

Pengembangan Aplikasi Balanced Scorecard untuk Pengelolaan Key Performance Indicator Perusahaan

Risma Nur'Aini¹, Allsela Meiriza^{2*}, Dinna Yunika Hardiyanti³, Khoirusy Syafaat⁴

^{1,2,3}Sistem Informasi, Universitas Sriwijaya, Indonesia

⁴Universitas Indo Global Mandiri, Indonesia

¹rismanuraini0510@gmail.com, ²allsela@unsri.ac.id, ³dinna.yunika@unsri.ac.id, ⁴khoirusysyafaat@students.uigm.ac.id

Info Artikel

Riwayat Artikel:

Received 2026-01-05

Revised 2026-01-29

Accepted 2026-02-06

Abstract – Key Performance Indicator (KPI) management plays a strategic role in evaluating organizational performance in a systematic and sustainable manner. However, KPI management processes that rely on manual or semi-digital approaches often lead to verification delays, data fragmentation, and limited accuracy in performance reporting. This study aims to develop a Balanced Scorecard application to support structured and digital-based KPI management. The system was developed using the Extreme Programming with the stages of planning, design, coding, and testing. System requirements were identified through observation and stakeholder discussions, while system design employed Unified Modeling Language (UML). The implementation utilized PHP and MySQL, with HTML and CSS for the user interface. System testing was conducted using black box testing to validate functional requirements. The results indicate that the developed Balanced Scorecard application effectively facilitates KPI input, verification, approval, and reporting processes in a more integrated and efficient manner. The system contributes to improving organizational performance management and supports timely and data-driven managerial decision-making.

Keywords: Balanced Scorecard; Extreme Programming; KPI; Information System

Corresponding Author:

Risma Nur'Aini

Email:rismanuraini0510@gmail.com



This is an open access article under the [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/) license.

Abstrak – Pengelolaan Key Performance Indicator (KPI) merupakan aspek penting dalam mengukur dan mengevaluasi kinerja organisasi secara sistematis dan berkelanjutan. Namun, proses pengelolaan KPI yang masih dilakukan secara manual sehingga berpotensi menimbulkan permasalahan, seperti keterlambatan verifikasi, ketidakterpaduan data, serta rendahnya akurasi pelaporan kinerja. Penelitian ini bertujuan untuk mengembangkan aplikasi Balanced Scorecard guna mendukung pengelolaan KPI secara terintegrasi dan berbasis digital. Metode pengembangan yang digunakan adalah Extreme Programming dengan tahapan planning, design, coding, dan testing. Pada tahap perencanaan dilakukan analisis kebutuhan sistem melalui observasi dan diskusi dengan pihak terkait. Tahap perancangan memanfaatkan Unified Modeling Language (UML) untuk memodelkan sistem dan merancang basis data serta antarmuka pengguna. Implementasi sistem dilakukan menggunakan bahasa pemrograman PHP dan basis data MySQL, dengan antarmuka berbasis HTML dan CSS. Pengujian sistem dilakukan menggunakan metode black box untuk memastikan seluruh fungsi berjalan sesuai dengan kebutuhan pengguna. Hasil penelitian menunjukkan bahwa aplikasi Balanced Scorecard yang dikembangkan mampu memfasilitasi proses input, verifikasi, persetujuan, dan pelaporan KPI secara lebih sistematis, terintegrasi dan terstruktur. Dengan demikian, sistem ini diharapkan dapat meningkatkan kualitas pengelolaan kinerja organisasi serta mendukung pengambilan keputusan manajerial secara tepat dan akurat.

Kata Kunci: Balanced Scorecard; Extreme Programming; KPI; Sistem Informasi

I. PENDAHULUAN

Pengelolaan kinerja merupakan proses pencatatan, pengukuran, dan evaluasi capaian kinerja yang bertujuan untuk memantau serta mendukung pencapaian tujuan strategis organisasi [1]. Setiap organisasi yang membutuhkan informasi terkait pencapaian target dan indikator kinerja tentunya akan melibatkan proses pengelolaan *Key Performance Indicator* (KPI) sebagai alat ukur kinerja organisasi [2]. Data kinerja ini dirancang dan dikelola sedemikian rupa agar mudah diakses, dianalisis, dan dimanfaatkan oleh manajemen dalam pengambilan keputusan strategis. Proses pengelolaan kinerja ini juga berlaku dalam lingkungan korporasi, dimana informasi KPI digunakan oleh pimpinan dan manajemen sebagai dasar evaluasi dan perumusan strategi organisasi [3].

Berdasarkan penelitian yang dilakukan oleh Nashihuddin dan Mubin berjudul “Penerapan Konsep *Balanced Scorecard* dan *Objective Matrix* Untuk Perancangan Sistem Pengukuran Kinerja Perusahaan” dalam penelitiannya menunjukkan bahwa sistem pengukuran kinerja yang masih dilakukan secara manual sehingga belum mampu menerjemahkan strategi perusahaan ke dalam tujuan strategis dan indikator kinerja secara terstruktur [4]. Penelitian lain yang dilakukan oleh Ali Mukhtar, Agung Setiawan, dan Djoko Adi Walujo yang berjudul “*Applied of Balanced Scorecard (BSC) Approach with Determination and Weighting of Key*

Performance Indicator (KPI) for Employee Performance Measurement” menunjukkan bahwa penilaian kinerja karyawan masih dilakukan secara konvensional tanpa dukungan sistem terintegrasi [5]. Kondisi ini menyebabkan penentuan dan pembobotan KPI menjadi kurang objektif serta meningkatkan risiko kesalahan dalam proses evaluasi kinerja. Selain itu, penelitian dari Sulhan dan Alek pada yang berjudul “Penerapan Metode *Balanced Scorecard* untuk Penentuan *Key Performance Indicator* di Departemen *Production Engineering* pada PT.XYZ” menunjukkan bahwa penentuan KPI pada Departemen *Production Engineering* masih dilakukan secara manual dan belum terintegrasi dengan sistem aplikasi perusahaan [6].

Permasalahan tersebut menunjukkan bahwa pengelolaan KPI yang belum terdigitalisasi secara optimal dapat menyebabkan ketidakkonsistenan data, keterlambatan proses verifikasi, serta kurang efisiennya penyusunan laporan kinerja. Proses evaluasi yang memerlukan waktu relatif lama juga berdampak pada lambatnya pengambilan keputusan manajerial. Kondisi ini menegaskan perlunya pengembangan aplikasi *Balanced Scorecard* yang mampu mendukung pengelolaan KPI perusahaan secara sistematis, terstruktur, dan terintegrasi [7].

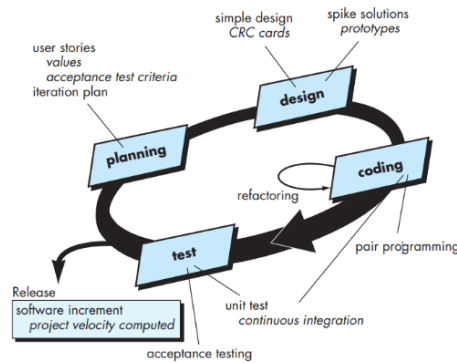
Meskipun penelitian terkait penerapan *Balanced Scorecard* dan KPI telah banyak dilakukan, sebagian besar penelitian tersebut masih berfokus pada perancangan konsep sistem pengukuran kinerja atau penentuan indikator kinerja tanpa mengembangkan aplikasi yang mampu mengelola seluruh siklus KPI secara terintegrasi. Selain itu, penelitian sebelumnya juga masih terbatas dalam mengakomodasi proses operasional pengelolaan KPI secara menyeluruh, mulai dari proses penetapan KPI, proses verifikasi berjenjang, pengelolaan dokumen realisasi, hingga penyajian laporan kinerja dalam satu sistem yang terintegrasi. Penggunaan metode pengembangan perangkat lunak yang adaptif terhadap perubahan kebutuhan pengguna seperti *Extreme Programming* juga masih jarang diterapkan dalam penelitian yang berkaitan dengan pengembangan aplikasi untuk pengelolaan KPI.

Berdasarkan kondisi tersebut, penelitian ini bertujuan untuk mengembangkan aplikasi *Balanced Scorecard* berbasis web yang dapat mendukung pengelolaan *Key Performance Indicator* perusahaan secara lebih sistematis dan terstruktur. Aplikasi yang dikembangkan dirancang untuk memfasilitasi proses penetapan KPI oleh admin, verifikasi oleh leader group, persetujuan oleh pimpinan, pengunggahan dokumen realisasi, serta penyajian laporan kinerja secara terintegrasi. Pengembangan aplikasi dilakukan dengan menggunakan metode *Extreme Programming* sebagai pendekatan pengembangan perangkat lunak yang bersifat fleksibel dan mampu menyesuaikan perubahan kebutuhan sistem selama proses pengembangan. Melalui pengembangan aplikasi tersebut, penelitian ini diharapkan dapat memberikan kontribusi dalam menyediakan solusi implementatif bagi organisasi untuk meningkatkan efektivitas pengelolaan KPI, mempercepat proses verifikasi kinerja, serta mendukung pengambilan keputusan manajerial berbasis data secara lebih akurat dan tepat waktu.

II. METODE

Penelitian ini menggunakan metode *Extreme Programming* dalam pengembangan aplikasi *Balanced Scorecard* untuk pengelolaan *Key Performance Indicator* perusahaan. Metode ini dipilih karena mendukung proses pengembangan aplikasi yang bersifat terbuka terhadap perubahan kebutuhan sistem selama proses pengembangan [8]. Berbeda dengan metode pengembangan linier yang kaku dan sulit menyesuaikan perubahan kebutuhan [9]. Secara metodologis, metode *Extreme Programming* paling sesuai untuk pengembangan sistem pengelolaan kinerja yang bersifat dinamis serta melibatkan beberapa peran pengguna seperti admin, leader group dan pimpinan.

Tahapan pada metode *Extreme Programming* mulai dari *planning* (perencanaan), *design* (perancangan), *coding* (pengkodean), *testing* (pengujian). Pendekatan ini mendukung proses perbaikan serta integrasi masukkan pengguna secara berkelanjutan pada setiap tahap pengembangan [10]. Untuk evaluasi, penelitian ini membandingkan waktu verifikasi KPI sebelum dan sesudah aplikasi *Balanced Scorecard* diterapkan, untuk menilai sejauh mana pengelolaan KPI menjadi lebih tertata dan proses verifikasi lebih cepat. Gambar 1 menunjukkan tahapan-tahapan pada metode *Extreme Programming*:



Gambar 1. Metode *Extreme Programming* [11]

A. *Planning* (Perencanaan)

Tahap *planning* bertujuan untuk memahami kebutuhan sistem pengelolaan *Key Performance Indicator* (KPI) pada aplikasi *Balanced Scorecard*. Pada tahap ini, peneliti melakukan analisis terhadap proses pengelolaan KPI yang sedang berjalan melalui wawancara dengan pihak terkait untuk memperoleh gambaran mengenai kebutuhan pengguna serta alur proses bisnis yang ada.

Hasil dari tahap *planning* berupa perumusan kebutuhan sistem yang akan digunakan sebagai acuan dalam pengembangan aplikasi [12]. Berdasarkan hasil analisis tersebut, diperoleh beberapa kebutuhan fungsional sistem yang meliputi fitur login pengguna, penginputan dan pengelolaan data KPI, proses verifikasi dan persetujuan KPI, pengunggahan dokumen realisasi, serta fitur pencetakan laporan KPI. Selain itu, ditentukan pula kebutuhan non-fungsional sistem, seperti kemudahan penggunaan aplikasi, pengaturan hak akses pengguna untuk menjaga keamanan data, serta kecepatan sistem dalam menampilkan informasi kinerja [13].

B. *Design* (Perancangan)

Pada tahap *design*, sistem dimodelkan berdasarkan hasil analisis kebutuhan yang telah dikumpulkan pada tahap sebelumnya. Tahap ini bertujuan untuk menggambarkan struktur dan alur kerja sistem sebelum proses pengembangan dilakukan. Perancangan sistem meliputi perancangan alur sistem dan perancangan desain antarmuka. Pemodelan sistem pada tahap *design* yaitu menggunakan *Unified Modeling Language* (UML) yang terdiri dari beberapa diagram sistem seperti *use case diagram* dan *class diagram* [14].

C. *Coding* (Pengkodean)

Tahap *coding* merupakan tahap implementasi dari hasil perancangan aplikasi *Balanced Scorecard* ke dalam bentuk aplikasi berbasis website. Proses pengembangan dilakukan menggunakan bahasa pemrograman PHP untuk mengelola logika sistem, MySQL sebagai basis data sertaku HTML dan CSS untuk menghasilkan tampilan yang responsif dan mudah digunakan [15]. Pada tahap ini diimplementasikan fitur-fitur utama dalam aplikasi seperti login, pengelolaan data KPI, verifikasi dan persetujuan KPI, pengunggahan dokumen realisasi, serta pembuatan laporan KPI.

D. *Testing* (Pengujian)

Pengujian merupakan proses untuk memastikan bahwa suatu aplikasi *Balanced Scorecard* telah dikembangkan sesuai dengan hasil analisis dan perancangan sistem. Tahap ini bertujuan untuk mengevaluasi fungsionalitas, keandalan, dan kinerja aplikasi guna memastikan bahwa setiap komponen bekerja sempurna [16]. Pengujian yang dilakukan adalah dengan menggunakan pengujian *black box* untuk mengetahui kualitas perangkat lunak yang telah dikembangkan. Pengujian ini memberikan beberapa skenario *input* dan *output* yang diharapkan. Selanjutnya, akan dilakukan uji validasi apakah *input* dan *output* yang dihasilkan sudah sesuai dengan yang diharapkan [17]. Selain pengujian fungsional, evaluasi kinerja aplikasi dilakukan dengan membandingkan waktu proses verifikasi KPI sebelum dan sesudah penerapan aplikasi *Balanced Scorecard*. Hasil evaluasi menunjukkan adanya peningkatan efisiensi proses verifikasi dan pelaporan KPI setelah sistem diimplementasikan.

III. HASIL DAN PEMBAHASAN

A. Planning (Perencanaan)

Tahap planning dilakukan untuk memahami secara mendalam kebutuhan sistem pengelolaan *Key Performance Indicator* (KPI) pada aplikasi *Balanced Scorecard* yang akan dikembangkan. Analisis difokuskan pada alur input KPI oleh admin, proses verifikasi oleh leader group, persetujuan akhir oleh pimpinan, dan pengelolaan dokumen realisasi KPI.

Hasil analisis menunjukkan bahwa proses pengelolaan KPI saat ini cukup terstruktur, namun terdapat beberapa kendala, seperti keterlambatan verifikasi dan kurangnya kemudahan akses bagi pengguna yang memiliki peran berbeda. Permasalahan tersebut menjadi dasar dalam perumusan kebutuhan sistem sebagai acuan dalam pengembangan aplikasi *Balanced Scorecard* untuk pengelolaan KPI perusahaan. Kebutuhan sistem fungsional dan kebutuhan non- fungsional. Berdasarkan hasil perencanaan, kebutuhan sistem dirumuskan menjadi dua kategori yaitu kebutuhan fungsional dan kebutuhan non-fungsional.

1. Kebutuhan Fungsional

Kebutuhan fungsional pada sistem ini dirumuskan untuk mendukung proses pengelolaan KPI dan disusun berdasarkan peran masing-masing pengguna. Pada level Admin, sistem menyediakan fungsi *login* dan *logout* kedalam sistem, penginputan KPI, pengeditan KPI, akses detail KPI, serta pengelolaan data KPI. Pada level Leader Group, sistem menyediakan fungsi *login* dan *logout*, akses detail KPI yang ditugaskan, verifikasi KPI, pengembalian KPI untuk di revisi admin, serta pengunggahan dokumen realisasi KPI. Pada level Pimpinan, sistem menyediakan fungsi *login* dan *logout*, akses detail KPI, persetujuan akhi KPI, evaluasi dokumen realisasi, serta *ekspor* laporan KPI.

2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan tambahan yang tidak memiliki input, proses dan output. Kebutuhan non- fungsional dapat dikategorikan berdasarkan kerangka kerja PIECES (*Performance, Information, Economics, Control, Efficiency, dan Service*). Berikut ini merupakan kebutuhan non- fungsional dari sistem berdasarkan kerangka kerja PIECES:

TABEL 1
KEBUTUHAN NON-FUNGSIONAL

No. NF-SRS	Kebutuhan Non-Fungsional	Penjelasan
NF-SRS-01	<i>Performance</i>	Sistem harus dapat diakses setiap saat selama tersedia koneksi internet yang stabil.
NF-SRS-02	<i>Information</i>	Sistem mampu menyajikan informasi terkait KPI yang akurat dan dapat diakses sesuai hak pengguna.
NF-SRS-03	<i>Economics</i>	Sistem dirancang untuk mengurangi beban kerja manual dalam pengelolaan KPI.
NF-SRS-04	<i>Control</i>	Sistem harus memiliki mekanisme keamanan berupa autentikasi dan pengaturan hak akses sesuai dengan peran pengguna.
NF-SRS-05	<i>Efficiency</i>	Sistem diharapkan dapat mempercepat proses input, verifikasi, approve, dan pelaporan sekaligus mengurangi duplikasi data.
NF-SRS-06	<i>Service</i>	Sistem harus memiliki antarmuka yang mudah digunakan oleh semua peran.

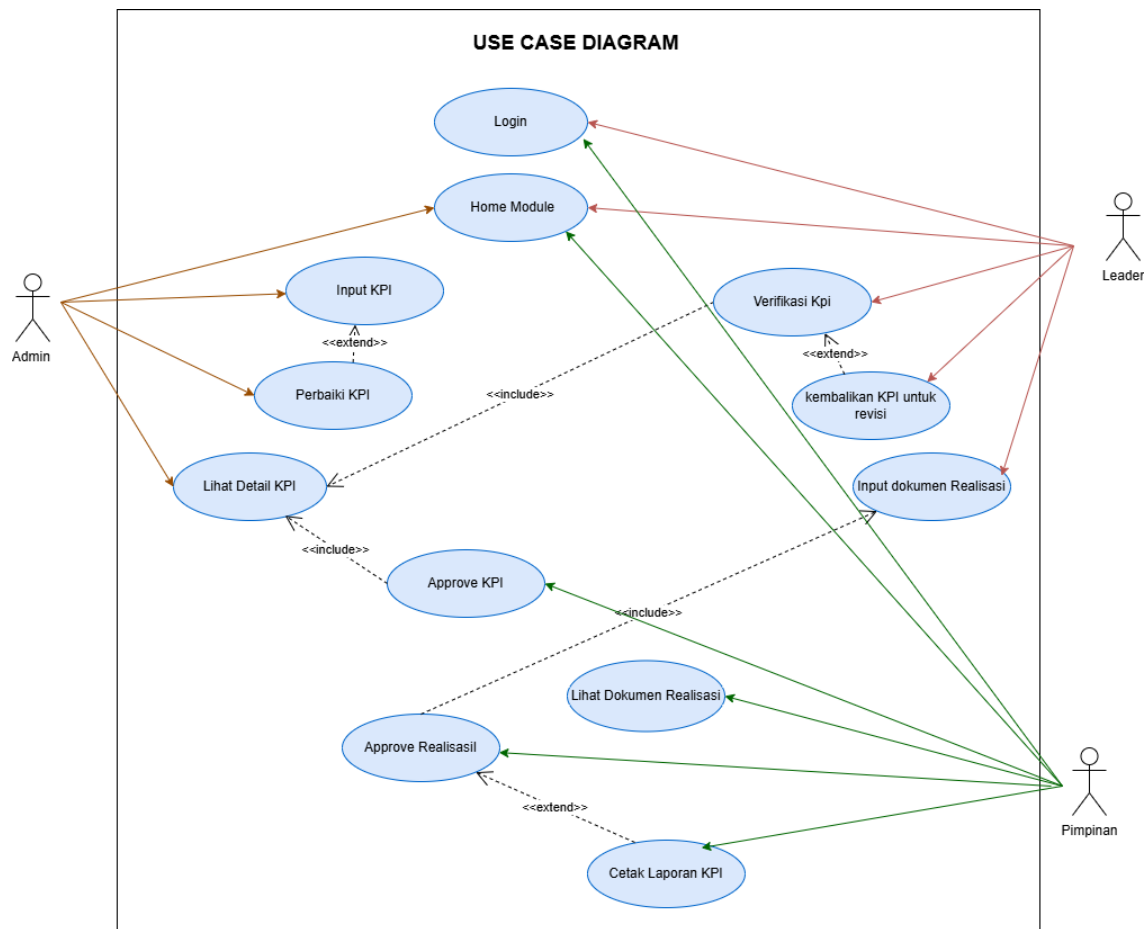
B. Design (Perancangan Design)

Tahapan ini dilakukan perancangan aplikasi menggunakan *Unified Modeling Language* (UML). UML digunakan untuk memvisualisasikan struktur dan alur kerja sistem, seperti use case diagram guna memahami interaksi antar komponen.

1. Use Case Diagram

Use case diagram adalah salah satu diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara pengguna (aktor) dan aplikasi. Diagram ini membantu

menjelaskan cara kerja aplikasi dari perspektif pengguna, termasuk fitur yang dapat diakses dan bagaimana mereka berinteraksi dengan aplikasi. Berikut merupakan rancangan use case diagramnya:

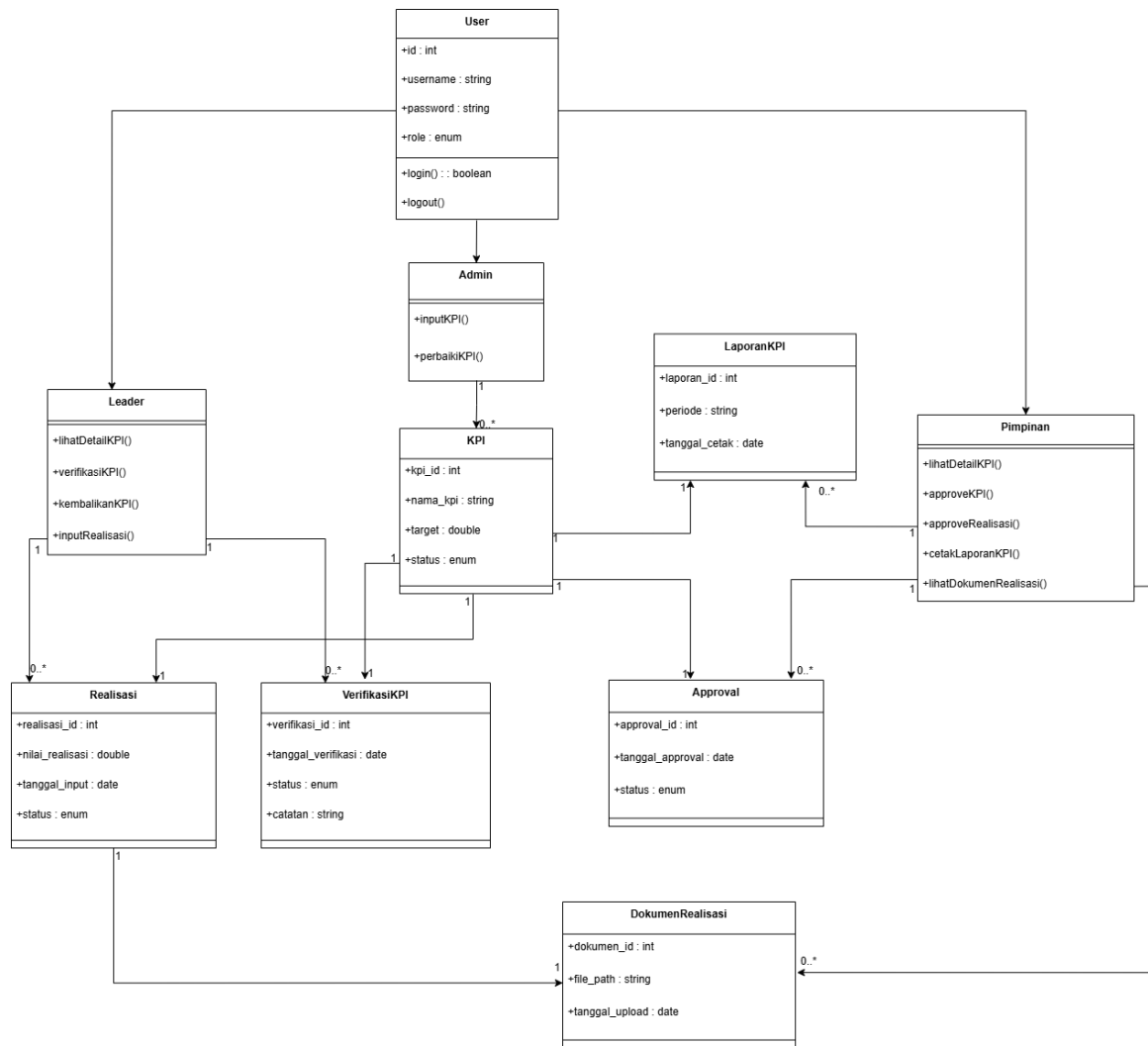


Gambar 2. Use Case Diagram Aplikasi BSC

Use case diagram tersebut menggambarkan alur kerja aplikasi pengelolaan KPI yang melibatkan tiga aktor, yaitu Admin, Leader, dan Pimpinan. Admin berperan dalam menginput, memperbaiki, melihat detail, dan menyetujui KPI yang telah dibuat. Leader group bertugas memverifikasi KPI yang diajukan oleh Admin, mengembalikan KPI untuk revisi jika belum sesuai, serta menginput dokumen realisasi KPI. Selanjutnya, Pimpinan berwenang melihat dokumen realisasi, menyetujui realisasi KPI, dan mencetak laporan KPI sebagai hasil akhir. Diagram ini menunjukkan proses KPI yang terstruktur mulai dari input data, verifikasi, realisasi, hingga pelaporan.

2. Class Diagram

Class diagram adalah sebuah diagram yang menunjukkan hubungan secara logika antar kelas dalam suatu sistem. Berikut adalah class diagram yang digunakan dalam sistem informasi KPI, yang menggambarkan struktur kelas, atribut, metode, serta relasi antar kelas. Diagram ini terdiri dari kelas User sebagai kelas utama yang memiliki turunan admin, leader, dan pimpinan sesuai dengan peran pengguna. Selain itu, terdapat kelas KPI yang berelasi dengan kelas verifikasi KPI, realisasi, approval, dokumen realisasi, dan laporan KPI untuk mendukung proses pengelolaan KPI dari input, verifikasi, realisasi, persetujuan, hingga pelaporan.



Gambar 3. Class Diagram Aplikasi BSC

C. Coding (Pengkodean)

Pada tahap ini, desain aplikasi yang telah dibuat pada tahap sebelumnya diubah menjadi kode program yang berfungsi penuh sesuai kebutuhan pengguna. Proses pengembangan dilakukan menggunakan beberapa bahasa pemrograman web, yaitu PHP untuk logika sistem, HTML dan CSS untuk tampilan antarmuka pengguna serta database MySQL untuk penyimpanan data.

Dalam proses *coding*, kode program disusun secara terstruktur agar memudahkan proses pengembangan aplikasi. Struktur kode program terdiri dari beberapa file yang memiliki fungsi masing-masing dalam menjalankan fitur aplikasi, seperti proses autentikasi pengguna, pengelolaan data *Key Performance Indicator* (KPI), proses verifikasi KPI, pengunggahan dokumen realisasi, serta pembuatan laporan.

Salah satu contoh implementasi pada tahap *coding* adalah proses autentikasi pengguna yang dilakukan melalui halaman login. Pada proses ini, sistem akan memverifikasi kesesuaian *username* dan *password* yang dimasukkan oleh pengguna dengan data yang tersimpan pada basis data. Proses verifikasi password dilakukan menggunakan fungsi *password_verify()* pada PHP untuk memastikan keamanan data pengguna. Contoh implementasi proses autentikasi pengguna ditunjukkan pada Kode Program 1.

Kode Program 1. Implementasi Autentikasi Pengguna

```
<?php
session_start();
require 'config.php';

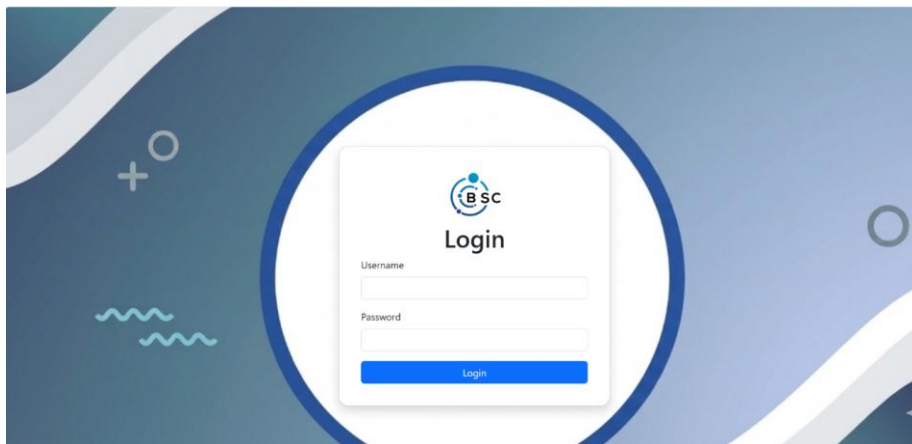
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = $_POST['username'];
    $password = $_POST['password'];

    $stmt = $pdo->prepare("SELECT id, password, role, group_id FROM users WHERE username = ?");
    $stmt->execute([$username]);
    $user = $stmt->fetch();

    if ($user && password_verify($password, $user['password'])) {
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['role'] = $user['role'];
        $_SESSION['group_id'] = $user['group_id'];
        header('Location: modul.php');
    } else {
        $error = "Invalid credentials.";
    }
}
?>
```

Kode program diatas berfungsi untuk melakukan proses autentikasi pengguna dengan mencocokkan data *username* dan *password* yang dimasukkan dengan data yang tersimpan pada basis data. Jika data yang dimasukkan sesuai, maka sistem akan membuat sesi pengguna dan mengarahkan pengguna ke halaman utama aplikasi sesuai dengan hak akses yang dimiliki. Sebaliknya, jika data tidak sesuai maka sistem akan menampilkan pesan kesalahan dan pengguna tidak dapat mengakses sistem.

Setelah proses *coding* selesai dilakukan, sistem menghasilkan tampilan antarmuka yang dapat digunakan oleh pengguna untuk berinteraksi dengan aplikasi. Sebelum mengakses aplikasi *Balanced Scorecard*, setiap pengguna baik admin, leader group, dan pimpinan harus login menggunakan akun yang sudah terdaftar. Pada halaman login yang ditunjukkan oleh gambar 4 menunjukkan halaman login sebagai antarmuka awal autentikasi pengguna, yang menampilkan form input *username* dan *password*.



Gambar 4. Halaman Login KPI

Setelah berhasil login, aplikasi secara otomatis mengarahkan pengguna ke dashboard sesuai dengan hak akses masing-masing, sehingga fitur yang ditampilkan hanya relevan dengan peran pengguna tersebut. Mekanisme ini bertujuan untuk menjaga integritas data KPI serta mencegah akses tidak sah terhadap informasi kinerja organisasi.

Pada halaman add KPI yang ditunjukkan oleh Gambar 5, admin menginput judul KPI, deskripsi, penanggung jawab, serta batas waktu pencapaian KPI ke dalam aplikasi. Seluruh data KPI yang telah diinput tersimpan secara terpusat dan dapat diperbarui apabila diperlukan. Proses ini menggantikan mekanisme manual yang sebelumnya dilakukan melalui dokumen terpisah, sehingga mengurangi risiko kesalahan pencatatan dan meningkatkan konsistensi data.

Gambar 5. Halaman Add KPI

Setelah KPI ditugaskan, leader group melakukan proses verifikasi terhadap KPI yang diajukan oleh admin. Pada tahap ini, leader group meninjau kesesuaian KPI dengan target strategis organisasi serta kelengkapan informasi yang disajikan. Apabila KPI dinilai belum sesuai, aplikasi menyediakan fitur pengembalian KPI kepada admin disertai catatan perbaikan. Sebaliknya, KPI yang telah memenuhi kriteria dapat disetujui untuk dilanjutkan ke tahap persetujuan pimpinan. Proses verifikasi ini dilakukan secara terintegrasi dalam aplikasi, sehingga mempercepat alur komunikasi dan pengambilan keputusan.

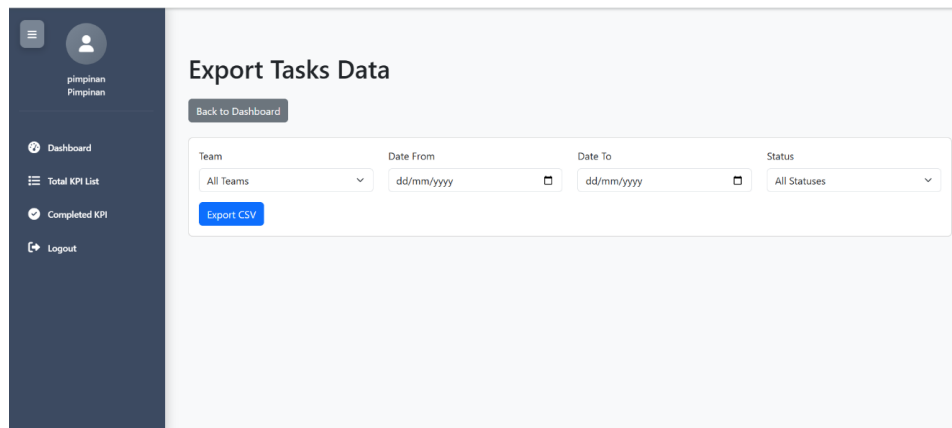
Pada halaman detail KPI yang ditunjukkan oleh Gambar 6, leader group dapat melihat informasi lengkap KPI, seperti judul, deskripsi, penugasan, deadline, dan status KPI. Halaman ini menjadi dasar bagi leader group untuk melakukan evaluasi sebelum mengambil keputusan.

Gambar 6. Halaman KPI Details

Pimpinan memiliki kewenangan untuk melakukan persetujuan akhir terhadap KPI yang telah diverifikasi oleh leader group. Melalui aplikasi ini, pimpinan dapat mengakses informasi KPI secara menyeluruh, termasuk riwayat verifikasi dan status KPI. Keputusan persetujuan dilakukan secara langsung melalui aplikasi tanpa memerlukan proses manual tambahan. Mekanisme ini memastikan bahwa pengambilan keputusan strategis dilakukan secara lebih efisien dan berbasis data yang akurat.

Setelah KPI disetujui, leader group bertanggung jawab untuk mengunggah dokumen realisasi sebagai bukti pencapaian KPI. Dokumen yang diunggah dapat berupa laporan, data pendukung, atau dokumen lain yang relevan. Selanjutnya, pimpinan melakukan evaluasi terhadap dokumen realisasi tersebut melalui aplikasi. Seluruh dokumen tersimpan dalam basis data terpusat sehingga mudah ditelusuri kembali dan digunakan sebagai arsip kinerja organisasi. Proses ini meningkatkan transparansi serta akuntabilitas pengelolaan kinerja.

Pada tahap akhir, aplikasi menyediakan fitur pelaporan dan ekspor data KPI yang dapat diakses oleh pimpinan. Laporan yang dihasilkan mencakup informasi KPI, status pencapaian, serta dokumen eviden yang telah diverifikasi. Fitur ini mendukung proses evaluasi kinerja secara menyeluruh serta memudahkan manajemen dalam menyusun laporan kinerja organisasi. Dengan adanya pelaporan terintegrasi, proses pemantauan kinerja dapat dilakukan secara lebih efektif dan efisien. Tahap akhir yang ditunjukkan Gambar 7 yaitu halaman *export* data, Pimpinan dapat mencetak laporan KPI secara keseluruhan. Sistem menyediakan fitur ekspor data KPI dalam bentuk laporan atau file (CSV) yang berisi informasi KPI, status, serta dokumen eviden sebagai arsip dan bahan evaluasi kinerja.



Gambar 7. Halaman Export Tasks Data

Fitur-fitur lain dalam aplikasi BSC, seperti proses autentikasi pengguna, penugasan KPI oleh admin, pengunggahan dokumen realisasi oleh *leader group*, serta persetujuan akhir oleh pimpinan, dirancang dengan alur yang terintegrasi dan konsisten sesuai dengan peran masing-masing pengguna. Seluruh proses tersebut berjalan secara sistematis dalam satu aplikasi terpusat, sehingga mendukung pemantauan status KPI secara end-to-end mulai dari penetapan KPI hingga pelaporan akhir.

E. Testing (Pengujian)

Pengujian merupakan tahapan penting dalam pengembangan aplikasi *Balanced Scorecard* untuk memastikan seluruh fungsi aplikasi berjalan sesuai dengan kebutuhan pengguna dan alur bisnis yang telah ditetapkan. Pada penelitian ini, metode pengujian yang digunakan adalah *black box* testing, yaitu metode pengujian yang berfokus pada pengujian fungsionalitas aplikasi tanpa memperhatikan struktur kode atau desain internal. Pengujian dilakukan dengan memberikan input pada setiap fitur aplikasi dan mengevaluasi apakah *output* yang dihasilkan telah sesuai dengan yang diharapkan. Fitur-fitur utama yang diuji meliputi proses login pengguna, input KPI oleh admin, verifikasi KPI oleh leader, persetujuan KPI oleh pimpinan, pengunggahan dokumen realisasi, serta pencetakan laporan KPI. Seluruh pengujian dilakukan berdasarkan skenario yang telah dirancang, dan hasil pengujian tersebut disajikan secara rinci pada Tabel 2.

TABEL 2
 PENGUJIAN *BLACKBOX*

Fitur	User	Input	Output yang diharapkan	Hasil pengujian
Login	Admin, Group Leader (pmo leader, eis leader, itops leader), pimpinan	User memasukkan username dan password valid	User berhasil masuk ke dashboard sesuai role	Valid
		User memasukkan data tidak valid	Sistem menampilkan pesan kesalahan	Valid
Input KPI	Admin	Admin menambahkan KPI baru	KPI berhasil disimpan dan terkirim ke Leader	Valid
		Admin mengosongkan data wajib	Sistem menampilkan peringatan	Valid
Verifikasi KPI	Leader Group (pmo leader, eis leader, itops leader)	Leader menyetujui KPI	KPI akan masuk ke pimpinan	Valid
		Leader mengembalikan KPI	KPI kembali ke Admin dengan catatan	Valid
Approve KPI	Pimpinan	Pimpinan menyetujui KPI	KPI telah disetujui dan masuk ke tahap realisasi	Valid
Input Realisasi	Leader Group (pmo leader, eis leader, itops leader)	Leader mengunggah dokumen realisasi	Dokumen realisasi tersimpan	Valid
		Leader tidak mengunggah dokumen	Sistem menampilkan peringatan	Valid
Approve Realisasi	Pimpinan	Pimpinan menyetujui realisasi	Status KPI menjadi selesai	Valid
Cetak Laporan	Pimpinan	Pimpinan mencetak laporan KPI	Laporan KPI berhasil diunduh	Valid

Selain pengujian fungsional menggunakan metode *black box testing*, evaluasi kinerja aplikasi juga dilakukan secara kuantitatif dengan membandingkan waktu proses verifikasi KPI sebelum dan sesudah penerapan aplikasi BSC. Sebelum aplikasi diimplementasikan, proses verifikasi KPI membutuhkan waktu rata-rata 3–5 hari kerja karena dilakukan secara manual dan terpisah antar unit. Setelah penerapan aplikasi BSC, waktu pengelolaan dan verifikasi KPI dapat dipersingkat menjadi 1-2 hari kerja karena seluruh proses dilakukan secara terintegrasi dalam satu aplikasi. Hasil ini menunjukkan bahwa aplikasi yang dikembangkan tidak hanya berfungsi sesuai dengan kebutuhan yang telah ditetapkan, tetapi juga mampu meningkatkan efisiensi proses pengelolaan KPI secara signifikan yang tercermin dari berkurangnya waktu verifikasi KPI, sehingga mendukung pengambilan keputusan manajerial yang lebih tepat waktu.

Jika dibandingkan dengan penelitian terdahulu, penelitian ini menunjukkan pengembangan yang lebih lanjut dari sisi implementasi aplikasi. Penelitian sebelumnya umumnya masih menunjukkan bahwa proses pengukuran kinerja dan penentuan *Key Performance Indicator* (KPI) dilakukan secara manual atau belum didukung oleh sistem aplikasi yang terintegrasi. Kondisi tersebut menyebabkan proses pengelolaan KPI menjadi kurang efisien dan berpotensi menimbulkan kesalahan dalam proses evaluasi kinerja. Dalam penelitian ini, pengelolaan KPI dikembangkan melalui perancangan aplikasi *Balanced Scorecard* yang terintegrasi dengan menggunakan metode *Extreme Programming* sebagai pendekatan pengembangan sistem. Melalui pendekatan tersebut, aplikasi yang dihasilkan mampu mendukung proses pengelolaan KPI secara lebih terstruktur, mulai dari penginputan data KPI, proses verifikasi, hingga penyajian laporan kinerja, sehingga proses pengelolaan kinerja dapat dilakukan dengan lebih efektif dibandingkan dengan penelitian sebelumnya.

Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan antara lain pengujian aplikasi yang hanya dilakukan hanya berfokus pada aspek fungsional melalui *black box testing*. Selain itu, evaluasi juga masih terbatas pada perbandingan waktu proses verifikasi sebelum dan sesudah aplikasi diterapkan. Penelitian ini belum melibatkan pengukuran yang lebih luas, seperti tingkat kepuasan pengguna. Oleh karena itu, pada penelitian selanjutnya pengembangan aplikasi dapat diarahkan pada penambahan metode evaluasi yang lebih komprehensif agar kualitas pengelolaan kinerja perusahaan dapat ditingkatkan.

IV. SIMPULAN

Penelitian ini menghasilkan pengembangan aplikasi *Balanced Scorecard* sebagai sistem informasi manajemen kinerja yang mendukung pengelolaan *Key Performance Indicator* (KPI) secara terstruktur dan terintegrasi. Aplikasi *Balanced Scorecard* yang dikembangkan mampu memfasilitasi seluruh tahapan pengelolaan KPI, mulai dari penetapan KPI oleh admin, proses verifikasi oleh leader group, persetujuan oleh pimpinan, pengelolaan dokumen realisasi, hingga penyajian laporan kinerja secara sistematis dalam satu aplikasi terpusat. Dengan demikian, aplikasi yang dihasilkan telah memenuhi tujuan penelitian dalam menyediakan solusi digital untuk pengelolaan kinerja organisasi. Penerapan metode *Extreme Programming* dalam pengembangan aplikasi mendukung proses pengembangan yang dilakukan secara bertahap melalui tahapan *planning, design, coding* dan *testing*. Pendekatan ini mendukung pengembangan aplikasi yang lebih fleksibel serta memudahkan penyesuaian terhadap kebutuhan pengguna dalam proses pengembangan berlangsung. Hasil pengujian menggunakan metode *black box testing* menunjukkan bahwa seluruh fungsi utama aplikasi berjalan sesuai dengan kebutuhan yang telah ditentukan. Selain itu, evaluasi terhadap proses bisnis menunjukkan bahwa implementasi aplikasi *Balanced Scorecard* mampu meningkatkan efisiensi proses pengelolaan KPI, khususnya dalam mempercepat proses verifikasi dan pelaporan kinerja. Hal ini menunjukkan bahwa penerapan sistem informasi terintegrasi berperan penting dalam mendukung pengelolaan kinerja organisasi secara lebih sistematis dan terstruktur. Secara praktis, implementasi aplikasi *Balanced Scorecard* ditujukan dapat meningkatkan efektivitas pemantauan kinerja, mendukung pengambilan keputusan strategis secara tepat waktu, serta mendorong pengelolaan kinerja organisasi yang lebih akuntabel dan terstruktur. Namun, penelitian ini masih memiliki keterbatasan, yaitu evaluasi aplikasi yang dilakukan masih terbatas pada pengujian fungsional dan uji coba operasional terbatas, sehingga belum mencakup evaluasi penggunaan jangka panjang serta pengukuran tingkat kepuasan pengguna secara komprehensif. Oleh karena itu, penelitian selanjutnya disarankan untuk melakukan evaluasi yang lebih menyeluruh dengan melibatkan lebih banyak pengguna, misalnya melalui survei kepuasan, guna memperoleh gambaran yang lebih komprehensif mengenai kualitas dan penerimaan aplikasi. Dengan pengembangan dan evaluasi lanjutan tersebut, aplikasi berpotensi untuk diimplementasikan secara lebih optimal pada skala organisasi yang lebih luas.

DAFTAR PUSTAKA

- [1] G. Adriansyah and P. S. Hadi, "Manajemen Kinerja Berbasis Key Performance Indicator dan Balanced Scorecard," *JISO: Journal of Industrial and Systems Optimization*, vol. 7, no. 1, pp. 1–10, 2024.
- [2] R. R. Ahmad and L. S. Putra, "Desain dan Implementasi Sistem Informasi Manajemen Kinerja Berbasis KPI," *International Journal of Information Systems and Project Management*, vol. 9, no. 3, pp. 45–58, 2021.
- [3] S. Mulyani, H. Setiawan, and D. Kurniawan, "Electronic Balanced Scorecard sebagai Alat Ukur Kinerja Digital di Era Industri 4.0," *Jurnal Sistem Informasi Bisnis*, vol. 13, no. 1, pp. 55–66, 2023.
- [4] N. Nashihuddin and M. Mubin, "Penerapan Konsep Balanced Scorecard dan Objective Matrix Untuk Perancangan Sistem Pengukuran Kinerja Perusahaan," *Jurnal Indonesia Sosial Teknologi*, vol. 3, no. 2, 2022, doi:10.59141/jist.v3i02.366.
- [5] A. Mukhtar, A. Setiawan, and D. A. Walujo, "Applied of Balanced Scorecard (BSC) Approach with Determination and Weighting of Key Performance Indicator (KPI) for Employee Performance Measurement," *Journal of Applied Industrial Engineering*, vol. 4, no. 2, pp. 131–139, 2021, doi:10.36456/tibuana.4.02.4029.131-139
- [6] Sulkhan and Alek, "Penerapan Metode Balanced ScoreCard Untuk Penentuan Key Performance Indicator di Departemen Production Engineering Pada PT XYZ," *SEED: Journal of Scientific Research*, vol. 2, no. 1, 2025, doi:10.70716/seed.v2i1.
- [7] D. Arisandy and S. Margaretha, "Implementasi Aplikasi CISEA dalam Meningkatkan Kepuasan Kerja Karyawan PT. Bukit Asam Unit Dermaga Kertapati," *Community Development Journal*, vol. 5, no. 1, pp. 1239–1243, 2024.
- [8] Y. Bustomi and S. Rahayu, "Pemanfaatan Teknologi Informasi untuk Penjualan Hasil Pertanian Jeruk di Kabupaten Garut Menggunakan Pendekatan Extreme Programming," *JRST (Jurnal Riset Sains dan Teknologi)*, vol. 7, no. 1, p. 51, 2023, doi:10.30595/jrst.v7i1.15501.
- [9] M. R. Maulana, "Evaluasi Metodologi Waterfall dan Agile," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 1, 2025, doi:10.23960/jitet.v13i1.5900.
- [10] D. Indrianita and R. Napianto, "Pengembangan Sistem Informasi Stok Barang berbasis Website pada Konveksi Mitra Jaya Abadi," *Sistemasi: Jurnal Sistem Informasi*, vol. 14, no. 6, pp. 2589–2601, 2025.
- [11] P. D. Nusantara, "Pengembangan Aplikasi Berbasis Responsive Web Design dengan Metode Extreme Programming," *Jurnal Informatika & Komputasi*, vol. 15, no. 1, pp. 61–73, Apr. 2021.
- [12] Q. E. Fazrin, T. Lisnawati, S. Nurhayati, J. B. Satya, and D. Alamsyah, "Penerapan Metode Pengembangan Sistem Extreme Programming (XP) pada Aplikasi Presensi Karyawan dengan QR Code," *Building of Informatics, Technology and Science (BITS)*, vol. 3, no. 3, pp. 164–170, Dec. 2021, doi:10.47065/bits.v3i3.1018.
- [13] N. Panji Saputra and N. R. D. P. Astuti, "Penerapan Metode Extreme Programming dalam Aplikasi E-Rapor Kurikulum 2013 Berbasis Web," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 4, no. 1, 2023, doi:10.30865/klik.v4i1.1161.
- [14] T. Wahyuni dan A. Z. Muttaqin, "Rancang Bangun Aplikasi Jatah Makan Karyawan Berbasis Android Dengan QR Code (Studi Kasus: PT Wijaya Karya Beton Majalengka)," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 10, no. 3, 2025, pp. -. doi: 10.30591/jpit.v10i3.8642.
- [15] P. Saraswati and Z. Amin, "Sistem Informasi Manajemen Apotek Menggunakan Metode Extreme Programming," *J. Inf. Syst. Res.*, vol. 4, no. 2, pp. 659–668, 2023.
- [16] A. Sayfullloh, "Perancangan Program Penjualan Mainan Berbasis Web Menggunakan Metode Extreme Programming," *REMIK Ris. dan E-Jurnal Manaj. Inform. Komput.*, vol. 5, no. 2, pp. 306–312, 2021.
- [17] P. M. Putra, "Implementasi Metode Extreme Programming dalam Pengembangan Sistem Informasi Penggajian di Kampus Politeknik Ganesha Guru," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, pp. 631–638, 2024, doi:10.23960/jitet.v13i2.6283.