kebutuhan operasional bimbingan belajar. Proses pengembangan yang dibagi ke dalam beberapa sprint memberikan ruang bagi evaluasi dan perbaikan fitur berdasarkan masukan pengguna, khususnya pada tahap *User Acceptance Test*. Hasil pengujian fungsional (*black-box testing*) menunjukkan bahwa seluruh fitur utama, seperti input absensi, perhitungan fee otomatis, serta validasi hak akses, berjalan sesuai dengan kebutuhan yang telah ditetapkan. Hal ini mendukung temuan Bintang et al. yang menyatakan bahwa pendekatan Agile efektif dalam menghasilkan sistem berbasis web yang responsif terhadap kebutuhan pengguna [12].

Secara keseluruhan, integrasi sistem absensi dan penghitungan fee tutor dalam satu platform berbasis web terbukti mampu meningkatkan keteraturan proses administrasi, konsistensi perhitungan honor, serta transparansi informasi bagi tutor dan pengelola Bimbel SMART. Sistem ini tidak hanya menggantikan proses manual yang rawan kesalahan, tetapi juga menyediakan mekanisme kontrol dan monitoring yang lebih terstruktur, sehingga mendukung pengambilan keputusan berbasis data.

Namun demikian, penelitian ini masih memiliki beberapa keterbatasan. Sistem absensi yang dikembangkan masih bergantung pada input manual oleh tutor, sehingga potensi ketidaksesuaian data kehadiran akibat praktik *self-reporting* masih dimungkinkan. Selain itu, pengujian sistem masih terbatas pada pengujian fungsional, sehingga aspek non-fungsional seperti performa sistem dan keamanan data belum dievaluasi secara mendalam.

Berdasarkan keterbatasan tersebut, penelitian selanjutnya direkomendasikan untuk mengintegrasikan mekanisme validasi kehadiran yang lebih kuat, seperti *geolocation* atau *biometrik*, serta memperluas cakupan pengujian hingga mencakup pengujian performa dan

keamanan sistem. Pengembangan lanjutan ini diharapkan dapat meningkatkan reliabilitas dan skalabilitas sistem sehingga dapat diadaptasi pada lembaga pendidikan atau bimbingan belajar lain yang memiliki permasalahan administrasi serupa.

5. Simpulan

Penelitian ini menyimpulkan bahwa pengembangan sistem absensi dan perhitungan honorarium berbasis web menggunakan metode Agile berhasil mengatasi permasalahan inefisiensi dan kesalahan rekapitulasi manual di Bimbel SMART. Berdasarkan hasil pengujian *black-box*, sistem terbukti mampu menjalankan fungsi vital secara akurat, khususnya dalam memvalidasi logika perhitungan honor otomatis yang secara tepat mengecualikan status ketidakhadiran (alpa) serta menegakkan keamanan data melalui pembatasan hak akses antara admin dan tutor. Keberhasilan pengujian skenario fungsional ini mengonfirmasi bahwa sistem telah memenuhi standar kebutuhan operasional untuk menjamin transparansi dan validitas data keuangan. Sebagai upaya pengembangan berkelanjutan, sistem ini direkomendasikan untuk dilengkapi dengan fitur notifikasi otomatis ke perangkat seluler pengguna dan integrasi modul keuangan yang lebih kompleks guna mendukung skalabilitas layanan pendidikan di masa depan.

Daftar Referensi

- [1] K. Keivin and J. Shadiq, "Sistem Informasi Penggajian Guru Berbasis Website Pada SMK Tiara Bangsa Bekasi," *Jurnal Mahasiswa Bina Insani*, vol. 4, no. 2, pp. 205–214, 2020.
- [2] T. Triyono, R. Safitri, and T. Gunawan, "Perancangan Sistem Informasi Absensi Guru dan Staff pada SMK Pancakarya Tangerang Berbasis Web," *SENSI Journal*, vol. 4, no. 2, pp. 153–167, Aug. 2018, doi: 10.33050/sensi.v4i2.638.
- [3] A. Pulungan and A. Saleh, "Perancangan Aplikasi Absensi Menggunakan QR Code Berbasis Android," *Jurnal Mahasiswa Fakultas Teknik dan Ilmu Komputer*, vol. 1, no. 1, pp. 1063–1074, 2020.
- [4] N. B. Ansanay and E. L. Tatuhey, "Web-Based Honorarium Management Application for Contract Teachers at SMK Pelayaran Ampari Jayapura," *Jurnal Ilmiah Sistem Informasi*, vol. 4, no. 3, pp. 394–416, Aug. 2025, doi: 10.51903/ehxthp91.
- [5] I. F. Yani, "Sistem Informasi Penggajian Guru TPQ/TPSQ Kota Padang Panjang Berbasis Web," Skripsi, IAIN Batusangkar, 2022.
- [6] H. Wijoyo, "Rancang Bangun Sistem Informasi Penggajian Dan Absensi Karyawan Megara Hotel Pekanbaru Berbasis Web," *Ekonomi: Jurnal Ekonomi, Akuntansi & Manajemen*, vol. 2, no. 2, pp. 56–76, Oct. 2020, doi: 10.37577/ekonomi.v2i2.286.
- [7] H. Putra, D. Y. Wulandari, and A. I. B. Siregar, "Sistem Informasi Absensi Dalam Meningkatkan Disiplin Karyawan Dan Staf Pengajar Berbasis Website," *Jurnal Komputer Teknologi Informasi dan Sistem Informasi (JUKTISI)*, vol. 3, no. 2, pp. 823–830, Sep. 2024, doi: 10.62712/juktisi.v3i2.265.
- [8] M. Nasir, S. W. Nurjihan, Nurbadillah, N. Faturrahman, I. M. Wiguna, E. M. Lasardi, and A. Wicaksono, "Analisis Decision Table Testing untuk Pengujian Blackbox Website Pusat Studi Bencana IPB," *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 11, no. 4, pp. 440–448, Dec. 2024.
- [9] K. Jaya, I. Maliki, G. C. Lestari, D. Irawan, and A. Wasid, "Perancangan Sistem Informasi Berbasis Web pada Bimbingan Belajar (BIMBEL) 'BNF,'" *Jurnal Informatika dan Komputasi: Media Bahasan, Analisa dan Aplikasi*, vol. 18, no. 2, pp. 72–85, Dec. 2024, doi: 10.56956/jiki.v18i2.388.
- [10] O. T. Septiawan, "Sistem Informasi Bimbingan Belajar PPA Berbasis Web," Skripsi, Universitas Negeri Yogyakarta, 2018.
- [11] D. A. Ramadhan, F. Ardiansyah, J. Adisantoso, A. Asfarian, and Y. Nurhadryani, "Investigasi Awal Penggunaan Layanan Digital Perguruan Tinggi Studi Kasus: IPB Mobile for Students," *Jurnal Ilmu Komputer dan Agri-Informatika*, vol. 9, no. 1, pp. 37–46, 2022.
- [12] S. Bintang, R. W. Ningrata, J. C. Rabika, M. A. Al Ataya, A. Wicaksono, and G. P. Mindra, "Pengembangan Website WardellTech dengan Agile Scrum dan Laravel untuk Mendukung Layanan Freelance," *JITU: Journal Informatic Technology and Communication*, vol. 9, no. 2, pp. 180–191, Nov. 2025.
- [13] N. Naofal dan M. R. Daffa, "Implementasi Framework Laravel pada Sistem Informasi Bimbingan Belajar (Studi Kasus: Glow Math Course)," *Sistemasi: Jurnal Sistem Informasi*, vol. 14, no. 3, pp. 1406–1417, 2025.

-
- [14] N. A. Atallah, "Penggunaan Metode Agile Scrum Pada Perancangan Sistem Informasi Surat Izin Penelitian di BAKESBANGPOL Lombok Tengah," *Jurnal Neptunus*, vol. 2, no. 3, pp. 371–384, Jul. 2024.
- [15] A. Jailani dan M. A. Yaqin, "Pengujian Aplikasi Sistem Informasi Akademik menggunakan Metode Blackbox dengan Teknik Boundary Value Analysis," *JACIS: Journal Automation Computer Information System*, vol. 4, no. 2, pp. 60–66, Nov. 2024.